

Sleep Well: A Sound Sleep Monitoring Framework for Community Scaling

H M Sajjad Hossain, Nirmalya Roy, MD Abdullah Al Hafiz Khan
Department of Information Systems, University of Maryland Baltimore County
riaj.sajjad@umbc.edu, nroy@umbc.edu, mdkhan1@umbc.edu

Abstract—Following healthy lifestyle is a key for active living. Regular exercise, controlled diet and sound sleep play an invisible role on the wellbeing and independent living of the people. Sleep being the most durative activities of daily living (ADL) has a major synergistic influence on people’s mental, physical and cognitive health. Understanding the sleep behavior longitudinally and its underpinning clausal relationships with physiological signals and contexts (such as eye or body movement etc.) horizontally responsible for a sound or disruptive sleep pattern help provide meaningful information for promoting healthy lifestyle and designing appropriate intervention strategy. In this paper we propose to detect the microscopic states of the *sleep* which fundamentally constitute the components of a good or bad sleeping behavior and help shape the formative assessment of sleep quality. We initially investigate several classification techniques to identify and correlate the relationship of microscopic sleep states with the overall sleep behavior. Subsequently we propose an online algorithm based on change point detection to better process and classify the microscopic sleep states and then test a lightweight version of this algorithm for real time sleep monitoring activity recognition and assessment at scale. For a larger deployment of our proposed model across a community of individuals we propose an active learning based methodology by reducing the effort of ground truth data collection. We evaluate the performance of our proposed algorithms on real data traces, and demonstrate the efficacy of our models for detecting and assessing fine-grained sleep states beyond an individual.

I. INTRODUCTION

Sleeping disorder plays as an indicator for many medical conditions such as sleep apnea, COPD (Chronic Obstructive Pulmonary Disease), Chronic renal diseases and various other medical conditions [1]. Sleep quality is greatly correlated to exhaustion, discomfort, depression and lack of concentration during the day. The quality of sleep can affect the intuitive symptoms due to the underlying disease. Clinical studies have suggested that if the sleep quality is improved the underlying symptoms of a patient might improve too. Early clinical sleep studies used to be based on Polysomnography (PSG) [2]. By monitoring patients biophysiological signals clinicians are able to have good understanding of the sleep disorder the patient is experiencing. As there has been a significant enhancement in sensor technology, nowadays the researchers are also using sensor technologies rigorously in sleep quality measurement. Various commercial products like Fitbit [3], Actiwatch [4], BASIS watch [5] etc are also available in the market to monitor the quality of the sleep. But the measurement method for most of these devices is focused on calculating the duration of sleep which gives a very good insight on someone’s sleep hygiene. Getting information on a more detailed sleep cycle is not possible with these devices. Only BASIS B1 band [5] classifies

sleep stages (REM, light sleep, deep sleep) and provides the means to identify patterns and triggers which are causing sleep disturbances.

Existing sleep monitoring researches use supervised learning algorithms where they collect and label a set of training data with some pre-defined classes which the system aims to detect. The more labeled training dataset leads to a better classification. As collecting ground truth is extremely difficult without violating privacy concerns while sleeping, it is a daunting task for test subjects and human annotators to label the sleep states properly. Also identifying and classifying only pre-defined sleep stages is not enough for medical diagnosis sometimes. For example, current state of the literature does not address if the patient is suffering from nightmare disorder, muscle contraction etc frequently. In this paper we propose a novel sleep monitoring model to classify previously unseen various sleep states which will further improve patients sleep hygiene by being able to pinpoint the causes of sleep disturbances. We train our model using supervised and unsupervised learning algorithm and identify basic sleep states - Rapid Eye Movement (REM), non-REM (NREM) sleep, awake, movement, getting up from bed, getting up and sitting.

Wearable devices are now becoming a huge trend and they are evolving rapidly by incorporating various sensors as attested by the new release of smart watches such as Google Android Wear [6], Intel Basis B1 band [5] etc. The world is moving more towards wearable technology which is being used in many health care applications [7]. We have used two different wearable devices - EZ430-Chronos [8] and wActiSleep BT [4] by putting them on the waist. After collecting the data we apply a variant of gradient descent to build our classification model. Then we apply importance weighted active learning to label the uncertain data points and also incorporate previously unseen sleep states. Active learning helps the annotation effort greatly and improves the performance of classification model. In order to discover abrupt changes on the data streams, increase the classification accuracy, remove noises and provide greater support for informativeness in active learning we propose an online change point detection algorithm. Also we show results of our proposed algorithm using publicly available benchmark dataset [9] which provides sleep phases determined by clinical polysomnography where the data was collected using wrist worn device. The main contributions of the paper are summarized below:

- We investigate several classification approaches and propose a gradient descent classification model for recognizing the underlying microscopic context states associated with the sleep disorders.

- We introduce an online change point detection based classification approach to help detect any abrupt changes on streaming dataset for better microscopic sleep state classification and data noise and uncertainty reduction.
- We develop active learning based sleep monitoring models which help reduce the extensive data annotation effort and ground truth data collection from the user personal space and help scale the model among a community of individuals.
- We evaluate our model based on two real-world datasets, one with polysomnography result along with the wrist worn accelerometer sensor values from 42 subjects [9], other with general labeled data collected from 17 test subjects.

II. RELATED WORKS

In medical studies Polysomnography (PSG) is the major sleep study to diagnose a patient's sleeping quality [2]. Polysomnography records biophysiological changes that occur during the sleep. Apart from PSG, some other sleep studies focused on techniques, such as Multiple Sleep Latency Test (MSLT) and Maintenance of Wakefulness Test (MWT) [10]. These diagnoses are cumbersome and need a lot of prior setup, for example, in case of PSG it requires 12 channels requiring almost 22 wire attachments to a patient. Obviously this imposes a great level of discomfort to the patients and its users. The authors of [11] developed a wearable neck cuff system for monitoring physiological signals in real-time. A sleep monitoring model using image analysis has been proposed in [12], but it has proved inefficient in case of low light condition at night. [13] used near-infrared cameras to overcome this challenge but the images still created non-uniformity. A novel sleep monitoring framework- LullaBy to capture and monitor the sleeping environment using microphone, light sensor and motion sensor has been proposed in [14].

Pressure bed sensors have been used to supervise the postures and movements of the users in sleep [15]. Though these methods are unobtrusive and do not create discomfort to the users, but still it has not been streamlined due to its cost and deployment issues. [16] used fine-grained body positions from accelerometer data using WISP tags attached to the sides of a bed. A novel framework for pressure image analysis to monitor sleep postures including a set of geometrical features for sleep posture characterization and three sparse classifiers for posture recognition has been proposed in [17].

