SleepMore: Sleep Prediction at Scale with Multi-Device WiFi Sensing

Camellia Zakaria[1], Gizem Yilmaz[2], Priyanka Mammen[1], Bryce Parkman[1],
Michael Chee[2], Prashant Shenoy[1] and Rajesh Balan[3]
[1]University of Massachusetts Amherst, [2]National University of Singapore, [3]Singapore Management University

ABSTRACT

Sleep is an essential part of health, providing the foundation for our daily functions and habits. Despite a large number of research and commercial endeavors to accurately predict sleep using different sensing modalities, it is still challenging to monitor sleep patterns at scale and across long time periods due to challenges such as high attrition rates, privacy concerns, or even the cost of buying wearable devices. The advancement of passive WiFi sensing can offer promising opportunities to address these challenges. Given the widespread adoption of multiple WiFi-enabled devices per user, we propose an approach that uses the WiFi device activity from multiple devices collected directly from the WiFi infrastructure to predict sleep. Our proposed solution, *SleepMore* provides an accurate easy-to-deploy sleep tracking solution that can be used to track the sleep patterns of entire households and dormitories with no changes in user behavior – in particular, no apps or wearable devices need to be worn or installed. *SleepMore* uses a two-step prediction technique that first employs an XGBoost semi-personalized model to classify a user's network activity behavior into sleep and awake states every 15 minutes. It then uses these state sequences to predict, using a moving average, the user's sleep onset, wake time, and sleep duration. We validate *SleepMore* using data collected from a user study involving 68 college students and compare it with a state-of-the-art Oura Ring wearable sleep tracker. Our results show that *SleepMore* can predict users' bedtime and wake time within 36 minutes and 28 minutes error, respectively, a significant improvement over prior work. In addition, the sleep statistics produced by *SleepMore* are statistically indistinguishable from those produced by the Oura. We perform an in-depth analysis to identify the sources of errors in *SleepMore*'s predictions and propose an anomaly detection solution to correct a large number of these errors.

1 INTRODUCTION

Sleep is essential to one's physical, emotional, and mental health. The National Sleep Foundation (NSF) recommends between 7 to 9 hours as appropriate for adults to maintain general wellness [16]. However, sleep deficiency is a common public health problem, with approximately 50 to 70 million Americans suffering from chronic sleep disorders [2]. Among university students alone, a survey conducted by the American College Health Association (ACHA) in 2020 found only 4.9% of the student population receiving the recommended amount of sleep on a weeknight [4]. Students make up a large user demographic at high risk of developing sleep problems caused by academic and societal issues [12, 26]. The risk factors associated with insufficient sleep are performance and cognitive deficits, and its long-term effects are correlated with severe consequences such as obesity, stress, depression, and stroke [2, 16]. Thus, accurately determining sleep habits remains a topic of immense interest.

The gold standard for measuring sleep quality is to use polysomnography (PSG)—a multichannel, multimodal approach that is performed in controlled environments like a sleep clinic by trained technicians. These factors make PSG challenging to use for large-scale and multi-day studies. More user friendly sleep tracking solutions have been provided by numerous relatively low cost *wearable sleep trackers* (e.g., FitBit, Jawbone) [8, 36]. However, these require direct participation and behavior changes. Indeed, many studies continue to face high attrition rates due to users who are uninterested in embracing wearables at bedtime [8]. Users who have the most disrupted sleep patterns tend to be those who do not wear sleep trackers.

To overcome this, researchers have explored contactless methods using radio and radar signals [17, 35]. However, the requirements to instrument building infrastructure limit their scalability. Recently, using WiFi signals such as channel state (CSI) and backscatter information has been proposed as viable solutions for sleep tracking [41]. This is made possible by both smartphones becoming common across all income levels[31] and WiFi solutions becoming prevalent in public, institutional and residential locations [29, 32]. These solutions attempt to detect when the user has fallen asleep by checking for vibrations and other signals. Unfortunately, even though these solutions have reasonable accuracy, they require custom hardware and are thus hard to deploy at scale.

