# Data-Driven Context Detection Leveraging Passively-Sensed Nearables for Recognizing Complex Activities of Daily Living

Ali Akbari

Department of Biomedical Engineering, Texas A&M University, College Station, Texas, USA, aliakbari@tamu.edu

Reese Grimsley

Department of Electrical Engineering, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, reeseg@cmu.edu

Roozbeh Jafari

Department of Biomedical Engineering, Computer Science and Engineering, and Electrical and Computer Engineering, Texas A&M University, College Station, Texas, USA, rjafari@tamu.edu

### ABSTRACT

Wearable systems have unlocked new sensing paradigms in various applications such as human activity recognition, which can enhance effectiveness of mobile health applications. Current systems using wearables are not capable of understanding their surroundings, which limits their sensing capabilities. For instance, distinguishing certain activities such as attending a meeting or class, which have similar motion patterns but happen in different contexts, is challenging by merely using wearable motion sensors. This paper focuses on understanding user's surroundings, i.e., environmental context, to enhance capability of wearables, with focus on detecting complex activities of daily living (ADL). We develop a methodology to automatically detect the context using passively observable information broadcasted by devices in users' locale. This system does not require specific infrastructure or additional hardware. We develop a pattern extraction algorithm and probabilistic mapping between the context and activities to reduce the set of probable outcomes. The proposed system contains a general ADL classifier working with motion sensors, learns personalized context, and uses that to reduce the search space of activities to those that occur within a certain context. We collected real-world data of complex ADLs and by narrowing the search space with context, we improve average F1-score from 0.72 to 0.80.

### CCS CONCEPTS

· Human-centered computing → Ubiquitous and mobile computing systems and tools;

### KEYWORDS

Context detection, context-aware, Bluetooth Low Energy, nearables, activity recognition, wearables

## 1 Introduction:

Development of wearable technology has provided a great opportunity for monitoring people's physiological state, behaviors, and activities, which provide important insight that can enhance the effectiveness of mobile health and wellness delivery paradigms [1]–[3]. For instance, in applications such as physical fitness monitoring, diet monitoring, assisted living, and remote health monitoring it is required to detect complex and high-level activities of daily living (ADL). More complex ADLs, such as cooking and housekeeping that require a combination of physical and cognitive efficiencies are critical for independent living. Therefore, there is a need to monitor those activities, especially among elderly and people with certain diseases and/or disabilities. For instance, caregivers could view this information actionable if they would become aware that patients with dementia cook repeatedly or walk at home too often [4]. In Diabetes or obesity, it is vital to monitor for how long the person is sitting or relaxing, how often they engage in physical activities such as running and exercising, and where and how often they eat [5]. In another example, in the application of stress monitoring, it is critical to understand people's daily routine and its changes over time as the progression of certain mental disorders are associated with less engagement in social and job-related activities [6]. Wearable devices have enabled a convenient way to achieve this goal [1]. However, wearable sensors are usually not aware of the context that they are working in. The context is a broad term that includes any additional information that could help better understand a specific situation. For activity recognition with motion sensors' data, this extra information could be the location of the user (e.g., home versus at work), their social interaction, and the time of day. Since the set of possible activities in each of these contexts could be different, understanding this contextual information improves the ability of the system to detect activities with complex motion patterns. Current state-of-the-art systems that use wearables are not capable of distinguishing a large number of complex activities, which may appear similar regarding movement of the hand but with vital differences in context, such as bicep curl exercise versus eating or running in the gym versus running in home. With the development of IoT technology, a wide range of information could be accessible from the devices in a particular environment [4]. Our goal is to leverage freely available data from "nearables" to identify context that could be associated to the user's surroundings, i.e., location, social groups, and nearby objects, in order to reduce the search space for the application of recognizing complex ADLs.

Contextual knowledge helps identify the most probable outcomes within an application domain based on the user's history within that context. With a reduced set of possible outcomes, models can be smaller and decision boundaries more distinct leading to a heightened accuracy. In the case of recognition of ADLs, these outcomes are the activities. The problem of activity recognition has been widely studied due to its paramount importance in remote health monitoring, assisted living, and smarthome development [7]–[9]. For instance, progression of Dementia can be comprehended from detecting ADLs over time. Wearable motion sensors such as accelerometer and gyroscopes have been widely used in conjunction with various machine learning algorithms to detect activities [10], [11]. However, detecting more complex ADLs with merely motion sensors is extremely challenging, if even possible, due to similarity in motion signature of various activities [4]. There is a lack in studies on detecting complex ADLs in the body of literature. To distinguish between those complex activities, understanding the users' surrounding is the key. In this case, knowing the context helps to narrow down the set of probable activities, which in turn facilitates the activity classification. Understanding the context can similarly assist wearables in any other sensing and recognition tasks.

To be useful and ubiquitous, this context (i.e., users' surrounding in this paper) must be developed realistically: the user should not have to deploy a network of sensors or other devices everywhere they go. Instead, context detection should take advantage of what exists already rather than burdening the user, and it must be personalized to individual's routine. Further, it should be scalable to many contexts and a variety of applications. Herein lies the issue with many prior works on context. To achieve good results, studies may constrain the environment or set of contexts they wish to discover. A common approach is to use GPS for location identification as an important context [12]. However, GPS-based solutions cannot account for social contexts that may exist irrespective of location. In addition, approaches relying on GPS have numerous issues from line-of-sight requirements to high power consumption to privacy concerns. Reliance on infrastructure placed by the user is another common approach; for instance, by placing known Bluetooth beacons in specific locations, researchers were able to detect a large set of complicated activities leveraging location information [4]. However, this method places a burden on users and is not scalable and ubiquitous as it requires the users to place a specific device in each room.

For an unobtrusive, scalable, and data-driven context detection we use freely available Bluetooth low energy (BLE) data broadcasted by devices in users' surrounding. It is also known as nearables [13]. These nearables could effectively be all devices equipped with Bluetooth such that they are passively detectable from the wireless signals they transmit. It is worth mentioning that, in this study we do not need to deploy any specific BLE device. Instead, we leverage the BLE data broadcasted by any device in the vicinity of the user and rely on the consistency of those devices over time to infer the context, which is further used to improve recognition of ADLs via a single wrist-worn motion sensor. To best of our knowledge, this is the first work that leverages freely available BLE information broadcasted by any device around the users for detection of such environmental context. Yet, free data is not necessarily good data. The reliable BLE devices will be observed consistently within a context, but many others will be inconsistent. Considering only a single outcome within the application space, i.e., a single ADL, we search for consistent patterns in the BLE devices scanned when this ADL was known to occur. Consistently co-present BLE devices are the basis for extracting these context patterns. With these useful patterns identified, we look back to the entire application domain to find which patterns were important to which ADLs. Using a probabilistic framework, we create an a-posteriori probability for each pattern-ADL pair suggesting the likelihood that an ADL is being performed when a context is observed. In this way, the passively sensed context will directly reduce the set of possible outcomes, i.e., the search space, the model needs to consider. The proposed system can start with a general activity classifier trained only on motion sensors data and learn the context from BLEs in an incremental and personalized fashion. It should be noted that the context training must be personalized since the set of BLE devices that each user visits during their daily living is unique and different from other people. The set of ADLs that we are targeting in this paper, are unique in a sense that it is very challenging to detect such complex and high-level activities with a single wrist-worn motions sensors in uncontrolled environments. It is important to note that the proposed context identification technique is not limited to ADL recognition. Contributions of this paper are as follows:

• An unsupervised pattern extraction method for detecting contextual patterns in static and mobile nearables.

• A probabilistic model leveraging the mutual relationship of context and the application domain such that the application helps build context and the resulting context improves performance of the application.

• Search space reduction based on the present context, permitting the context-aware application to use smaller models with higher accuracy rates.

• A case study on recognizing complex ADLs in the wild, built on 100+ days worth of real-world data collected with smartwatches, to demonstrate the effectiveness of our methods without constraining the environment.

## 2 Related Works

Context-awareness in mobile computing is considered an integral component of the ubiquitous computing paradigm, i.e., the third-wave of computing [14], [15]. The idea that an understanding of the current situation will help computational models is by no means a new one. There has been much work into the structure of context and its usage such that it can be applied to various applications through a centralized framework [16], [17]. Often, these studies build context or suggest building context by merging the output of heterogeneous sensors, e.g., GPS, proximity sensors, and microphone into some structured yet flexible format, such as an ontology, for other applications to use [4], [18]. Since we fixate on the user-centric context, we consider what the user's mobile devices are capable of sensing.