Sleep related research are gaining attention due to the recent proliferation of low-cost easy-to-deploy technologies based on mobile and ambient sensors and its large penetration in the market. Commercial wearbale devices, such as Fitbit [3], Zeo [18], Actigraph [4], Jawbone, Sleep Tracker etc have been used extensively these days for monitoring sleep and activities of daily living (ADLs). iSleep [19] uses the built in microphone sensor of smartphone to detect the events which are closely related to sleep quality like body movements, coughing, snoring etc. The authors of [20] used the accelerometer sensor of the smartphone to track the sleep duration and user movement patterns. [21] proposed a passive approach to track some stationary features, such as user silence, ambient

light, phone usage and charging etc for monitoring sleep habits and developed a mobile application BeWell [22] for unified health monitoring. [23] used the daily context information of an user to define the sleep quality. Sleep Hunter [24] used the accelerometer and microphone sensors of the smartphone a fine-grained detection of sleep stage transition for sleep quality monitoring and intelligent wake-up call. [25] proposed to use change-point segmentation on PSG data to differentiate macrostructural organization of sleep. A point process based novel model for the assessment of heart rate variability and respiratory sinus arrhythmia based on PSG data has been proposed in [26].

In this paper we take a radically different approach and look into the fundamental problem of scaling the sleep monitoring models beyond a specific individual. To realize this first we analyze the microscopic physiological contexts and psychological clauses behind a sound or bad sleep. We investigate traditional classification algorithms to successfully detect those events and propose a novel online change point detection based method for enhancing the classification accuracy and eventually help guide the design of a community scaling model using active learning.

III. OVERVIEW OF SLEEP WELL FRAMEWORK

Sleep is not just a dormant part of our lives, we remain very active and pass through several important stages of sleep. Interference or disturbance in these states can cause impatience, drowsiness and lack of concentration during the regular activities of daily living. Therefore for maintaining a good sleep hygiene we have to sleep a certain amount of time in each of those sleep states. There are two main types of sleep states:

- Non-Rapid Eye Movement (NREM) (also known as *quiet sleep*). NREM consists of three states (stage-1, stage-2, stage-3).
- Rapid Eye Movement (REM) (also known as *active sleep*).

A complete and healthy cycle of sleep consists of a progression from states 1 to 3 before reaching REM state, then the cycle starts over again. If the REM sleep is disrupted and the person wakes up, the person's circadian cycle is disrupted. In order to complete the cycle the person will move to REM state directly next time. Thus it is very important to sleep a good amount of time each day and maintain a good sleep cycle. REM sleep is considered as active sleep because in this state people dream. If a person is having a nightmare disorder too often it is possible that he/she is having problems to complete the sleep cycle. In this paper, we first focus on properly classifying the sleep cycle into these finer states. We also propose to integrate some other broader intermediate sleep states such as *movement*, *getting up from bed* and *getting up and sitting*. These other states would help to identify the casual and formal causes of sleep disturbance and sleep latency and provide meaningful insights on designing scalable sleep monitoring technologies and automated assessment methodologies.

A. Sleep Well Architecture

Our proposed sleep well framework consists of the following logical components.

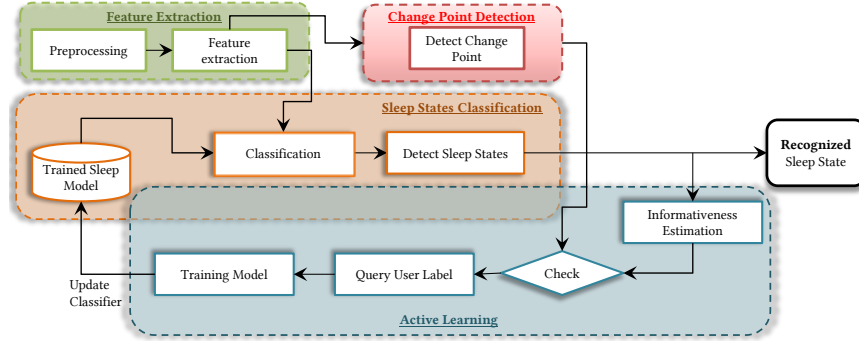


Fig. 1. An architectural overview of Sleep Well Framework

- Feature Extraction:** After collecting raw sensor data, this component preprocesses and extracts the low level signal features as shown in Table I from the processed raw sensor data. (Details in IV-A).
- Change Point Detection:** After extracting features and analyzing the sleep data, we noticed that change point occurs in sleep transitions (transition from one stage to another, for example unconscious movement during sleeping, waking up, being restless in bed etc.). The importance of these change points has proven to be very effective as it help removing noises in the data and detect the exact point of the sleep transition. For example, when a person gets up from the bed and starts walking, the accelerometer readings other than sleep classes become noisy. Therefore by identifying change points we can partition the data and have more fine-grained information for easing the training effort. (Details in IV-B).
- Classification:** At this stage we train our model using the features from processed raw sensor data and build up our classification model to recognize the several intermediate sleep states. We investigated an online gradient descent [27] as our classification algorithm. This is different from traditional gradient descent by dealing with importance weights to collaborate with active learning and learning reductions. To calculate the average loss during the classification process we propose to use squared loss function (Details in IV-C).
- Active Learning:** After feeding the test data into our classification model and getting the prediction, active learning helps to calculate the informativeness of each data points. If any data point falls within an uncertain space and while predicting it is found to be the most informative, then if the actual label of the point is provided it would have more significant impact on the classification model. This component then initiate prompt for “*query user label*” and get the ground truth from the user. Subsequently the labels are then used for re-training and updating the model. This component helps to ensure better classification accuracy with minimal user feedback. This also helps to scale the sleep monitoring model across multiple individuals. The input from change point detection method strengthens the active learning query selection

by asking the user to label the appropriate sleep state transitioning step. (Details in IV-D).

IV. SLEEP WELL FRAMEWORK DESIGN

In this section we describe in details the design of our Sleep Well framework. We first discuss about several micro-states of sleep and feature extraction process. Next we discuss an online change point detection algorithm to have a better handle on the microscopic sleep state classification problem.

A. Sleep Event Detection and Feature Extraction

We extract low-level features using each of the three components of the triaxial accelerometer signal to capture the aspects of movements while sleeping. We use both time and frequency domain features in our framework. As the user is not physically active while sleeping, very few number of movements are involved, so we choose a lower sampling frequency. We extract features from data using windows of 60 samples, corresponding to 1 second of accelerometer data. From each window we calculate the features mentioned in Table I. Time domain features help to differentiate dynamic to static movements. The frequency domain features help identifying patterns within acceleration data, which aids in discriminating discrete movements and their intensities.