Smartphones have been used for sleep detection and monitoring. Prior approaches use smartphone activity as a proxy for user activity and long periods of device inactivity to infer sleep periods. These approaches include both client-side methods, based on monitoring screen activity [6], as well as network-side methods, based on monitoring WiFi network activity of the device [21]. Since users tend to habitually use their mobile device before they sleep and upon waking up, device activity, or lack thereof, is a feasible approach for detecting sleep periods. While feasible, they are not yet practical for everyday use from producing significant errors, often up to an hour or more, in detecting sleep and wakeups.

Our paper focuses on making the approach practical by employing a multi-pronged approach to reduce errors. First, it uses network activity from multiple user devices rather than a single device such as a smartphone. In doing so, it reduces errors seen by single device methods [6, 21] since it can handle varying user behavior where a different device (e.g., a tablet) may be used prior to bedtime and upon wakeup. Second, some prior methods [21] are based on unsupervised learning that uses priors to capture user behavior. Ours is based on supervised learning methods, and we show that even a small amount of training data is adequate to reduce errors seen in unsupervised methods. Finally, our analysis of the error distribution shows that a small number of outlier days contribute to a large fraction of the observed error in smartphone-based methods. Sparse or very noisy data cause these outliers. We remove these outliers and significantly improve the prediction accuracy. We hypothesize that using multiple mobile devices, a small amount of training data, and the filtering of sparse data using anomaly detection can significantly improve the accuracy of mobile device-based sleep detection techniques, making them a practical approach for sleep detection and monitoring.

Accordingly, we develop *SleepMore*, a lightweight machine learning (ML) approach, leveraging passive WiFi sensing of multiple devices used by a user to predict their sleep patterns. *SleepMore* collects network activity information of all user-owned devices directly from the WiFi access points (AP). Thus it achieves scalability and accuracy with no active user participation, and only requires that users connect all their devices to the same WiFi network. Also, *SleepMore* does not collect any packet information beyond the fact that a packet was generated (i.e., *SleepMore* only knows that the device is communicating but has no details beyond that). *SleepMore* also works even with a single AP, such as a home, as it does not do any localization – beyond identifying APs (through their BSSIDs) that correspond to possible sleep locations. We explicitly chose this sensing modality to scale across homes and dormitories without changing user behavior. All the user has to do is connect to the WiFi network using their devices, which is already part of their daily routines.

Leveraging WiFi network device activity, *SleepMore* extracts features representing the user's device use for a machine learning classifier to identify sleep states. A key challenge in our approach is estimating their bedtime, wake time, and sleep duration from coarse WiFi as device use does not reflect the user's physical state. While it is not unusual for a person to use their device before bedtime, it is also not unusual for a person to take some time to fall asleep after putting their device away. We implement moving average prediction to determine the start and end of a user's nocturnal sleep, providing robust results across different users and sleep behaviors. Our experiments explain circumstances where sleep is not always detectable through this sensing modality. Challenges are attributed mainly to sparse data resulting from users not being home at most parts of the evening or sudden and drastic changes in their sleep pattern.

We validated *SleepMore* via an IRB-approved longitudinal study conducted over 4 weeks during the Spring 2021 academic semester. 68 undergraduate students who resided on campus participated and provided us with their sleep and wake time baseline information through diary logs. As part of the study protocols, the participants also wore the *Oura ring* [27] wearable sleep tracker for additional baseline and were required to connect to the campus WiFi while in their residential dormitories. Students supplied the WiFi MAC addresses of the multiple devices they owned (i.e., smartphones, tablets and laptops) so that we could identify these devices directly from the WiFi infrastructure and extract their network event logs (by default, all WiFi traces are anonymized).