A common approach to context detection is to use technology embedded in the infrastructure of the interesting contexts, such as specific rooms in the home or the user's desk and common meeting rooms at work [19], [20]; these may be considered "logical" or "semantic" locations. Prior investigations have studied using Bluetooth beacons/metadata to detect location [21], social interaction [22], and activity recognition [4], in both supervised and unsupervised manners. Bluetooth-based localization was used to push mobile advertisements to user's cellphones [21]. This system performs localization based on the presence of a known Bluetooth device in the vicinity of a fixed Bluetooth sensor. Therefore, this system needs to know user's Bluetooth address and phone number as well as specific location of the Bluetooth sensor. Another study proposed a probabilistic matrix factorization method for identifying both time characteristic and people involved in a person's social circle using Bluetooth scans [22]. Although this is an unsupervised method that reveals the social circles in which a person is involved, this system assumes that all the Bluetooth packets belong to the cellphones of the study participants. In other words, it ignores the BLE packets coming from stationary devices such as a smart TV. Another study on ADL recognition placed one known BLE beacon into each room, and used the RSSI from each beacon as an input feature for classification; the classifier thus indirectly uses context [4], [19]. Yet if the location and identity of each beacon is a priori knowledge, it becomes trivial to determine the user's logical location [4]. This paper used BLE beacons placed in known locations to identify context, and using that it could extensively improve the current activity recognition systems by detecting complex and fine-grained ADLs, which are challenging to be detected via only motion sensors. They explored activities such as sitting and eating, sitting on sofa vs. bed, standing and using sink vs. talking, walking outdoor to indoor vs. indoor to outdoor, and lying on bed vs. sofa. They showed that by adding location information to the motion data, the accuracy improved from ~78% to ~85%. However, the primary issue here is that the user must deploy infrastructure and provide corresponding information to the context detection system. In all these studies, some prior knowledge about the type of devices and their owners is assumed to be accessible. In contrast our proposed system does not leverage any prior knowledge about the type of BLE devices and their specific location. Moreover, it does not need one to deploy any device in particular locations.

Sensing absolute location via a GPS sensor is another option [23]–[25]. The location captured by GPS has been used to narrow down the search space for classifying the type of food from images [17]. However, several issues preclude GPS from being a good context sensor: 1) it only detects location and is not capable of detecting user interactions as well as other environmental information, 2) it requires line of sight and thus, cannot be used indoors, 3) absolute location has a privacy stigma, and 4) it is not power efficient [26].

Another approach to coarse-grained localization for context detection is through unsupervised fingerprinting of Wi-Fi Access Points [27], [28]; this does not require the user to place infrastructure in their environment, but is only applicable to locational context. These studies facilitate automatic detection of context on passively sensed data. Yet the problem is that they cannot identify social aspects of context well, if at all. While it might be possible to determine social context by passively scanning all Wi-Fi packets, this would require unattainable permissions on commercial devices or specialized hardware [29]. In addition, these techniques need recalibration for each new location that puts extra burden on users. Prior works have attempted to use Bluetooth devices on their own, but for constrained scenarios. These studies often perform statistical feature extraction on Bluetooth scans for use in a classifier [20], [30]. However, just like studies which used known devices as features, whether by presence alone or RSSI value, this will at best lead to an indirect learning of context within the model. These techniques may place high importance on specific features whose values depend on context. Over time, the context may evolve, meaning the model may lose its effectiveness over time. To our knowledge, few studies in general exist which facilitate purely data-driven detection of context. Further, no such studies exist for automatically detecting both locational and social contexts using passively observed nearables. We note that our algorithm does not explicitly derive social interactions from BLE beacons. In fact, since our proposed algorithm considers all regularly observed BLEs from any devices around the user for context modeling, the identified contexts could potentially be related to interaction between users (e.g., when the BLE comes from a friend's device), actual location (e.g., when BLE belongs to a stationary device such as a smart TV in the living room), or objects and devices (e.g., when BLE belongs to user's vehicle). However, our algorithm is blind to the exact type of social interaction. In fact, the main advantage of the proposed method over existing BLE-based methods is learning the context from all BLE packets possible instead of limiting the system to a set of pre-known BLE devices or placing BLE beacons in particular locations.

# 3 Methods

The creation of context in a completely unsupervised way is not a trivial task; it becomes significantly more difficult if many elements used to build the context are unreliable. This is the case when considering passively sensed devices in the user's vicinity because many nearables could be mobile. For the given application of activity recognition, a particular activity may occur in several different contexts; further, there may be

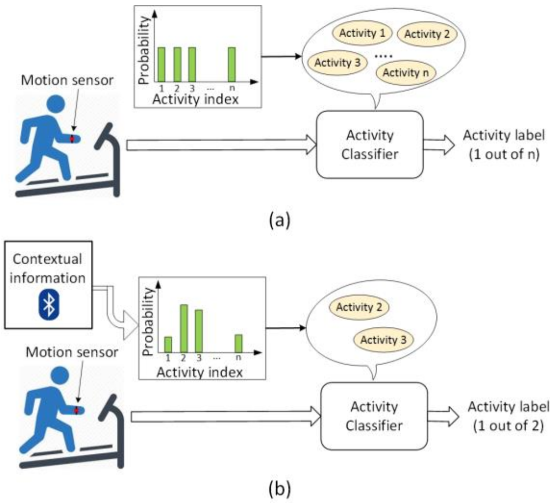


**Figure 1. Classification search space (a) without and (b) with context**

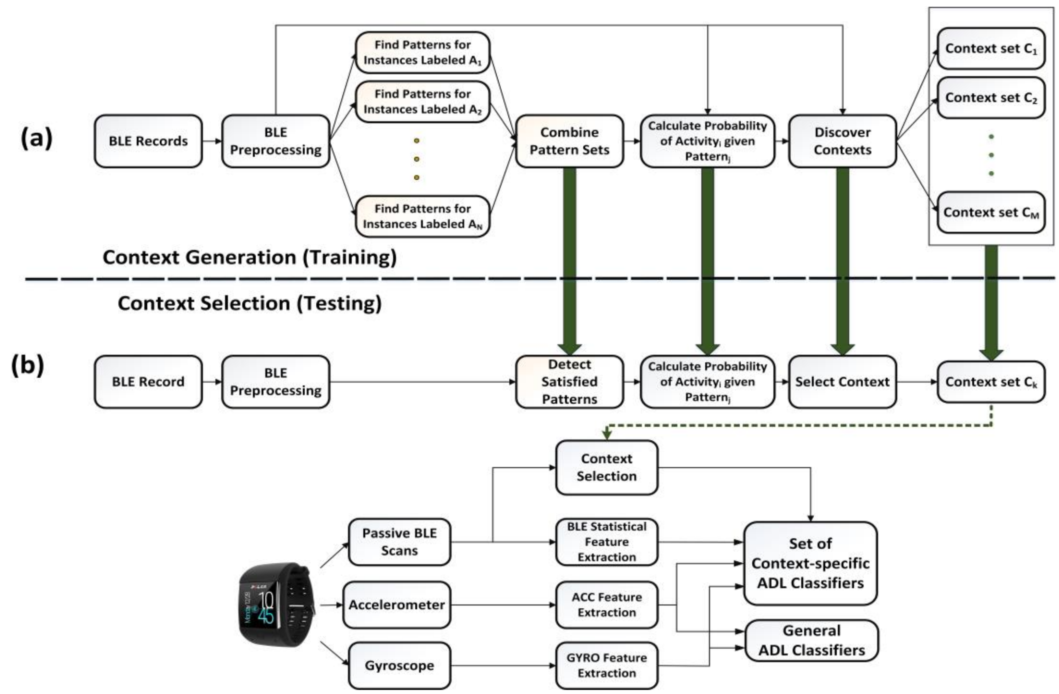


**Figure 2. Overview of the proposed system for context learning**

multiple distinct activities possible within the same context. In this way, the relation between the user's action and context may not be 1:1. Here we focus on complex ADLs such that motion signals alone are not enough to distinguish all activities. Examples include attending a meeting or class, cooking, eating, working at one's desk, etc. Knowledge of the environment can help alleviate this issue. Figure 1 shows the effect of context w.r.t. ADL recognition. No context is used in Figure 1-a where any of *n* activities may be detected by the model. However, context helps estimate which activities are actually feasible, shown in Figure 1-b. In this study, contextual knowledge is derived from passively sensed BLE devices. Throughout this paper, we use words BLE devices and nearables interchangeably.

We assume that there is a general activity classifier trained on the data of the motion sensor embedded in a smartwatch. This classifier has a limited accuracy due to similarity in the motion patterns of complex activities. As the users start using the smartwatch during their normal daily life, the system starts to learn the

## Table 1: Term Definitions

| Term | Definition |
|---|---|
| $M$ | Set of all observed BLE beacons during data collection |
| $m_j$ | The MAC address of the $j^{th}$ element in $M$ |
| $R$ | The set of BLE MAC addresses (i.e., record) scanned during some one-minute segment of time |
| $B_i$ | A binary vector of size $\|M\|$ representing the BLE record at the $i^{th}$ timestamp |
| $b_j$ | A binary element corresponding to $j^{th}$ MAC in $M$ |
| $x_i$ | Sensor readings at $i^{th}$ timestamp |
| $y$ | The output label at $i^{th}$ timestamp |
| $L$ | Total number of possible output labels |
| $A$ | Set of output labels $A = \{1, \dots, L\}$ |
| $a$ | An output label where $a \in A$ |
| $n_a$ | Total number of samples with label $a$ |
| $p_k$ | $k^{th}$ context pattern which is composed of one or more BLE beacons |
| $K$ | Total number of context patterns |
| $Co_p^a$ | The coverage of pattern $p$ for label $a$; i.e., the percentage of samples labeled $a$ for which context pattern $p$ is observed |
| $\Phi$ | An $L$ by $K$ matrix that includes the probability of action $a_i$ given context pattern $p_j$ in $i^{th}$ row and $j^{th}$ column |
| $P^R$ | Set of patterns satisfied by record $R$ |
| $\phi^R$ | A subset of columns in $\Phi$ that corresponds to the patterns $P^R$ satisfied by record $R$ |
| $\Theta$ | A small lower threshold to filter out patterns that are rarely present during an action |
| $IoU(p_1, p_2)$ | Intersect over union calculated between two patterns |
| $N_{p_1}$ | Number of samples/instances in which pattern $p_1$ is observed |
| $S_u$ | Motion signal, i.e., either acceleration or angular velocity, along axis $u$ |