1) *Feature Selection:* Our model is primarily focused on community scaling, so fewer features will scale the model computationally effective if we can achieve similar accuracy with more features. We select the subset of features, best fit for our model by applying Restricted Forward feature Selection (RFS) algorithm. It was performed in two steps. First we applied Forward feature Selection (FS) algorithm which ranks the features in decreasing order of their accuracy. The FS algorithm iterates through the feature space and measures the Leave-One-Out-Cross-Validation (LOOCV) error for each component in the feature space $\{f_1, f_2, f_3, \dots, f_N\}$. In case of traditional FS, after the first iteration, FS calculates the best individual feature f_i . In the next iterations, FS finds the best subset consisting of two components, f_i and one other from remaining $N - 1$ features. In the following iterations, FS ranks more features and evaluates the subset accordingly, so that after N iterations, the winner is the overall best feature set in these N iterations. In the second step we invoke the RFS to restrict the number of features to rank at each iteration. After the first iteration we consider only the first $N/2$ ranked features for the following iteration. After adding another feature to

TABLE I. FEATURES USED FOR SLEEP MICRO-STATES CLASSIFICATION

	Name	Definition
Time Domain Features	Mean	AVG ($\sum x_i$), AVG ($\sum y_i$), AVG ($\sum z_i$)
	Mean-magnitude	AVG $\sqrt{x_i^2 + y_i^2 + z_i^2}$
	Magnitude-Mean	$\sqrt{\bar{x}^2 + \bar{y}^2 + \bar{z}^2}$
	Variance	VAR ($\sum x_i$), VAR ($\sum y_i$), VAR ($\sum z_i$)
	Co-Variance (Two-axis correlation)	$cov(xy)$; $cov(yz)$; $cov(xz)$
	Standard Deviation	$\sigma_x = \sqrt{\frac{\sum(x-\bar{x})^2}{n-1}}$; $\sigma_y = \sqrt{\frac{\sum(y-\bar{y})^2}{n-1}}$; $\sigma_z = \sqrt{\frac{\sum(z-\bar{z})^2}{n-1}}$
Frequency Domain Features	FFT-Magnitude	$m_j^{(x)} = a_j + b_j i $; $m_j^{(y)} = a_j + b_j i $; $m_j^{(z)} = a_j + b_j i $
	FFT-Energy	$\frac{\sum_{j=1}^N (m_j^2)}{N}$ for x, y, z respectively

the winner of the first iteration at the second iteration, we consider the first $N/3$ components of the remaining ranks. RFS repeats this process until it finds the best m feature sets. The difference between conventional FS and RFS is that RFS considers only a part of the remaining ranked features, whereas FS considers all the features. Out of 8 features, the feature selection algorithm chose 4 features (FFT-Magnitude, FFT-Energy, Mean-Magnitude and Co-Variance) which help to attain classification accuracy closer to using all the 8 features.

B. Change Point Detection

Change point detection helps find the abrupt variations in the sleep data stream. While some change points provide meaningful insights and some not, our motivation in this work is to find the sleep transitions by calculating the change points (abrupt signal changes) and distinguish between the important and unimportant changes. This is not only helpful to detect the sleep-related events appropriately but also help remove noisy data points from the dataset. We develop a Bayesian online changepoint detection [28] based algorithm for finer sleep related event identification and online data noise reduction.

We first partition the entire sleep dataset in different regions based on a run length [28]. Let, $x_{1:N} = \{x_1, x_2, x_3, \dots, x_N\}^T$ denote the N data points observed over time T which is divided into non-overlapping partitions. Consider if we find K change points then let the data set of partitioned data be $\{\rho_1, \rho_2, \rho_3, \dots, \rho_k\}$ at time indices $\{t_1, t_2, t_3, \dots, t_k\}$ where by definition $t_0 = 0$ and $t_{k+1} = N$. The discrete probability distribution over a time interval t_i to t_j is denoted by $g(t_i - t_j)$. Each partition ρ_t denotes a segment of the data at time t . The length of the each partition or time since the last change point occurred, is defined as ‘‘run length’’, r . The run length goes back to 0 if change point occurs, otherwise it increases by 1 as follows.

$$r_n = \begin{cases} 0, & \text{if changepoint occurs at } (n-1) \\ r_{n-1} + 1, & \text{otherwise} \end{cases}$$

The conditional probability that a change point occurs on time t_k after the last change point at time t_{k-1} is

$$P(t_k | t_{k-1}) = g(t_k - t_{k-1}), \text{ where } 0 < k - 1 < n \quad (1)$$

We assume that the predictive distribution of a change point at any time instant t only depends on the recent data. So the change points are assumed to follow markov process. Thus the prior probability of a change point at a time instant t_k is dependent on the probability distribution of the observed data

over the time interval and the preceding change point.

$$P(t_k) = \sum_{i=0}^{k-1} g(t_k - t_i) P(t_{k-1}) \quad (2)$$

The change point detection algorithm finds the number of change points and their position by calculating the posterior probability $P(r_n | x_{1:n})$ and integrating it with the predictive distribution $P(x_{n+1} | x_n)$. We do this by calculating the joint distribution of the current run length and observed data $P(r_n, x_{1:n})$.

$$\begin{aligned} P(r_n, x_{1:n}) &= \sum_{r_{n-1}} P(r_n, r_{n-1}, x_{1:n}) \\ &= \sum_{r_{n-1}} P(r_n, x_n | r_{n-1}, x_{1:n-1}) P(r_{n-1}, x_{1:n-1}) \\ &= \sum_{r_{n-1}} P(r_n | r_{n-1}) P(x_n | r_{n-1}, x_{n-r:n}) P(r_{n-1}, x_{1:n-1}) \quad (3) \end{aligned}$$

Where $P(r_n | r_{n-1})$ is the transition probability and $P(x_n | r_{n-1}, x_{n-r:n})$ is the data segment likelihood probability. We calculate the transition probability using equation

$$P(r_n | r_{n-1}) = \begin{cases} h(r_{n-1} + 1), & \text{if } r_n = 0 \\ 1 - h(r_{n-1} + 1), & \text{if } r_n = r_{n-1} + 1 \end{cases}$$

where $h(x) = g(x) / \sum_{i=x}^{\infty} g(i)$. We calculate the posterior probability using Bayes’s rule:

$$P(r_n | x_{1:n}) = \frac{P(r_n, x_{1:n})}{\sum_{i=0}^{n-1} P(r_i, x_{1:i})} \quad (4)$$

We calculate the posterior probability of the run length at that time index which corresponds to a new data sample. The pseudo code of this procedure is summarized in Algorithm 1.

Algorithm 1 Change Point Detection

- 1: Initialize: $P(r_0) = 1$
 - 2: **for** Each new data point x_n **do**
 - 3: Calculate the data segment likelihood probability, $P(x_n | r_{n-1}, x_{n-r:n})$
 - 4: Calculate the transition probability, $P(r_n | r_{n-1})$
 - 5: Calculate the joint distribution, $P(r_n, x_{1:n})$
 - 6: Find the posterior distribution on current run length, $P(r_n | x_{1:n})$
 - 7: Calculate the predictive distribution of x_n based on previous observation. $P(x_n | x_{n-1})$
 - 8: **end for**
-

C. Classification

We classify the sleep states using an online gradient descent method which leverages the importance weight on streaming data samples. To build up our classification model accurately we consider other sleep contexts such as body movements, but most of the sample data points resemble stationary states during the sleeping. Online gradient descent with importance weight aware updates [27] helps to overcome this limitation of data by assigning weight to classes with lesser data points. The key principle here is: *The assignment of importance weight h to a sample that make it appears like a regular example of h times in the dataset.* We assume C is our classification model and use squared loss function for examining the consistency of C . The goal of our classification model is to minimize the loss function which reflects better accuracy. After each iteration of gradient descent, C is not altered, rather it is improved by adding an estimator h in order to optimize the loss function. We assume y is the true label and p is prediction of our model, where $l(p, y)$ is the loss function as shown in Eqn. (5). At each step C is updated using Eqn. (6)

$$l(p, y) = \frac{1}{2}(y - p)^2 \quad (5)$$

$$C_{m+1} = C_m + h(x) \quad (6)$$

Let w be the vector of weights and the training set is a set of (x_i, y_i, h_i) , $i = 1, \dots, T$ where x_t is a vector of d features. For linearity we assume $p = w^T x$. Our goal is to assign w in such a way so that the model C converges to the optimized solution. Assigning weight to a data point (x, y) , h times in a row have a cumulative effect with scaling factor $k(h)$ as shown in Eqn. (7). This scaling factor is defined by Eqn. (8) where η is the learning rate [27]. At each iteration this weight is updated accordingly to the loss function l . Our proposed classification algorithm for finer non-stationary sleep states detection is shown in Algorithm 2.

$$w_{i+1} = w_i - k(h)x \quad (7)$$

$$k(h) = \frac{p - y}{x^T x} (1 - e^{-h\eta x^T x}) \quad (8)$$

Algorithm 2 Importance Weighted Sleep Classification

- 1: **Input:** Extracted feature vectors from raw data.
 - 2: **Output:** Predicted Sleep Class y .
Update the importance weight of the data points.
 - 3: **Initialize:** $\forall_i w_i \leftarrow 0$
 - 4: Get the feature vector for data point x_i
 - 5: Predict the class label y_i for all x_i
 - 6: Calculate the scaling factor $k(h)$
 - 7: **for** $i = 0$ to N **do**
 - 8: Calculate the weight w_i for each x_i
 - 9: **Update:** $w_i \leftarrow w_i - \frac{p - y}{x^T x} (1 - e^{-h\eta x^T x})x$
 - 10: **end for**
-

D. Active Learning based Community Scaling

Our goal in this paper is to scale the sleep monitoring model to a community of individuals. While a significant research has been done on sleep monitoring and assessment and intervention strategies, lack of novel scaling algorithms prohibit the deployment, large scale validation, and acceptance of these technologies for healthy lifestyle, smart health and independent living applications. In this section we investigate

how active learning based machine learning algorithms help build an informative model in presence of a minimal labeled datasets. We also depict how change point detection based time-series data analytics methodology help reduce the data uncertainty and guide to the selection of most informative query.

Active learning has been proved to be very effective by combining it with supervised learning when a large pool of unlabeled data is available. Though traditional passive learning takes the initiative to label the unlabeled data randomly, but most of the data points which are selected randomly does not ensure better classification. It is difficult to collect all of the sleep related ground truth information from the user though by using the accelerometer sensor it is possible to broadly monitor the user sleep behavior and the specific sleep duration. To collect more fine-grained details about the sleep we train our proposed gradient based classifier with the causes of sleep disruption (such as waking up from nightmare, muscle cramp etc). By applying active learning we propose to collect the labels of these informative data points so that our model can better classify the sleep stages and conditions and help scale this model in presence of minimal amount of ground truth. While applying active learning, one constraint is we have to assure that the whole labeling process doesn't become too intrusive. Crowd-sourcing can help us overcome this constraint by collecting a large amount of labeled data via arbitrary participants and provides aid in community scaling.

1) *Query selection:* In the following we briefly discuss the query selection approaches for active learning:

- **Query Synthesis:** The active learner asks the human annotator for "label membership" by using membership queries. In this approach the learner generates instances rather than sample from existing unlabeled set. But the problem with this approach is human annotator may have difficulty interpreting and labeling arbitrary instances.
- **Stream based selective sampling:** Each unlabeled instance is drawn at a time from the input source and the learner may decide instantly whether to query the instance or not. As we are using online classification algorithm and the data are processed in stream, we use this sampling strategy for our active learner.
- **Pool based sampling:** Evaluates and ranks the entire collection of unlabeled data before selecting the best query from a pool of instances.

2) *Sampling metrics:* Different sampling metrics such as least confident, margin sampling or maximum entropy based sampling are common in active learning algorithms. We propose to use the importance weighted active learning approach to build our community scaled sleep monitoring model [29]. To decide which points are most informative, we first calculate the utility measurements of unlabeled data points. Whether a data point x_t will be queried or not depends on the history of labels seen so far based on our change point detection, gradient based classification and the identity of the point. If a change point is detected at data point x_t at time index t_n , and the label of x_t is inconsistent with the label of current run r_n , we invoke active learning. A probability measure p_t is maintained

for each data point x_t . A coin flip, $Q_t \in \{0, 1\}$ with $E[Q_t] = p_t$ determines whether the data point will be queried or not. If the data point is queried based on the past history, then we update importance weight by $\frac{1}{p_t}$.

The active learning algorithm maintains an effective hypothesis space H_t through out the process. Initially, H_t contains all of the hypotheses from global space H . The expected loss of a hypothesis, $h \in H$ at time T is defined by Eqn. (9).

$$L_T(h) = \frac{1}{T} \sum_{t=1}^T \frac{Q_t}{p_t} l(h(x_t), y_t) \quad (9)$$

As it progresses, H_t becomes narrower by taking a subset, and ensuring that the factual loss of H_{t+1} is not much worse than the smallest loss, L_t^* in H_t .

$$H_{t+1} = \{h \in H_t : L_{t+1}(h) \leq L_t^*(h)\} \quad (10)$$

For each data point x_t , the active learning algorithm looks at the range of predictions and their losses by hypotheses in H_t and sets the sampling probability to the size of this range.

$$p_t = \max_{f, g \in H_t} \max_y l(f(x_t), y) - l(g(x_t), y) \quad (11)$$

If the range is too high than the rejection threshold then the hypotheses disagree greatly with each other. This certifies that the current prediction of x_t lies in the uncertain region. The active learning algorithm then queries for the label to settle the uncertainty. Our proposed active learning algorithm for largely reducing the micro-sleep states annotation effort is shown in Algorithm 3.

Apart from using only predefined class labels, the user can introduce new unseen class along with indicative attributes with the help of active learning. While prompting for label of data point x_t , we also collect the reason for their choice of label in restricted number of words. We find specific attributes from the provided reason and associate that attribute with the data point x_t . For example, if a user labels a data point as “getting up & sitting” and specifies the reason as “woke up from nightmare”, Sleep Well framework extracts the attribute “Nightmare” from the provided reason. Subsequently we re-evaluate our classification model and apply a recursive classification to associate the provided attributes to similar data points. This help our model to achieve microscopic sleep state classification, and finer evaluation for more elaborative and accurate diagnosis of patients and eventually scale the model beyond an individual premises.

V. SLEEP WELL FRAMEWORK EVALUATION

To evaluate our Sleep Well framework we focus on the following specificities. i) The performance of different classification algorithms in comparison to our classification approach, ii) Cross-user performance by building model with a user’s sleep model and testing with someone else’s model, iii) Performance of our framework using different wrist-band devices with accelerometer sensor, iv) Impact of active learning on our model, v) Precision of classifier when sleep attributes are introduced in the model by active learning.

A. System Implementation

We have implemented and tested our model by using two separate devices - wActiSleep-BT [4] and EZ430-Chronos [8], both of these devices contain 3-axis accelerometer sensor.