context by looking at the patterns of BLE devices around the user. The overview of the proposed system is shown in Figure 2. When a new user starts using the device, the general activity classifier is used to assign labels to each sample using motion data, and the most certain labels are chosen for context learning. In addition to motion data, we collect BLE scans broadcasted by devices in the vicinity of the user. After a period of data collection, as shown in Figure 2-a, the context is created separately for each possible action the user may take based off of patterns extracted from BLE devices scanned while that action was being performed. A probabilistic model then relates these possible outcomes to the observed contexts which is used to narrow the search space to probable actions only. A separate classifier is trained for each set of actions that share context that is called context-specific classifier. For a new sample then, as shown in Figure 2-b, the set of satisfied patterns is identified based on observed BLE devices. Using the probabilistic model, the patterns are then mapped to the set of probable actions; these are used to select the appropriate context-specific classifier, which is trained only on that reduced set of possible outcomes. When a new unknown context is observed, again the most reliable labels generated by the general classifier are gathered to retrain the context patterns. In fact, the context learning is an ongoing process that never stops. The ongoing training is required because the set of BLE devices around a user can dynamically change over time. In the following, we first explain our context training approach including BLE preprocessing, extracting context patterns and associating them to the activities in a probabilistic fashion. Lastly, in Section 3.3, we explain the process of labeling the data on-the-fly by leveraging general activity classifier and context-specific activity classifier. Before introducing the details of our approach, notation definitions are summarized in Table 1 to enhance the readability of the article and equations.

## 3.1 BLE Representation and Preprocessing

Many BLE devices periodically transmit ‘advertisement’ packets containing basic information about a ‘beacon’, allowing us to collect a small amount of data about each advertising device in the user’s vicinity without any handshaking between the transmitter and receiver. Traditionally, beacons are BLE devices whose primary purpose is to advertise information about the environment; for this study, we refer to all BLE devices as beacons because we are only interested in their advertisement packet which indicates their presence in the environment. BLE is an appropriate modality for detecting both the social and locational components of environmental context because it is a protocol common to both static and mobile devices. The former may be telemetry beacons, infrastructural sensors or other stationary electronics while the latter may be


**Figure 3. Mobility of user and nearby BLE beacons**

smartphones, wireless headphones, wearable electronics, etc. Clearly, static devices would be effective for determining location similar to studies using Wi-Fi Access Point fingerprints [26], [27], [30], and mobile devices, when exhibiting consistent patterns with respect to each other, could help determine social groups [22]. However, distinguishing between static and mobile devices is difficult as both the user and the BLE device could be mobile. Please note that the main assumption here is that there is no information available about the type of the device when its BLE packet is received. This is a realistic assumption because: 1) many BLE packets include the MAC id of the device but there is no information about the type of the device in their packet, and 2) relying on the type of the devices will significantly limit scalability of the system. Thus, we choose to model all beacons as equivalently purposed, and we refer to all BLE devices as 'beacons' regardless of their physical purpose.

We log the 6-byte MAC address of each device that we scan. Collecting all MAC addresses of beacons over many days will nonetheless result in noisy data. For instance, many people encountered in a public place will not be seen again. Clearly, most of these beacons will be noise. We use a simple rule to filter out beacons which are obvious noise: if a device is present for a short period of time (less than 15 minutes a day in our study) then it is not useful. This simple filtering rule removes a large portion of beacons; for example, for one of our subjects, we filter from around 20,000 beacons detected across seven days down to around 8,200. With this reduced set of beacons, we aim to find patterns in the BLE records to help understand the user's context.

As depicted in Figure 3, the user's context may be represented by the devices near them at a point in time. This provides a fingerprint of the user's context. However, a single scan may miss some nearby beacons due to differences in the scanning period versus advertisement intervals. For this reason, we represent the user's current environment using the set of BLE beacons that are detected over multiple scans within a relatively short interval of time (one minute for this study based on the limitations of the smartwatch device to complete a scan of all nearby BLE devices). Let us consider this one-minute scan to be a BLE 'record'. Intuitively, we can represent a record simply by making a binary vector $B$ of length $|M|$ where $M$ is the set of all observed beacons during data collection period after filtering. Each element of a binary vector corresponds to a MAC address such that an element corresponding to beacon $b_i$ is 1 if MAC address $m_i$ was present and zero otherwise. Equation 1 formalizes this, where $R$ is the set of all MAC addresses scanned during a one-minute segment.

$$B = (b_1, b_2, \dots, b_{|M|}) \quad , \quad B[i] = b_i = \begin{cases} 1 & , \quad m_i \in R \\ 0 & , \quad o.w. \end{cases} \qquad (1)$$

## 3.2 Data-Driven Context Detection

When the user consistently visits a particular context, certain patterns should exist in the BLE scans corresponding to that particular context. The context patterns should be generalizable in the sense that they should not be too sensitive to the requirement of many observed beacons being co-present. For instance, in the training data, we might see three beacons that are always present together when the user is at home. If one of those beacons is no longer present during the testing time, the system should still be able to recognize the home context. At the same time, the patterns should also be selective. Beacons that are present in too many different contexts, such as the cellphone of the user, are not selective enough since they cannot distinguish different contexts. Therefore, we are interested in extracting consistent, selective, and generalized patterns of beacons that help understand the users' surrounding and facilitate ADL recognition.

### 3.2.1 Context Pattern Formulation

Recall that in this study, "context" refers to information about the user's surroundings; we assume that there is no provided information available about the true context of the user. In other words, there is no supervision and consequently, no labels available for the context which is a realistic assumption in real-world scenarios. In general, we use no a priori knowledge about the context. Although we have knowledge on the user's actions, which herein is provided by a pretrained motion-based activity classifier (or it can be provided by the user, see Section 4.3), there is not a 1:1 correlation between a possible action outcome and the context. Thus, we build the context for each outcome separately from the others. That is, for each action, we describe the context in which that action might be performed by extracting patterns from BLE records.

A BLE context pattern is a set of different BLE beacons that are consistently present in a specific context. To be formal, assume that $(X,Y)$ is all of the training data where $x_i \in X,\ i=1,\dots,n$ is the sensor readings, *e.g.*, accelerometer and gyroscope, for $i^{th}$ data sample and $y_i \in Y$ is the label for $i^{th}$ sample. $Y=\{1,\dots,L\}$ where $L$ is the total number of possible labels. Please note that the label $y_i$ could either be generated by a pretrained motion sensor-based activity classifier or provided by any external source such as the user. In this paper, in section 3.3 we will explain how we leverage the labels generated by the motion-based classifier to avoid burdening the users. We assume that there is a context associated with each sample, which is unknown and should be learnt for each user. In addition to the input data $x_i$, we have a set of observed beacons $B^i$ where $B^i=(b_1^i, b_2^i, \dots, b_{|M|}^i)$ is a vector of beacons in which $b_j^i$ is 1 if $j^{th}$ beacon is present in $i^{th}$ sample and 0 otherwise.

*Definition 3.1 (Context Pattern)*. $p_k$ is called a "context pattern" consisting of a set of specific beacons. Here $k=1,\dots,K$ is the index of the pattern where $K$ is the total number of context patterns extracted from the data. With this definition, one might consider a context pattern as an atomic location or environment; for instance, each pattern may indicate a different room, device or person around the user.

There is always a trade-off between the selectivity and generalizability of context patterns as mentioned before. To deal with this tradeoff, we design a modified version of hierarchical agglomerative clustering (HAC) to generate the patterns, termed accumulative hierarchical agglomerative clustering (AHAC), along with a probabilistic framework for mapping the context patterns to the actions w.r.t. the application domain using an action-context probability (ACP) estimation algorithm. AHAC allows us to have general and selective patterns at the same time. By estimating the probability of user actions given contexts (Section 3.2.3), we prioritize the probable actions then to narrow down the search space.

### 3.2.2 Context Pattern Learning

This section discusses how to extract meaningful patterns that are both generalized and selective from an abundance of observed BLE beacons. Recall that we create the context patterns for each possible outcome separately. For an action $a$, assume that $\mathcal{B}^a$ is the set of all training records that have label $a$ where $\mathcal{B}^a=\{B_1^a, \dots, B_{n_a}^a,\}$ and $n_a$ is the total number of samples with label $a$. Recall that $B$ is a BLE record represented as a one-hot-encoded vector that contains $|M|$ beacons, where $|M|$ is the total number of different beacons in the dataset. We first remove the beacons that are rarely seen during action $a$ using Equation 2:

$$P^a = \left\{ j : \left( \frac{1}{n_a} \sum_{k=1}^{n_a} B_k^a[j] \right) > \Theta \} \ \ , j = 1, \dots, |M| \right\} \qquad (2)$$

where $j$ is the $j^{th}$ beacon in the record, $P^a$ is the set of all beacons that we keep to generate patterns for action $a$, and $\Theta$ is a constant threshold.