Algorithm 3 Active Learning with Importance Weighted Sampling

```

1: Input:  $L =$  set of labeled instances  $\{(x, y)^l\}_{l=1}^L$ 
    $U =$  set of unlabeled instances  $\{(x)^u\}_{u=1}^U$ 
   A classifier model,  $C_\theta$ 
2: Output: Updated classifier model,  $C_\theta$ .
3: Updated importance weight of queried data points.
4: for every instance in  $U$  do
5:   set  $p_t$  of instance  $x_t$  using equation (11)
6:    $y_t \leftarrow$  Prediction of  $C_\theta$  for  $x_t$ 
7:    $queried \leftarrow False$ 
8:   \* Check if  $x_t$  is a change point or not \*
9:   if  $x_t$  falls in between successive run  $r_{n-1}$  and  $r_n$  using
   the posterior probability  $P(r_n|x_{1:n})$  then
10:    if  $y_t$  is not same as the label of current run  $r_n$  then
11:      query label  $y_t$ .
12:       $L_t \leftarrow L_{t-1} \cup \{x_t, y_t, \frac{1}{p_t}\}$ 
13:       $queried \leftarrow True$ 
14:    end if
15:  end if
16:
17:  if  $p_t$  is greater than rejection threshold and  $queried = False$  then
18:    query label  $y_t$ .
19:     $L_t \leftarrow L_{t-1} \cup \{x_t, y_t, \frac{1}{p_t}\}$ 
20:  else
21:     $L_t \leftarrow L_{t-1}$ 
22:  end if
23:  Update the hypothesis space  $H_t$ 
24: end for

```

EZ430-Chronos device also has heart monitor, pressure and temperature sensor. We collected raw accelerometer data from both of these devices using API provided by the manufacturers. We implemented our own software to extract raw data using C# programming language and then extracted the features using python numpy library. We sampled the data in 60Hz frequency. For importance weighted classification and active learning we used the machine learning tool- Vowpal Wabbit [30].

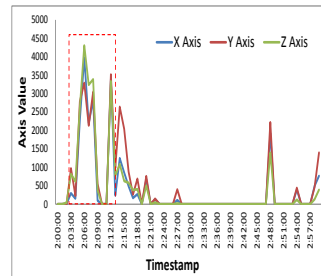


Fig. 2. Accelerometer reading when standing

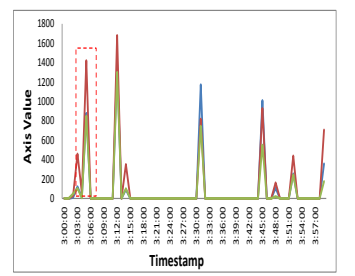


Fig. 3. Accelerometer reading when lying

B. Ground Truth Collection

We asked the users to log their sleep habits using sleep diary to correctly label the data points. We asked the participants to note down their sleep routines (preferred sleeping postures, regular hours of sleep, light intensity and sleep latency) each day of the experiment. There were many challenges involved

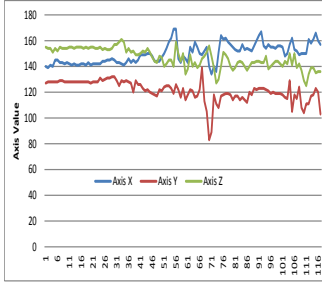


Fig. 4. Raw accelerometer data from Dataset V-C1.

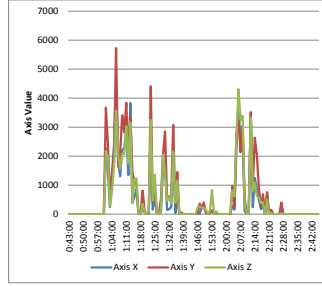


Fig. 5. Raw accelerometer data from dataset V-C2

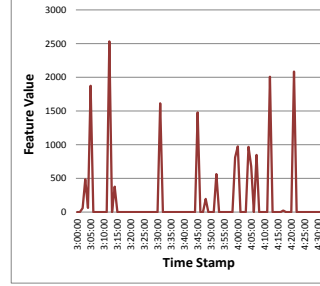


Fig. 6. Timestamped raw accelerometer data.

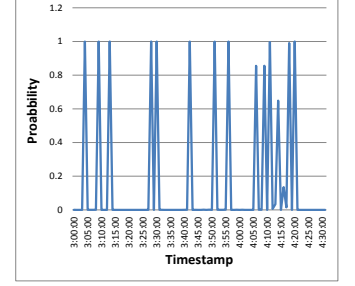


Fig. 7. Detected change points associated to figure 6

while collecting the ground truth from the sleep diary. For example, consider two different scenarios, 1) the user is *awake & lying* and 2) *awake & not lying*. In case of stationary states (when the user is not moving but he is either lying or just sitting in the bed) the accelerometer readings are almost identical. Also when a user gets up in the middle of the night and performs some activities (checking his phone, going to bathroom etc.), there are movements involved. It was challenging to identify which movements were during sleep and which were due to some activities. The user was unable to correctly state the reason of movements in some cases. In Fig. 2 and 3, we can see two different movements (awake and standing, awake and lying). The user went to bathroom at 2:03 AM and came back to bed at 2:12 AM. On the other hand at 3:03 AM, the user was moving while lying. Therefore to assist the ground truth collection, we investigate a posture analysis using the inclination of the accelerometer. We observe that when perpendicular to gravity, accelerometers are more sensitive to small changes in inclination, but as the inclination increases the accelerometer becomes less sensitive to it. To resolve this issue we propose to use two axis. As we are using wrist worn bands, inclination of axis y and z are used to define the posture of the user. The z axis measures the direction of the gravity in the horizontal position, so coupling with the inclination of x axis help infer the posture of the user. We calculate the inclination of the device by using Eqns. (12) and (13). We faced a challenge to define the threshold values

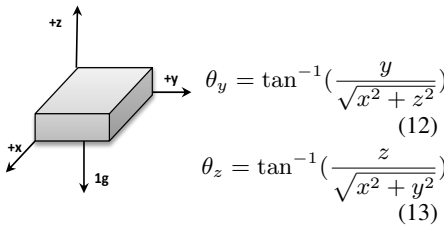


Fig. 8. Accelerometer orientation

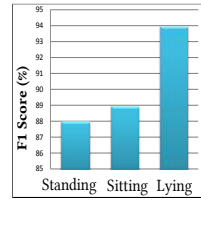


Fig. 9. Inclination measurement accuracy

for these inclinations as different users have different sleeping postures. We experimented with different sleep positions (On side, Face down, On your back) and calculated the inclination of the device in those positions. We ran the J48 decision tree classifier on the postural data. Based on the results of the classifier we defined the inclination threshold for different states, such as if $\theta_y < 16^\circ$ then the user is standing, if $16^\circ < \theta_y < 61^\circ$ the user is considered sitting, and for $\theta_x > 61^\circ$ the user is considered lying. Fig. 9 shows the results of our posture calculation using the inclination method. We

also installed couple of motion sensors in the environment to strengthen our ground truth collection. We put two motion sensors (Aeotec Multisensor [31]) near both sides of the bed and another one mounted near the user's body when he/she is sleeping. The sensor mounted near the body captures the motion when the user is in the bed while the other two on the sides of the bed monitor when the user is out of the bed. While extracting information from the sensor, we assumed that consecutive two data points from the sensors mounted on the sides correspond to getting in and then getting out of the bed. These multisensors also have built in light sensor, so we can detect the light condition in the sleeping environment using these sensors. Now we are able to validate the movements of the users and calculate the overall time he/she remained out of the bed efficiently by consolidating inclination measurement, motion sensor and data from sleep diary. We were able to label most of the data points correctly and remove noisy data points.

C. Datasets

We use real data traces collected from ≈ 60 users to validate the performance of our Sleep Well framework. We also compare our results for data from different body position.

1) *Dataset with Clinical Ground Truth*: We evaluate our model using publicly available benchmark dataset from Technische Universität Darmstadt [9] which provides sleep phases determined by clinical polysomnography. The data set consists of timestamped raw acceleration data collected using wrist worn data logger at a sampling rate of $100Hz$ and includes the sleep stages (movements, awake, NoREM 1-3, REM, unknown) from 42 lab patients. The trend of raw accelerometer reading in this dataset is shown in Fig. 4. There are seven different classes in this dataset among which majority of the data points are labeled as unknown (51%) and awake (24%) with only a few important data points which affect the classification model. After inspecting the dataset, we note that the value of different data points of different classes were very close which imposes bias in our classification model. We handle this bias by assigning less weight to abundant data points (unknown and awake) and improve the classification process and accuracy.

2) *Actigraph and Chronos Dataset*: We collected sleep data using wActiSleep-BT and EZ430-Chronos at a sampling rate of $60Hz$ from 17 participants. We noticed that wActiSleep-BT device has better sensitivity due to slight movements rather than EZ430-Chronos which help differentiate between actual movements and sleep patterns from a user. Almost 65% data

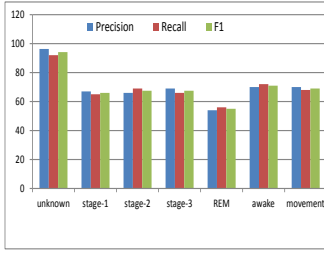


Fig. 10. Precision, recall and F1-measurement with increasing number of users (dataset V-C1)

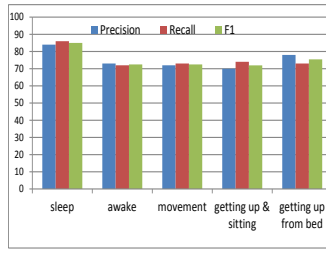


Fig. 11. Precision, recall and F1-measurement with increasing number of users (dataset V-C2)

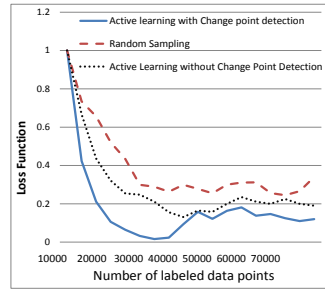


Fig. 12. Different active learning techniques for Dataset V-C1.

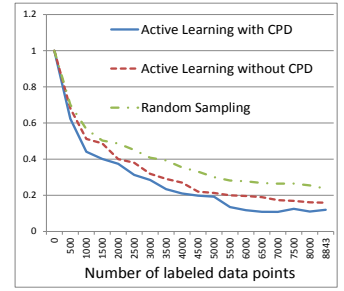


Fig. 13. Different active learning techniques for Dataset V-C2.

points of this dataset belong to *Sleep* class and 22% to *Awake* class. Fig. 5 shows the raw readings from ActiSleep device.

D. Evaluation Methodology

1) *Supervised Learning*: We carried out our experiments with 17 participants (11 males and 6 females) over two weeks where each participant has provided data for 8-10 days. We recorded the time when a subject went to bed and woke up from the participant’s sleep diary. Each participant was asked to put on the device on their waist before going to bed. Out of these 17 participants, 7 wore the EZ430-Chronos device and other 10 put on wActiSleep-BT. We split each data set into two parts, one for training and other for testing. We applied our classification models on the datasets mentioned in sections V-C1 and V-C2.

TABLE II. ACCURACY (DATASET : V-C1) (%) (INTRA USER)

	OSGD	SVM	MP	RF	LR	LB	DT
Unknown	98.76	96.50	85.80	99.01	98.93	98.37	97.95
Stage-1	69.45	44.44	60.36	58.57	47.63	61.82	70.40
Stage-2	70.29	41.02	58.54	59.09	48.18	68.46	71.87
Stage-3	68.36	39.15	63.36	48.24	49.33	60.77	63.94
REM	58.22	37.28	49.11	41.55	38.31	41.03	59.10
Awake	74.78	68.12	72.10	70.01	64.96	66.90	72.73
Movement	72.59	69.31	62.88	70.66	65.60	63.27	71.25
Average	73.20	56.54	66.13	63.87	58.99	65.80	72.46

TABLE III. ACCURACY (DATASET : V-C2) (%) (INTRA USER)

	OSGD	SVM	MP	RF	LR	LB	DT
Sleep	87.79	87.98	80.8	84.01	73.32	76.63	88.52
Awake	77.9	71.35	75.66	67.91	73.56	70.21	75.69
Movement	76.25	74.87	68.32	68.41	70.27	72.11	75.14
Getting up & sitting	72.11	64.58	67.39	68.11	64.85	69.15	70.02
Getting up from bed	78.21	69.89	70.36	70.1	71.19	62.39	73.39
Average	78.45	73.73	72.50	71.70	70.63	70.09	76.54

a) *Intra User Classification*: We tested different classification models with our proposed Online Stochastic Gradient Descent (OSGD) method - Support Vector Machine (SVM), Multilayer Perceptron (MP), LogitBoost (LB), Random Forest (RF), Logistic Regression (LR), and Decision Tree (DT) - using different user’s dataset. The accuracy of different classification model using one of the subject’s dataset from each datasets is shown in Table II and III. The average accuracy of OSGD is 73.20% for a patient from dataset V-C1 and 78.45% for dataset V-C2. This attests that consideration of inclination and sensor data and using it to correct labels in dataset V-C2 help yield better classification results. Also the results indicate that putting the device on the waist endows better accuracy.

We investigated this disparity, and found that hand movements are more abrupt and arbitrary which results in more confusing data points. Also very subtle body movements are difficult to distinguish when using a wrist worn accelerometer.

The major accuracy improvement was noticed for inferring the micro sleep state. Although individual accuracy for classes - *Stage 1*, *Stage 2* and *REM* for Decision Tree (DT) classifier was better in dataset V-C1, but the average accuracy of inferring sleep states (sleep stage 1-3, REM) is 66.58% which is better than the average of DT classifier (66.32%). While for our dataset we achieved 87.79% accuracy.

b) *Cross User Classification*: It is important that a classification process will not only recognize the sleep states of an already seen user, but also help generalize the classification for new users. We cross validated our approach with inter user classification model. We trained our model using 20 patient’s data from dataset V-C1 and tested the trained model with remaining 22 patient’s data. The average accuracy was 69.79%. With data from dataset V-C2 we achieved 75.46% overall accuracy. Fig. 10 and 11 shows the results in Precision (percentage of times that a recognition result made by the model is correct), Recall (percentage of times that a sleep state is detected) and F1-measurement (combination of both recall and precision) for both the datasets. Also in Fig. 14 represents the trend of loss function for different datasets.