After this stage, one can just keep the single beacons as representation of the context for action $a$. However, such a basic method has a poor selectivity as it fails to deal with beacons such as the user's cellphone that is present in many different contexts. To address this issue, we can merge beacons that are often present together in order to create more specific and representative patterns. For this goal, we can utilize a clustering algorithm to put beacons into clusters whose contents represent more specific context patterns. We modify the

**ALGORITHM 1:** AHAC

**Input:** all training records $\mathcal{B}^a$ for action $a$ and its sample size $n_a$
**Output:** set of patterns (*i.e.*, clusters) $P^a$ for action $a$
1- $P^a = [\ ]$
2- **for** *(every beacon j = 1* **to** *|M|)* **do:**
3- **if** $(\frac{1}{n_a}\sum_{k=1}^{n_a} B_k^a[j]) > \Theta$)
4- $P^a \leftarrow j$
5- **end if**
6- **end for**
7- $IoU = 1$
8- Assign each element of $P^a$ to a cluster
9- **while** *( IoU > 0.75 )*
10- find *clusters i* and *j* with max *IoU(i ,j)*
11- $P^a \leftarrow$ [*cluster i* , *cluster j*]
12- $IoU = IoU(i , j)$
13- **end while**

hierarchical agglomerative clustering (HAC) to fit into our problem and call it Accumulative HAC (AHAC). Algorithm 1 shows the steps of AHAC algorithm that is done only during the offline training phase. Intuitively, AHAC elects to keep the unmerged components of a cluster in the entire set rather than removing them once they are merged. In fact, we grow the set of patterns to contain both generalized and selective patterns altogether. Later, by looking at the distribution of patterns in various actions we can effectively select the most probable context and set of activities subsequently.

This method, starts by putting every single beacon in $P^a$ into a single cluster. This means that at the beginning, the number of clusters is equal to $|P^a|$, each of which contains a single BLE beacon. We use the words 'cluster' and 'pattern' interchangeably because once we finish clustering, we use each cluster as a context pattern. We consider a pattern to be satisfied if all beacons within its cluster are present in a data sample. To merge two clusters together at each step of AHAC, we use the intersect over union (IoU) between the two clusters as the measure of similarity. Again, note that we have a separate set of clusters for each possible activity label. Equation 3 shows how IoU is calculated for two patterns. This measure tells us about the proportion of co-occurrence of two patterns. Higher IoU means that the two patterns (i.e., set of BLE beacons) are consistently present simultaneously, which increases the likelihood that they belong to the same context.

$$IoU(p_1, p_2) = \frac{N_{p_{1,2}}}{N_{p_1} + N_{p_2} - N_{p_{1,2}}} \qquad (3)$$

In Equation 3, $N_{p_1}$ and $N_{p_2}$ are the number of samples that pattern $p_1$ and $p_2$ apply to, respectively, and $N_{p_{1,2}}$ is the number of records that apply to both patterns, *i.e.*, both patterns are present at the same time.

At each iteration of clustering, the two patterns/clusters with the maximum IoU are chosen to be merged together. The two selected patterns are merged and added to the whole set of patterns. Here we make a modification to HAC by keeping all the original clusters and adding the merged cluster to the total set, meaning the cluster count increases by one with each iteration. In this way, we refer to our pattern extraction technique as Accumulative HAC (AHAC). We make this change because if we remove patterns $p_1$ and $p_2$ when they are merged, as is the case with unmodified HAC, we progressively create overly specific patterns such that it becomes more difficult to fit our dataset reliably, especially for new data. As opposed to the normal HAC where the number of clusters decreases by one at each iteration, this number increases by one in AHAC, meaning this process will not naturally terminate itself. In our approach, clustering is finished once the maximum IoU of any two previously unmerged patterns is below a threshold, which is set to 75% for our study. This process is done in offline training to create a set of important context patterns.

In fact, this approach can be seen as an iterative clustering algorithm. In first iterations, the BLE beacons are combined together, but as the algorithm proceeds, the clusters (patterns), which are a set of mutually observed beacons measured by Equation 3, are combined to create more specific patterns. Each pattern generated in this way could potentially be a combination of people and/or static devices in a particular location.

During the testing for a new sample, a single BLE scan/record is evaluated to find which patterns are satisfied; a pattern is satisfied when all beacons within that pattern are present in the record. Then, the action-context probability (ACP) estimation engine creates the probability of actions given the satisfied context patterns, selects the best classifier for the set of possible activities, and then sensor data is fed into that chosen classifier to recognize the action.

#### 3.2.3 Action-Context Probability Estimation

This method aims to estimate the probability that a particular outcome (i.e. an activity) occurs given a particular context pattern *p* is seen. Algorithm 2 formalizes this procedure for generating action-context probability matrix; it populates this matrix using Bayes rule to measure the probability of some action given a context pattern. Algorithm 3, then formalizes the ACP estimation procedure in which the distribution of action-context probabilities for context patterns that are satisfied by a BLE record are used to decide which actions are probable such that the search space may be reduced. We use Bayes theorem to represent these action-context probabilities in Equation 4.

$$Pr(a|p) = \frac{(Pr(p|a)Pr(a))}{\sum_{\alpha\in A} Pr(p|a)\,Pr(\alpha)} \qquad (4)$$

**ALGORITHM 2:** Creating ACP Matrix

**Input:** all training samples, set of user actions, set of all patterns,
**Output:** action-context matrix Φ
**for** ***(****each action a = 1* **to** *L****)*** **do:**
    **for** ***(****each pattern p = 1* **to** *K****)*** **do:**
        $\Phi_{a,p} = Co_p^a / \sum_\alpha^L Co_p^\alpha$ //calculated by Equation 4 and 5
    **end for**
**end for**

**ALGORITHM 3:** ACP Estimation

**Input:** the observed record R for the current sample, Φ
**Output:** Set of mostly probable activities *Action_set*
*Action_set* = [ ]
$\phi^R = [\,]$
**for** ***(each*** *pattern* $p_j$***)*** **do:**
    **if** ($p_j \in R$):
        add column *j* of Φ to $\phi^R$
    **end if**
**end for**
**for** ***(each*** *action a = 1* **to** *L****)*** **do:**
    $Prob_a$ = mean $a^{th}$ row of $\phi^R$
    **if** ***(*** $Prob_a > \frac{1+\varepsilon}{}$ ***):***
        *Action_set* ← *a*
    **end if**
**end for**

where $Pr(a)$ is the prior probability of action $a$, for which we consider a uniform distribution, $A$ is the set of all possible actions, and $Pr(p|a)$ is the probability of observing context pattern $p$ while performing action $a$, which can be estimated from the training data as shown in Equation 5.

$$Pr(p|a) = Co_p^a = \frac{\#\ of\ appearance\ of\ p\ in\ a}{\#\ of\ appearance\ of\ a} \quad (5)$$

We call $Co_p^a$ the 'coverage' of pattern $p$ for action $a$, *i.e.*, the percentage of samples labeled $a$ for which context pattern $p$ was satisfied. Assume that for all the $L$ actions possible in the chosen application domain, we have a total of $K$ different context patterns. Then, we use the entire training dataset to generate an $L$ by $K$ matrix, $\Phi$, which is a matrix containing all action-context probabilities. In the $i^{th}$ row and $j^{th}$ column of this matrix, we put the probability of action $a_i$ given context pattern $p_j$ as shown in Equation 6.

$$\Phi_{i,j} = Pr(a_i|p_j) \ (6)$$

We retrieve a subset of this matrix's columns to generate a temporary matrix for the BLE record of the current sample based on the patterns $P^R$ satisfied in the current record $R$. The retrieval of this matrix is shown in Equation 7 as well as Algorithm 3, and allows us to estimate the set of possible actions for the record $R$ that may share context. These actions are called the 'Shared Context Set'.

$$\phi^R = \left(\mathbf{\Phi}_j : p_j \in P^R, j \in \{1, \dots, K\}\right) \ (7)$$

For this goal, we calculate the mean probability for each possible outcome based on the probabilities of that action given each pattern in $P^R$. In other words, we take the average value over the columns in matrix $\phi^R$ for each row. We pick the actions for which this average is greater than a threshold as shown in Equation 8. The threshold in our study is one plus a parameter $\varepsilon$ to reduce sensitivity (in our study $\varepsilon = 0.25$) over the total number of actions, $L$.

$$Shared\ Context\ Set^R = \left\{A_i : \frac{1}{|P^R|}\sum_{j=1}^{|P^R|}\phi_{ij}^R > \frac{1+\varepsilon}{L}\right\} \quad (8)$$

The set of possible actions are identified based on the outcome of Equation 8. In offline training phase, we identify possible sets of actions for all training samples and use them to train context-specific classifiers as explained in Section 3.3. In online testing phase, we first determine the set of possible actions for each new sample based on the context using Algorithm 3, and then pick appropriate context-specific classifier from the pool of classifiers.