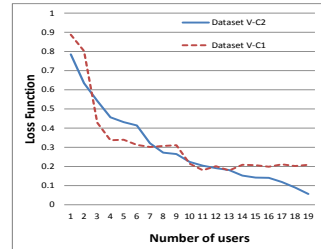


Fig. 14. Trend of loss function for inter user classification in different datasets

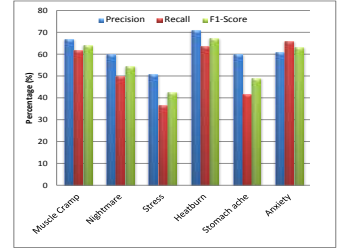


Fig. 15. Different attributes classification result in getting up & sitting partition.

2) *Active Learning Experiments*: In addition to supervised learning, we evaluate how we can improve the classification result using active learning with minimal user feedback. We have discussed our active learning algorithm in section IV-D. We sampled both the datasets with a window of 60 seconds on accelerometer data. Each sample is a feature vector with 16 dimensions. Initial labeled dataset L_1 consisting of 135089 samples (from dataset V-C1) and L_2 consisting of 42,000

samples (from dataset V-C2) are provided to the individual classifier C_1 and C_2 for training. Then unlabeled dataset U_1 of 510113 (dataset V-C1) and U_2 of 121,147 samples (dataset V-C2) are used to test the classifier C_1 and C_2 . The samples are provided sequentially with respect to timestamp.

The uncertain data points, meaning the points which the classifier was unable to classify are queried in accordance with our active learning algorithm (3). We calculated the loss function at each phase after a data point is queried and the model is re-trained. We compared our result with randomly selected samples for labeling. To further assist the active learning process we validated the results with our change point detection (CPD algorithm discussed in IV-B. When a change point is detected in the dataset, we cross validated the change points with the classification result with respect to the timestamp. Figs. 6 and 7 plot the association of change points with timestamped accelerometer data points. If the label of the sample is not consistent between each of the model we imposed active learning and queried the data point. Initially with L_1 and L_2 we note the average classification accuracy as 63.8% and 70%. We applied importance weighted active learning, and see that the model converged faster with change point detection. 86719 samples (17% of total samples) from U_1 and 8843 samples (7.3% of total samples) from U_2 were queried for the model to converge in presence of CPD which helped achieve 72% (dataset V-C1) and 76.89% (dataset V-C2) accuracy, while with randomly selected data points 68% and 73% accuracy was observed. Fig. 12 and 13 shows the change in loss function with random sampling, active learning with and without CPD techniques with different datasets. We see that active learning with CPD outperforms the other strategies. In case of dataset V-C1 we notice from Fig. 12 that the change in loss function is irregular. After analyzing the dataset V-C1 we found out that due to the presence of noisy data points the loss was increased.

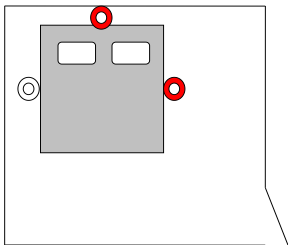


Fig. 16. Visual illustration of sensor activation.

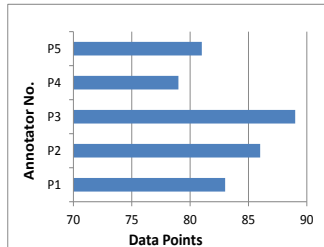


Fig. 17. Percentage of data points properly labeled via crowdsourcing.

We ran a simulation for crowd-sourcing using vovpal-wabbit toolkit [30]. During this process we faced a challenge regarding what kinds of data to show which can reflect the sleep classes. As audio, video or image data violate the privacy of the user so we had to come up with a different methodology rather than traditional image based crowd-sourcing. We presented some semantic information from the users sleeping habit (regular hours of sleep, sleep latency, average number of times the user gets up at night, how much the user moves on average in percentage and light condition) and a visual illustration of sensor activation (discussed in section V-B) to the annotators. In Fig. 16, we show an example of visual illustration. The double circled objects represent sensors and

the activation is marked by red color. In this example the sensor mounted near the head and the sensor mounted near the right side of the bed are activated as the user was getting up from the bed. Five participants participated in the crowd-sourcing experiment and each participant was given 100 data points to label. In Fig. 17 we show the number of data points each participant was able to label. The average percentage of correctly labeled data was around 83.6% which is sufficiently high and can be chosen as input to the classifier.

3) *Introduction of New Unseen Class and Attributes:* A user is able to personalize the model by introducing new unseen classes and attributes with the help of active learning. We simulated our active learning algorithm by introducing new class labels in the classification model. While collecting the query label we also asked for the reason behind choosing the label from the annotator, so that we can look for important indicators for the clinicians. We restricted the length of the reason in 5 words. For example, if a sample is queried and the annotator labels the sample as *getting up and sitting*, he can also state the reason for labeling the data such as *muscle cramp, stress or anxiety, nightmare* etc., which are microscopic events for sleep disruption. We applied a nested classification by considering these microscopic events as class labels. After classifying using our defined general class labels, we partitioned each class label data and applied our classification algorithm in separate partitions again by considering the provided attributes as labels. For example, let us assume an user states reason ‘A’ as the cause of sleep disruption or any kinds of changes in the pattern. Our framework then partitions the data and the number of partition is equal to the number of class labels (in our model it is 5), as a result in each partition the data points are of same class. Sleep Well framework then performs a classification on separate partitions with class label ‘A’. This nested classification process ascertain the microscopic sleep events. The precision, recall and F1 score of recorded attributes (muscle cramp, heatburn, stomach ache, stress, anxiety, and nightmare) for parent class “*getting up and sitting*” is presented in Fig. 15.

VI. DISCUSSION AND FUTURE DIRECTIONS

In the current version of Sleep Well framework, we did not discuss about individual sleep scoring based on our sleep state classification. Most of the sleep scoring models like Pittsburgh Sleep Quality Index (PSQI) [32] or Webster et al. [33] do not consider the habit of individual’s sleep. In our experiment, we found that one participant was moving frequently while sleeping and woke up 2 or 3 times at night to use the bathroom. Even if the participant had disrupted sleep according to our data, but in accordance to participant’s feedback he had a sound sleep. We look forward to investigate the sleep habits of individuals (like movements and getting up frequently) using change point detection in future and devise a dynamic sleep scoring module.

In current version of our framework, we invoke the change point detection only in case of active learning. As a future research direction, it is possible to combine change point detection with classification model to perform a more thorough time series analysis. For example, abrupt sleep disturbances (muscle cramp, nocturnal panic attacks etc.) cause sudden changes in the data points. It would be beneficial to capture

these subtle changes and correlate the run length with the classification process.

In the current implementation, we considered only the inclination of the device to infer the user's current posture. Therefore, another future direction is to integrate the locomotive activity (such as sitting, standing, walking etc.) recognition with our framework for improving the noise reduction methodology. For active learning experiments we assumed that the user will always provide the correct label. In real life, it is possible that the user may provide wrong label or leave it blank. In such cases, imperfect annotation handling and optimal querying can be further studied to improve the performance of active learning. This is also applicable for crowd-sourcing the sleep data for malicious user identification. In our current state, we collect the reason of the label from the user and extract the important microscopic attributes from the provided reason. For future, these feedbacks can be further leveraged to inspect the nature of various sleep disturbances at the microscopic level which will greatly help in longitudinal sleep assessment and diagnoses.