## 3.3 General and Context-specific Classification

The context training needs to be personalized since the set of BLE devices around each user could be different. Moreover, for a user, the set of observed BLEs in a certain location may change over time. To account for this, the system should be able to learn the context for each user incrementally and the context learning should be a non-stop process. We assume that, at the beginning, there is a general activity classifier available that is trained merely on motion sensors' data with no contextual knowledge. This general classifier is trained solely on the motion data, but its accuracy obviously affects the performance of the system (see Section 4.3), so the best performance could be achieved if the motion-based classifier is trained or fine-tuned for the data of each user. The system starts learning the context when it is used by a new user and the context learning is updated periodically. Context training process begins with an empty set of patterns for each action $a$, and there is no context-specific classifier available when the user starts using the device for the very first time. For a couple of days, seven days in this study, the system only stores all the motion sensors as well as BLE data and uses the general motion-sensor-based classifier to assign labels to those samples. The samples about which the general classifier is confident, i.e., its confidence score is above a threshold, are selected as the samples for extracting context patterns as described in Section 3.2 and training the context-specific classifiers. Based on the output of ACP algorithm during this offline training, we end up with multiple lists of user actions, where all share context, i.e. Shared Context Sets. Consider an example: based on training data we realize that one subject's activities of 'eating', 'cooking', and 'cleaning' fall into one list as all of them happen in the same location. For each reduced set of probable outcomes, determined using the context, we train a distinct context-specific classifier. Since the context-specific classifiers are trained on a reduced set of activities and they also leverage contextual features, they are more accurate than the general classifier, which may confuse activities with similar motion signature.

After the first period of context learning, the context-specific classifiers are ready to be used to narrow down the list of activities to be classified. In the testing phase, for each sample $i$, we look at the observed BLE record $R^i$ and retrieve the set of satisfied patterns. If there is any pattern $p$ satisfied by the record $R^i$, the appropriate context-specific classifier is chosen using Algorithm 3 and used to detect the activity label. In this case, if the label generated by the general classifier is not in the list of probable activities that are determined by the context, and if the general classifier is confident about its decision, then that sample and the label generated by the general classifier will be stored for context retraining in future. This is important because there might be cases that the user performs a new activity in a previously known context. Please note that, the proposed ACP algorithm that maps the context to the probability of activities, considers the frequency of occurrence of the activity in each context; therefore, if the user randomly performs an activity in a specific context just for a very short time and non-consistently, that outlier will not affect the outcome of the algorithm. If there is no pattern satisfied by $R^i$, which means the user is visiting a previously unknown context, that sample along with the label generated by the general classifier will be stored for context retraining only if the confidence of the general classifier is high. These data that are collected during the testing time are fed into algorithms 1-3 to update patterns and action-context matrix. It should be noted that after every three days of data collection the context training is updated with the new set of labeled data.

Although we select the most confident samples labeled by the general classifier, the overall accuracy of this general classifier can impact the output of the algorithm. The required labels of the activities, however, can be provided by any external source such as the users themselves depending on the availability of that source. For example, the user could provide the labels for all the activities within the first week of data collection for context training and then the system will sporadically ask the user when it faces an unknown context or less confident predictions. In Section 4.3, we compare the effect of getting labels from a reliable source such as the user versus the general motion-based classifier.

For our case study on ADL recognition, we use SVM (with regularization parameter set to 1) as the classifier and use the probability scores as the measure of confidence of the classifier [31]. We extract a set of various time and frequency domain features from accelerometer and gyroscope sensors along with statistical features from BLE beacons as shown in Table 2 [32]. The features from motion sensors are calculated for each of X, Y, and Z axis along with the magnitude of the signals, calculated by Equation 9, which is orientation independent [33].

$$A = \sqrt{S_x^2 + S_y^2 + S_z^2} \qquad (9)$$

In Equation 9, $A$ is the magnitude of the signal, and $S_u$ could be either acceleration or angular velocity signal around axis $u$ measured by accelerometer and gyroscope respectively.

For the BLE statistical features, we consider the number of devices present and the 'turnover' of beacons. Turnover is effectively the change rate of beacons: by looking at how much the set of nearby beacons

**Table 2: Classifier Input Features**

| IMU Features | | | | | BLE Features |
|---|---|---|---|---|---|
| Mean | Min | Variance | Power in 0-0.5, 0.5-1, 1-2, and 2-5 Hz bins | Mean-cross-rate | Number of devices |
| Standard Variation | Max | Root mean square | Frequency of maximum FFT | Autoregressive coefficients of order 10 | Turnover (Beacon Change Rate) |

**Table 3: Set of Activities**

| Activities | | |
|---|---|---|
| Biking | Working | Eating |
| Exercising | Class attending | Cleaning |
| Walking | Meeting | Shopping |
| Driving | Cooking | |
| Relaxing | Getting Ready | |

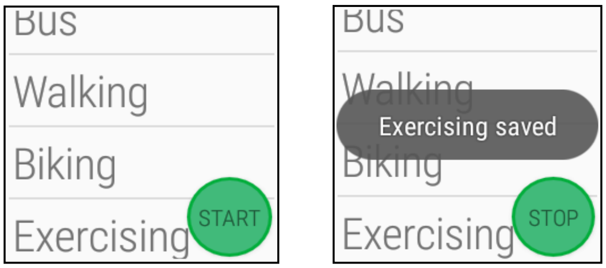


**Figure. 4: (left) data collection app interface, (right) activity logged, data collection running**

changes, we can get a rough idea of how much the environment is changing. If the subject is traveling or stays in a public place, there would be a high rate of changes. We calculate turnover as the IoU (Equation 3) between the current record and the previous one. In total we have 180 features that are normalized to range [0, 1]; 176 of these are calculated from motion signals.

It should be noted that the length and frequency of each activity was very different in this study depending on person's routine. This results in a very unbalanced training dataset. Our strategy to mitigate the effect of unbalanced training data is to reweight the cost functions for each class. Explicitly, we calculate the weights $w_i$ as shown in Equation 10, where $c_i$ is the number of instances of activity $i$ in the training set. These weights are used in the cost function for SVM classifier to mitigate the effect of unbalanced samples.

$$w_i = \frac{\max_j c_j}{c_i} \qquad (10)$$

# 4 EXPERIMENTAL RESULTS

In this section we validate our proposed framework by showing how the environmental context helps recognize high level activities of daily living that could be nearly impossible to distinguish with merely wrist-worn motion sensors. Five subjects were equipped with smart watches and asked to collect data and labels on their daily activities. Each subject collected data for approximately 20 days. All data for this study were collected in the wild while users performed their normal daily activities. For data collection in this study, we used a commercial smart watch, a Polar M600 shown in Figure 4-a, running the Android's smart watch operating system, Wear OS 2.0, based on Android 8.0. We collected accelerometer, gyroscope, and passive BLE data with a custom Android application. This app features a simple interface shown in Figure 4 for the user to start and stop data collection, as well as a list of labels that will log the label and time stamp when an entry in the list is tapped as shown by the pop-up in the figure. We collected around 100+ days worth of data, which includes more than 580 hours, from all the subjects. The data from first two subjects were used for training the general motion-sensor-based classifier while the data of other subjects were used for context training as explained in Section 3.3. Table 3 shows the full list of all activities assessed in this study. In this study, the participants provided the labels for only the listed activities and we ignored the label of other activities not included in Table 3. It should be noted that this is an extensive set of ADLs compared to the existing literature in activity recognition. In fact, detecting such complex ADLs without leveraging contextual information and by merely using motion sensors would be very challenging and inaccurate. In the following, we first show the importance of context in recognizing these complex activities. We then compare

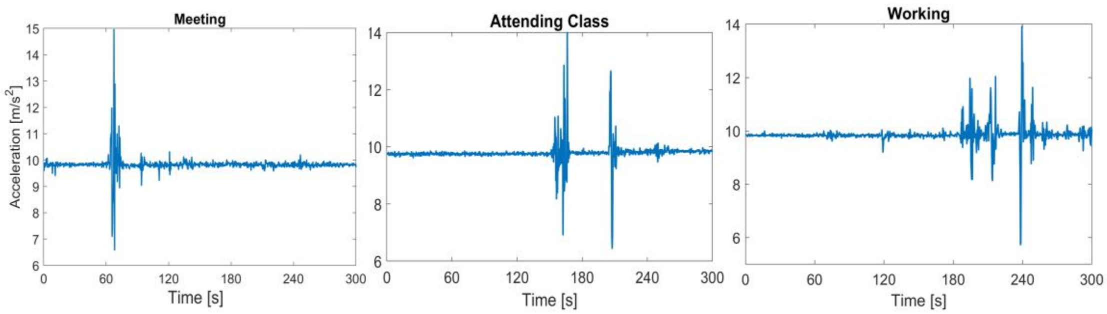


**Figure5. Acceleration signal during three activities with extended periods of sedentary activity**

our context modeling approach with baselines since there is no similar work in modeling the context from freely-available BLE beacons. Finally, we assess the effect of the general classifier's accuracy on the overall performance and the impact of having more reliable source of labeling such as the user. It should be noted that there is no similar study or publicly available dataset in the literature that includes all the BLE scans for a couple of days in uncontrolled environments.

## 4.1 General versus context-specific activity classification

We start by demonstrating the need for context by comparing performance of ADL recognition using the general activity classifier that does not leverage the context with our system that uses the context. We first show that contextual information is essential for detecting complex ADLs. For many people, daily activities consist of long stretches of sedentary positions. For instance, during working or attending a class, there may be minor movements like typing on a keyboard, using a smartphone, or turning pages of a book, but the overall motion characteristics of largely immobile activities are likely to be similar if not indistinguishable. Attempting to recognize activities using only features from motion data, will perform poorly due to the similarity of the signals. Figure 5 shows the acceleration signal from five minutes of attending a class, working, and meeting activities. In our findings, most parts of signals for more sedentary activities, *i.e.*, class, working, meeting, exhibit periods of inactivity with short, intermittent bursts of motion activity. If the duration and intensity of these bursts is to vary across instances of these activities, then there is little to distinguish said activities using motion alone. In contrast motion signals during walking, running, and driving show more unique characteristics that are likely sufficient for classification with merely motion sensors. In truth, experiments have shown acceptable performance for high-mobility activities like walking, biking, and exercising, yet low-mobility ADLs perform poorly when only motion signals are used. Several other examples of such complex ADLs are investigated in [4] by placing pre-known BLE beacons in specific locations in user's home. This poor performance has a dramatic effect on the total accuracy of the system given that sedentary activities usually occur over long portions of time as discussed later in Section 4.2.

To remain fair in comparing our method using context identification with general activity classifier that does not leverage context, for the general classifier we use both the motion features as well as BLE statistical features as listed in Table 2. However, in the case of general activity classifier there is a single classifier that attempts to detect all the activities. In contrast, by modeling the context we have a series of context-specific classifiers each of which works on a specific set of activities. Figure 6 depicts the F1-score for each activity for the three subjects when context is or is not applied to reduce the search space of possible activities. It is clear that including context affects some activities much more than others. Generally, activities which have a greater degree of motion, such as 'biking', 'exercising' or 'walking', do not change much when context is included. In these cases, there is ample information in the motion features alone to confidently decide which activity the user is performing. However, we see this is not the case for sedentary activities such as attending 'class', 'meeting' or 'working'. In each of these, the motion and a few statistics from BLE scans are not enough to consistently choose the correct activity; however, an understanding of the context makes it much easier to detect the correct activity. For instance, including context results in F-1score improvement of 0.17

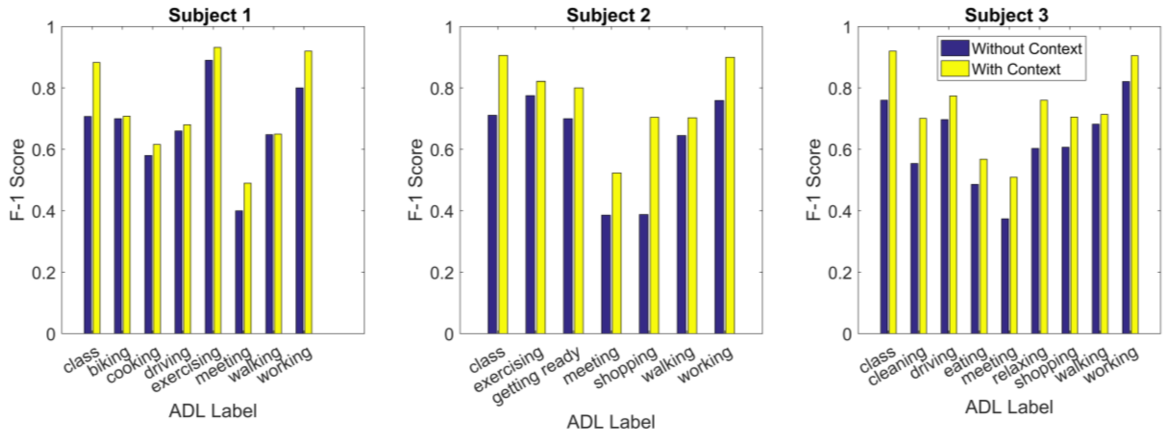


**Figure 6. Comparison of class-based F-1 score between the general activity classifier (without context) and context-specific classifier**

in attending ‘class’, 0.11 in ‘working’, 0.12 in ‘meeting’, 0.18 in ‘cleaning’ and 0.21 in ‘shopping’ on average. Note that the worst performing activity is ‘meeting’: this activity shares context with many instances of ‘working’ activity and has few distinguishing motion patterns of its own, making it extremely difficult to recognize this activity consistently, regardless of context knowledge. Also, modes of transportation such as ‘driving’ and ‘walking’ have lower performance (around 0.62 and 0.68, respectively) than would be expected given the amount of motion. The reason is two-fold: firstly, there were few instances of these activities in the dataset, so even a few misclassifications are highly damaging, and secondly, instances near the time of departure or arrival may detect context patterns from the previous activity, causing them to be incorrectly placed into a context model that does not consider the true activity as a possible outcome. Generally, these locomotive activities were found to have no detectable context.

Nonetheless, we see the weighted average of F-1 score increases from 0.72 to 0.80 on average. In addition to F1-score which considers both the precision and recall, the overall accuracy of activity classification is shown in Figure 7. By modeling the context and using it to narrow down the search space of activity recognition, the average accuracy increases from 73% to 82% compared to using a single classifier for activity recognition, although it is fed with both motion and BLE statistical features. It is without doubt we can claim context is advantageous to this application domain. Please note that the activities studied in this paper are mostly complicated activities of daily living where detecting all of them with merely motion sensors is very challenging and nearly impossible [4]. Examples of such activities in this study are attending class, meeting, relaxing, working as well as cooking, shopping, cleaning, and getting ready. This is a significant improvement regarding the list of activities to be detected compared to the existing works in the field of activity recognition with wearable sensors. Moreover, it should be mentioned that there is no similar study and/or dataset in the literature that contains the BLE data from all the devices in the environment for detecting such high-level ADLs with a single wrist-worn sensor to compare with our method. In this study we did not include short-term activities such as toileting or hygiene, although these activities of daily living are very important in healthcare monitoring. It would be an important future direction to expand this work by collecting more data with a more exhaustive set of labeled activities. Since most of these activities happen in a certain location, the BLE would be helpful to narrow down the search space of activities so that motion sensors can easily differentiate them without confusing with other similar activities.

## 4.2 Context Learning Performance

In Section 4.1 we saw that using context for explicitly narrowing the search space achieves a higher accuracy than training a single model for the entire set of activities. In this section, we assess different components of our context modeling system and their impact on the performance of detecting the ADLs. It should be mentioned again that the studies that used BLE for complex ADL recognition all rely on setting up known

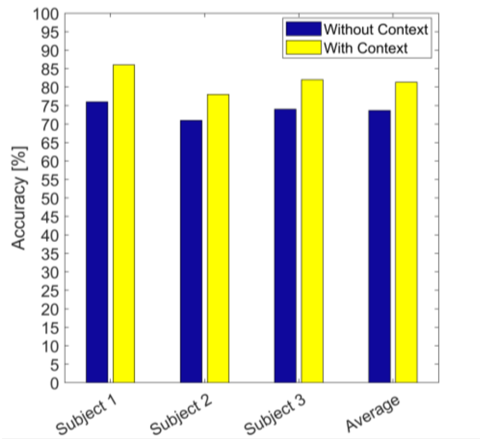


**Figure.7: Comparison of overall accuracy between the general activity classifier (without context) and context-specific classifier**

BLE beacons in specific locations, which limits their scalability and convenience [4]. There is no study and/or dataset in the literature that contains BLE readings from all the devices without need to setting up specific hardware. As there are few related works whose methods can be applied here for a direct comparison, and also to demonstrate the importance of each component of our context training system, we compare our methods with respect to several more basic methods. In our approach, we build a set of patterns for each activity as described in Section 3.2.2. We refer to this method as Accumulative Hierarchical Agglomerative Clustering (AHAC), formally expressed in Algorithm 1. We relate patterns with activities probabilistically, i.e., the action-context probability (ACP), and use the distribution of these probabilities to infer context and narrow the set of possible activities as shown in Algorithms 2 and 3.

Therefore, we compare our methods to more basic ones based on the two aforementioned components of pattern creation and pattern usage to show the importance of each component of our framework. One basic approach is to use all singular beacons as patterns if they exceed the minimum coverage threshold in Equation 2. We refer to these as "Single-Beacon Patterns." In fact, in this approach we ignore the AHAC module and assume each single beacon is associated with a context. Another basic approach is to ignore probabilistic association between the context and activities and use the patterns such that if a single pattern is discovered from an activity's records, then we consider that activity as a candidate for recognition in that context. This is similar to building a look up table that assumes an activity $a$ is probable in context $p$ if there is even one observation of $a$ at $p$. We refer to this as "Basic Pattern Usage." One other approach uses a single classifier where each beacon's presence is encoded as a binary feature in addition to the same set of IMU and BLE statistical features used in all the other classifiers. We also show results when patterns are extracted using the typical hierarchical agglomerative clustering (HAC) instead of our proposed modification (AHAC).

In order to account for probable noise in the detected nearables, mimic some real-world scenarios, and better evaluate our proposed method in response to challenging cases where BLE beacons can change drastically over time, we add or remove some beacons to/from the BLE records. To summarize these real-world scenarios, we simulate 1) a personally owned device by adding a specific beacon to 80% of all records, 2) beacons owned by friends or coworkers who span multiple contexts by adding specific beacons to 50% of instances of activities that may happen in the same context such as exercising and attending class, and 3) removal of some of the highly consistent beacons. We then compare our method with those described in the previous paragraph as shown in Figure 8.