ACKNOWLEDGEMENT

This work is supported partially by the UMB-UMBC Research and Innovation Partnership Grant, NSF Award #1344990, and Constellation E^2 : Energy to Educate Grant.

VII. CONCLUSIONS

In this paper, we described the design, implementation and evaluation of Sleep Well, a sleep monitoring framework which classifies the microscopic sleep states using wearable devices. The proposed algorithm formulated in a gradient descent approach, cooperate with importance weight aware updates. We also consolidated our framework by blending change point detection and active learning in the process. Our classification achieved 78% accuracy with the aforementioned experimental setup. The empirical results demonstrate the effectiveness of our framework in determining different sleep states. The result increased by 7% when active learning was advocated. Therefore, the result support faster convergence to optimal accuracy using minimal user feedback in presence of active learning. Besides with the help of change point detection, we were able to validate and interpret the transitions between these sleep states. In future we will investigate the combination of change point detection and classification to further improve the accuracy. Also by conforming attributes from user provided feedback into our architecture will further provide meaningful insights for better understanding of sleeping behavior.

REFERENCES

- [1] J. M. Parish, "Sleep-related problems in common medical conditions," *CHEST Journal*, vol. 135, no. 2, pp. 563–572, 2009.
- [2] A. L. C. et al, "Practice parameters for the indications for polysomnography and related procedures," *Sleep*, vol. 20, pp. 406–422, 1997.
- [3] "Fitbit," <http://www.fitbit.com/>.
- [4] "Actigraph," <http://www.actigraphcorp.com/>.
- [5] "Basis band b1," <http://www.mybasis.com/>.
- [6] "Android wear," http://www.android.com/intl/en_us/wear/.
- [7] S. Jonas, A. Hannig, C. Spreckelsen, and T. M. Deserno, "Wearable technology as a booster of clinical care," 2014.
- [8] "Ez430-chronos," <http://www.ti.com/tool/ez430-chronos>.

- [9] M. Borazio, E. Berlin, N. Kücükıldız, P. M. Scholl, and K. Van Laerhoven, "Towards benchmarked sleep detection with inertial wrist-worn sensing units," in *ICHI 2014*, Verona, Italy, 09/2014 2014.
- [10] M. W. Johns, "Sensitivity and specificity of the multiple sleep latency test (mslt), the maintenance of wakefulness test and the epworth sleepiness scale: Failure of the mslt as a gold standard," *Journal of Sleep Research*, vol. 9, no. 1, pp. 5–11, 2000.
- [11] M. Rofouei, M. Sinclair, R. Bittner, T. Blank, N. Saw, G. DeJean, and J. Heffron, "A non-invasive wearable neck-cuff system for real-time sleep monitoring," in *BSN*. IEEE, 2011, pp. 156–161.
- [12] K. Nakajima, Y. Matsumoto, and T. Tamura, "A monitor for posture changes and respiration in bed using real time image sequence analysis," in *Engineering in Medicine and Biology Society*, vol. 1, 2000.
- [13] W.-H. Liao and C.-M. Yang, "Video-based activity and movement pattern analysis in overnight sleep studies," in *ICPR*, 2008, pp. 1–4.
- [14] M. Kay, E. K. Choe, and et al., "Lullaby: A capture & access system for understanding the sleep environment," in *UbiComp*, 2012.
- [15] N. Foubert, A. McKee, R. Goubran, and F. Knoefel, "Lying and sitting posture recognition and transition detection using a pressure sensor array," in *Medical Measurements and Applications Proceedings (MeMeA)*, 2012, pp. 1–6.
- [16] E. Hoque and J. A. Stankovic, "Monitoring quantity and quality of sleeping using wisps," in *IPSN*, 2010, pp. 370–371.
- [17] J. Liu, W. Xu, and et al., "A dense pressure sensitive bedsheet design for unobtrusive sleep posture monitoring," in *PerCom*, 2013, pp. 207–215.
- [18] "Zeo," <http://www.myzeo.com/sleep>.
- [19] T. Hao, G. Xing, and G. Zhou, "isleep: Unobtrusive sleep quality monitoring using smartphones," in *SenSys*, 2013, pp. 4:1–4:14.
- [20] M. Fahim, L. Vui, I. Fatima, S. Lee, and Y. Yoon, "A sleep monitoring application for u-lifecare using accelerometer sensor of smartphone," in *Ubiquitous Computing and Ambient Intelligence. Context-Awareness and Context-Driven Interaction*, 2013, vol. 8276, pp. 151–158.
- [21] Z. Chen, M. Lin, F. Chen, N. Lane, and et al., "Unobtrusive sleep monitoring using smartphones," in *PervasiveHealth*, 2013, pp. 145–152.
- [22] N. D. Lane, M. Mohammad, and et al., "Bewell: A smartphone application to monitor, model and promote wellbeing," in *Pervasive Computing Technologies for Healthcare*, 2011.
- [23] Y. Bai, B. Xu, Y. Ma, G. Sun, and Y. Zhao, "Will you have a good sleep tonight?: Sleep quality prediction with mobile phone," in *Proceedings of the 7th International Conference on Body Area Networks*, 2012.
- [24] W. Gu, Z. Yang, L. Shangguan, W. Sun, K. Jin, and Y. Liu, "Intelligent sleep stage mining service with smartphones," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '14, 2014, pp. 649–660.
- [25] D. B. F. J. Kaplan A, Röschke J, "Macrostructural eeg characterization based on nonparametric change point segmentation: application to sleep analysis," in *Journal of Neuroscience Methods*, 2001, pp. 81–90.
- [26] K. E. B. R. Citi L, Bianchi MT, "Instantaneous monitoring of sleep fragmentation by point process heart rate variability and respiratory dynamics," in *Engineering in Medicine and Biology Society*, 2011., 2011.
- [27] N. Karampatziakis and J. Langford, "Online importance weight aware updates," in *Proceedings of the Twenty-Seventh Conference on Uncertainty (UAI) in Artificial Intelligence*, 2011, pp. 392–399.
- [28] R. P. Adams and D. J. MacKay, "Bayesian online changepoint detection," Cambridge, UK, 2007.
- [29] A. Beygelzimer, S. Dasgupta, and J. Langford, "Importance weighted active learning," in *ICML*, 2009, pp. 49–56.
- [30] "Vowpal wabbit," <http://hunch.net/~vw/>.
- [31] "Aeotec multisensor," <http://aeotec.com/z-wave-sensor>.
- [32] D. Buysse, C. Reynolds, T. Monk, S. Berman, and D. Kupfer, "The pittsburgh sleep quality index: a new instrument for psychiatric practice and research," in *Psychiatry Research*, vol. 28, 1989, pp. 193–213.
- [33] J. B. Webster, D. F. Kripke, and et al., "An activity-based sleep monitor system for ambulatory use," in *Sleep*, vol. 5, 1982, pp. 389–399.