According to Figure 8, our approach to pattern usage consistently outperforms the other approaches to context detection. Clearly, Basic Pattern Usage, in which only a single context pattern must be satisfied for an activity to be considered a possible outcome, performs most poorly as it is very susceptible to outliers. The mobile devices, such as subject's personal device causes different activities that do not naturally happen

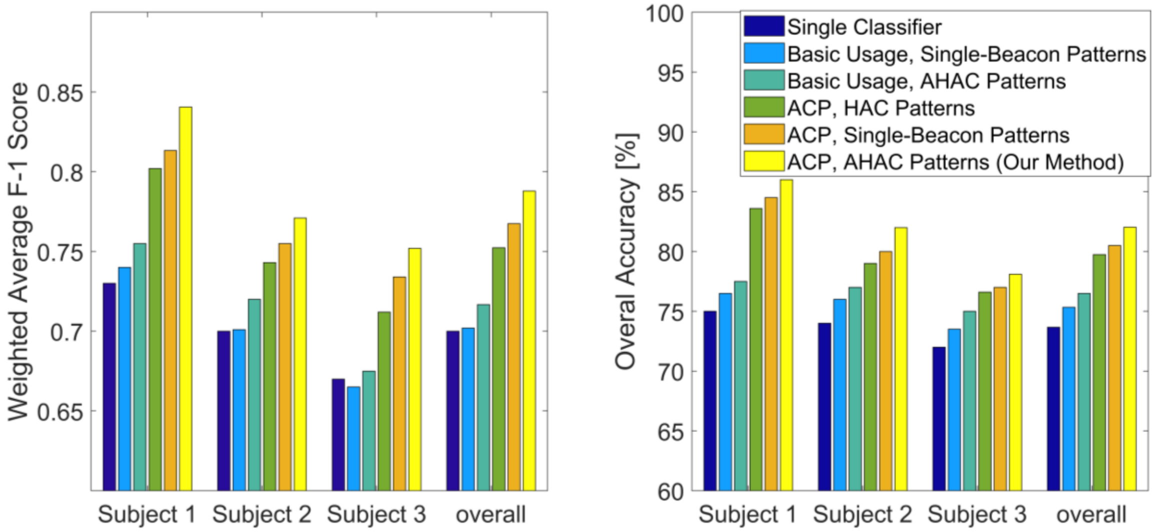


**Figure 8. Overall performance of different context detection approaches on noisy BLE data**

in the same environment to be considered probable, meaning that the model will perform little better than a single classifier using IMU and BLE statistics as input features. The difference in accuracy and F-1 score, for approaches that use ACP but different pattern sets are due to the more nuanced problems AHAC pattern extraction is designed to address. Patterns developed with AHAC can associate the mobile beacons with static beacons (or other consistent mobile beacons) from a particular context, thereby reducing the adverse effects that mobile beacons have on the distribution of ACPs when said beacon is also present in other contexts. However, AHAC will produce multi-beacon patterns composed of beacons that are frequently co-present. From these multi-beacon patterns, we get a more defined estimate of the set of feasible activities because these multi-beacon patterns will be more selective than their individual components. Even when important static beacons are not present, the combinations of multi-beacon patterns often still provide enough information to narrow the search space.

Using the single classifier, for subject 1, not a single instance of 'cooking' is recognized. Here, there was likely much reliance on specific beacons such that features from motion signals were entirely ignored. Moreover, for the single classifier, 'class' and 'working' are commonly confused with each other. We also observed that numerous instances of 'working' were misclassified as 'walking'. This is a surprising result due to the sedentary nature of working. Upon close inspection, we found that the subject walked enough within the building they work in to have a meaningful effect on the context models. This means that some beacons from their typical 'working' contexts also contain beacons considered important to walking. When we remove some highly consistent beacons from activities labeled 'working', some of those central to a context for walking still remain, causing the wrong context model to be chosen. However, when we use AHAC to extract patterns, that beacon which has a significant connection to walking forms multi-beacon patterns with beacons consistent with the 'working' activity. This more often causes the model to select the classifier trained on 'working' or both 'working' and 'walking'. In any case, the weighted-average F1-score of AHAC compared to Single-Beacon patterns is 0.78 compared to 0.70, suggesting AHAC has a substantial benefit.

To generalize these results, we see that noisier context data necessitates a clever usage of patterns. Our ACP estimation approach is effective at ignoring much of this noise by focusing on the overall set of patterns and how they relate to individual outcomes in the application domain. The means by which patterns are extracted is also important but plays a lesser role than their exact usage; we note that this is highly dependent on the data set. To quantify this, Figure 8 shows that Basic Pattern Usage has an average weighted F1-score of 0.70 and accuracy of 73% on average across all subjects, while applying ACP to those same patterns results in an increase to 0.76 F-1 score and 80% accuracy; this further increases to 0.78 and 82% when AHAC is used for pattern extraction, suggesting ACP to be the more influential part of our approach.

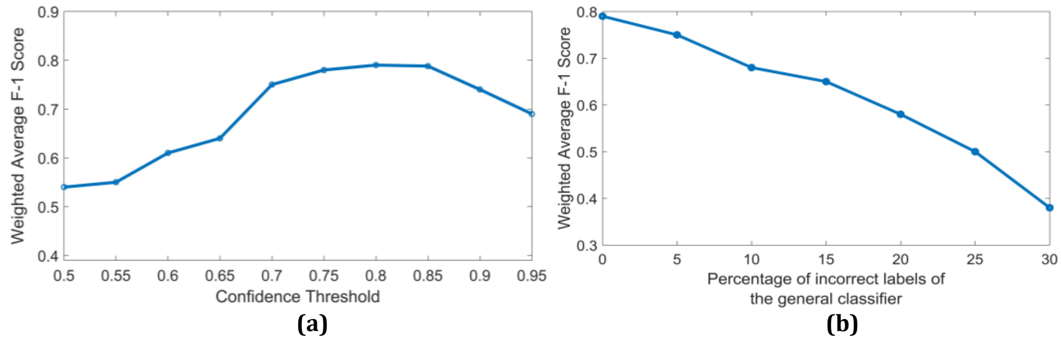


**Figure 9. (a) The effect of threshold for choosing most confident samples for context learning on the performance of the system; (b) The impact of general classifier's accuracy on the performance of the system**

It should be noted that the use of BLE devices for context identification in the literature is limited to deploying pre-known BLE beacons in specific places for localization purposes. First, this approach is burdensome for users as they need to deploy the hardware in their living environment. Second, extra infrastructure is needed to integrate the beacons. Third, it is limited to localization in specific places such as user's home. On the other hand, our proposed method leverages all the freely available BLE devices in user's locale, which could be very noisy information, but can account for both the location and the people and/or devices around the user. For example, when a user attends a class, the context may be identified as other students in the same class rather than the physical location of the classroom. Since this study is one of the first attempts towards modeling the context from freely available BLE devices broadcasted by any device around a user, we could not provide quantitative comparison with other context identification models. However, this study shows the feasibility of modeling context based on freely available BLE devices, which can enable new paradigms in detecting complex activities of daily living with wearable sensors.

## 4.3 Impact of Activity Labeling on Context Learning

As explained in Section 3, our proposed framework does not require to use any direct label of the context for the purpose of context learning. This is a realistic assumption since it is not feasible to get such labels from users about the details of their surrounding context all the time. Instead our algorithm leverages the labels of the activities to learn the context in order to narrow down the search space. To achieve this goal, in Section 3.3 we described how the system can use the most confident labels generated by the general motion-based classifier. In this section, in Figure 9, we show how the accuracy of that general classifier can impact the overall performance of context learning. Moreover, we demonstrate the improvement that can be achieved by getting the correct activity labels from the user in Figure 10.

Figure 9-a shows that varying the threshold, which is used to select the most confident decisions from the general classifier for context learning (see Section 3.3), affects the overall accuracy of the model. The Y axis in this figure is the weighted F-1 score averaged over all the subjects. As the figure shows, very low threshold reduces the accuracy because the system ends up using samples that can be misclassified by the general classifier, which results in learning inaccurate context. Similarly, higher threshold causes choosing very small number of biased samples which results non-sufficient context learning. Based on this figure, we chose threshold of 0.8 as a reasonable number to keep the confident samples for context training given our collected dataset. Figure 9-b shows that as the error of the general classifier increases, the accuracy of the context-based activity recognition decreases because the system ends up associating context to incorrect activities. In this figure, the X axis shows the percentage of labels to which we assigned a randomly incorrect label to mimic the error in general classifier, and Y axis is the F-1 score averaged over all the subjects. Therefore, more accurate motion-based classifier can help to better context learning, which in turn helps to more accurate recognition of activities.

A possible limitation of this approach is that when BLE devices are added or removed to/from the subject's surroundings, the system should relearn the changed context. This is a challenge since the BLE devices around a user can dynamically change over time as the person visits new places and people. Therefore, this

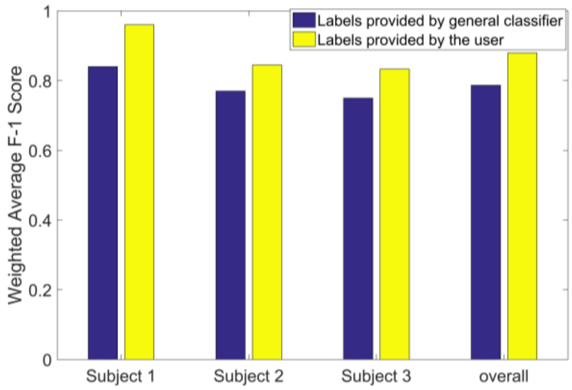


**Figure 10. The impact of getting label from the user on the overall performance of the system**

system must always be updated via an online learning approach. It could also be combined with a supervised online learning in which the annotations can be queried from the user for more accurate training. The need for user intervention, however, could be minimized by detecting consistently observed BLE patterns and then query the user for annotation. Hence, most part of this online learning can be automated without the need for extensive user interference. This inspires an excellent future work in this area.

Figure 10 shows the performance when the labels of activities for context learning are provided by the user, which can be interpreted as the upper bound of the accuracy. As this figure shows, the overall accuracy of the system can be increased to 0.88 when the reliable labels are provided, which is 0.08 more than the accuracy when the labels are supplied by the general motion-based classifier.

# 5 CONCLUSION

We explored data-driven context detection on passively observable devices in the user's daily environment. Our approach relies on knowledge within a chosen application domain for training the context model. In this work, the application is recognition of ADLs, and context is built from passively-sensed BLE devices. Our approach builds an offline context model by leveraging the consistency at which individual BLE devices are observed near the user when a particular action of the user, i.e., activity, is being performed. This consistency is used to generate a set of context patterns which are then mapped probabilistically back to the actions to aid performance via a reduction in the set of probable outcomes. Using this method, we achieved an average F-1 score of 0.80 for detecting ADLs from data collected in real-world as the users went about their daily lives. This method offers a significant improvement over a single classifier approach for the same input features and has been shown to be more robust to realistic types of noise than similar methods. Important advantages of our proposed method compared to the state-of-the-art environmental context detection approaches are that it does not require one to deploy additional hardware and infrastructure, reduces users' burden, and can potentially infer the context related to both physical location and people and/or devices around a user. The proposed technique enhances the capabilities of wearable sensors by helping them understand their working environment. It also enables detecting complex ADLs with a single wrist-worn motion sensor.

# ACKNOWLEDGMENT

This work was supported in part by the National Institute of Health, under grant 1R01EB028106. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding organizations

## REFERENCES

[1] A. Akbari and R. Jafari, “Transferring activity recognition models for new wearable sensors with deep generative domain adaptation,” in *Proceedings of the 18th International Conference on Information Processing in Sensor Networks*, 2019, pp. 85–96.

[2] B. Ibrahim, J. McMurray, and R. Jafari, “A Wrist-Worn Strap with an Array of Electrodes for

Robust Physiological Sensing," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2018, pp. 4313–4317.
[3] M. Esfahani, M. Iman, and M. A. Nussbaum, "Classifying Diverse Physical Activities Using Smart Garments," *Sensors*, vol. 19, no. 14, p. 3133, 2019.
[4] P. Bharti, D. De, S. Chellappan, and S. K. Das, "HuMAn: Complex Activity Recognition with Multi-modal Multi-positional Body Sensing," *IEEE Trans. Mob. Comput.*, vol. 18, no. 4, pp. 857–870, 2019.
[5] R. Solis, A. Pakbin, A. Akbari, B. J. Mortazavi, and R. Jafari, "A human-centered wearable sensing platform with intelligent automated data annotation capabilities," in *Proceedings of the International Conference on Internet of Things Design and Implementation*, 2019, pp. 255–260.
[6] G. Sprint, D. Cook, R. Fritz, and M. Schmitter-Edgecombe, "Detecting health and behavior change by analyzing smart home sensor data," in *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*, 2016, pp. 1–3.
[7] P. Bharti, A. Panwar, G. Gopalakrishna, and S. Chellappan, "Watch-dog: detecting self-harming activities from wrist worn accelerometers," *IEEE J. Biomed. Heal. informatics*, vol. 22, no. 3, pp. 686–696, 2017.
[8] L. Wang, T. Gu, X. Tao, H. Chen, and J. Lu, "Recognizing multi-user activities using wearable sensors in a smart home," *Pervasive Mob. Comput.*, vol. 7, no. 3, pp. 287–298, 2011.
[9] A. Bulling, U. Blanke, and B. Schiele, "A tutorial on human activity recognition using body-worn inertial sensors," *ACM Comput. Surv.*, vol. 46, no. 3, p. 33, 2014.
[10] A. Akbari, J. Wu, R. Grimsley, and R. Jafari, "Hierarchical signal segmentation and classification for accurate activity recognition," in *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, 2018, pp. 1596–1605.
[11] A. Akbari and R. Jafari, "An Autoencoder-based Approach for Recognizing Null Class in Activities of Daily Living In-the-wild via Wearable Motion Sensors," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 3392–3396.
[12] J. C. Krumm, E. J. Horvitz, and R. Hariharan, "Integration of location logs, GPS signals, and spatial resources for identifying user activities, goals, and context." Google Patents, 2011.
[13] E. R. Sykes, S. Pentland, and S. Nardi, "Context-aware mobile apps using iBeacons: towards smarter interactions," in *Proceedings of the 25th Annual International Conference on Computer Science and Software Engineering*, 2015, pp. 120–129.
[14] G. W. Musumba and H. O. Nyongesa, "Context awareness in mobile computing: A review," *Int. J. Mach. Learn. Appl.*, vol. 2, no. 1, p. 5, 2013.
[15] M. Weiser, "The computer for the 21st century," *IEEE pervasive Comput.*, vol. 1, no. 1, pp. 19–25, 2002.
[16] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *IEEE Commun. Surv. tutorials*, vol. 16, no. 1, pp. 414–454, 2014.
[17] L. Herranz, S. Jiang, and R. Xu, "Modeling restaurant context for food recognition," *IEEE Trans. Multimed.*, vol. 19, no. 2, pp. 430–440, 2016.
[18] R. Reichle *et al.*, "A comprehensive context modeling framework for pervasive computing systems," in *IFIP International Conference on Distributed Applications and Interoperable Systems*, 2008, pp. 281–295.
[19] D. De, P. Bharti, S. K. Das, and S. Chellappan, "Multimodal wearable sensing for fine-grained activity recognition in healthcare," *IEEE Internet Comput.*, vol. 19, no. 5, pp. 26–35, 2015.
[20] A. Filippoupolitis, W. Oliff, B. Takand, and G. Loukas, "Location-enhanced activity recognition in indoor environments using off the shelf smart watch technology and BLE beacons," *Sensors*, vol. 17, no. 6, p. 1230, 2017.
[21] L. Aalto, N. Göthlin, J. Korhonen, and T. Ojala, "Bluetooth and WAP push based location-aware mobile advertising system," in *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, 2004, pp. 49–58.
[22] J. Zheng and L. M. Ni, "An unsupervised learning approach to social circles detection in ego bluetooth proximity network," in *Proceedings of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*, 2013, pp. 721–724.
[23] M. Han, J. Bang, C. Nugent, S. McClean, and S. Lee, "A lightweight hierarchical activity

recognition framework using smartphone sensors," *Sensors*, vol. 14, no. 9, pp. 16181–16195, 2014.
[24] A. Subramanya, A. Raj, J. A. Bilmes, and D. Fox, "Recognizing activities and spatial context using wearable sensors," *arXiv Prepr. arXiv1206.6869*, 2012.
[25] J. Chon and H. Cha, "Lifemap: A smartphone-based context provider for location-based services," *IEEE Pervasive Comput.*, vol. 10, no. 2, pp. 58–67, 2011.
[26] N. Bulusu, J. Heidemann, D. Estrin, and others, "GPS-less low-cost outdoor localization for very small devices," *IEEE Pers. Commun.*, vol. 7, no. 5, pp. 28–34, 2000.
[27] T.-B. Nguyen, T. Nguyen, W. Luo, S. Venkatesh, and D. Phung, "Unsupervised inference of significant locations from wifi data for understanding human dynamics," in *Proceedings of the 13th International Conference on Mobile and Ubiquitous Multimedia*, 2014, pp. 232–235.
[28] A. Alvarez-Alvarez, J. M. Alonso, and G. Trivino, "Human activity recognition in indoor environments by means of fusing information extracted from intensity of WiFi signal and accelerations," *Inf. Sci. (Ny).*, vol. 233, pp. 162–182, 2013.
[29] H. Hong, C. Luo, and M. C. Chan, "Socialprobe: Understanding social interaction through passive wifi monitoring," in *Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*, 2016, pp. 94–103.
[30] Z. Chen, Y. Chen, S. Wang, J. Liu, X. Gao, and A. T. Campbell, "Inferring social contextual behavior from bluetooth traces," in *Proceedings of the 2013 ACM conference on Pervasive and ubiquitous computing adjunct publication*, 2013, pp. 267–270.
[31] T.-F. Wu, C.-J. Lin, and R. C. Weng, "Probability estimates for multi-class classification by pairwise coupling," *J. Mach. Learn. Res.*, vol. 5, no. Aug, pp. 975–1005, 2004.
[32] J. Wu, L. Sun, and R. Jafari, "A wearable system for recognizing American sign language in real-time using IMU and surface EMG sensors," *IEEE J. Biomed. Heal. informatics*, vol. 20, no. 5, pp. 1281–1290, 2016.
[33] J. Wu and R. Jafari, "Orientation independent activity/gesture recognition using wearable motion sensors," *IEEE Internet Things J.*, 2018.