To the best of our knowledge, this is the first work to accurately predict sleep patterns using a scalable WiFi-based technique using inputs from multiple user-owned devices. Our evaluation demonstrates *SleepMore* achieving a mean accuracy of 36 minutes at detecting the bedtime (i.e., sleep onset) and 28 minutes at detecting the wake time using multiple device WiFi-sensing. Overall, *SleepMore* provides an accurate easy-to-deploy sleep tracking solution that can be used to track the sleep patterns of entire households and dormitories with no changes in user behaviour – in particular, no apps or wearable devices need to be worn or installed. Our primary contributions are:

(1) We present *SleepMore*, a solution for accurately detecting user sleep patterns using passively collected WiFi network activity data from multiple user devices. In particular, *SleepMore* does not require users to wear any sleep trackers to bed or to change their behaviour.

(2) We show that *SleepMore* has high prediction accuracy even though it is using coarse-grained WiFi data as its sole input. In addition, we show that using data from multiple devices provides a significant improvement over just using a single device. Overall, *SleepMore*'s sleep duration outcomes are statistically insignificant ($p>.1$) compared to those generated by the Oura ring.

(3) We show that *SleepMore* is robust to sleep and user characteristics, including chronotype and regularity patterns. Given its comparable performance to the Oura ring, *SleepMore* offers a promising workaround to addressing low user compliance with wearables without posing privacy threats or interfering with their natural behavior.

2 BACKGROUND AND MOTIVATION

This section provides background on sleep monitoring applications and sensing techniques, and motivates our multi-device passive-sensing approach.

Contact-based wearables and contactless sensing have shown feasibility in monitoring sleep. Zambotti's defined *sleep wearable trackers* as "over-the-counter, relatively low-cost devices available without prescription or clinical recommendations." [8] . These devices vary from wristbands to smartwatches, earbuds to rings. Challenges with wearables include a high user attrition rate over long monitoring periods [8] and the need for regular "use the wearable" reminders [22, 43]. Low compliance with wearables has motivated the design of contactless sensing approaches. Examples include the use of cameras [18], RF or radar sensing to characterize sleep stages [35] and posture [42]. However, RF sensing systems have still not reached commercial adoption.

Smartphone-based sensing The significant rise in smartphone dependency [30] has led to sleep monitoring effects using the "phone-as-a-sensor". Some of these efforts include distinguishing respiratory patterns through a microphone [38] or inferring user sleep behavior from monitoring screen interactions and application usage [1, 7, 13, 14, 24]. The observation that smartphones are the last thing many people use before they sleep and the first thing they use when they wake up [39] makes it viable as a sleep monitoring sensor.

2.1 Motivation

Prior work by Mammen *et al.* has shown that it is possible to use WiFi connection logs collected from the WiFi infrastructure to predict sleep [21] by employing an unsupervised Bayesian inference method. *SleepMore* extends on this idea in the following ways:

(1) The prior *WiSleep* work achieves an average error of 102 minutes at predicting sleep time. This is quite far from the accuracy achievable by sleep wearable trackers such as the Oura Ring that can achieve 96% recall at detecting sleep [3]. With this result in mind, we aim to propose a solution that also uses WiFi sensing but can significantly outperform prior work. [21].

(2) Prior work only uses data collected from a single smartphone based on the assumption that users spend most of their time online and on their smartphones [40]. We hypothesize that including all devices owned by the user will significantly improve the accuracy of sleep detection.

(3) Prior work[21] has used Bayesian unsupervised approaches due to the lack of training data. In contrast, we apply supervised machine learning methods to train a model with data collected during an extensive data collection user study. In addition, we compare our results with baseline data collected by an Oura sleep sensing ring [27] worn by all study participants.

(4) Finally, we developed methods to detect noise and other anomalies in the data that could lead to high prediction errors. We show that by detecting and removing these anomalies, we can significantly improve the overall prediction accuracy.

Design Rationale. Our work uses the WiFi network device activity collected from multiple devices to monitor an individuals' sleep. Specifically, it monitors the devices that are associated with the network to infer the user's bedtime and wake-up time (referred to as T_{sleep} and T_{wake}). Adding new user-owned devices is easy as they would only need to be connected to the same WiFi network. In addition, our solution significantly preserves privacy as it does not require decoding the actual content of any packets sent by a device. For each user-owned device (denoted by their MAC address), we capture the WiFi connection events that these devices make to the network and compute the frequency at which each device establishes network connections.

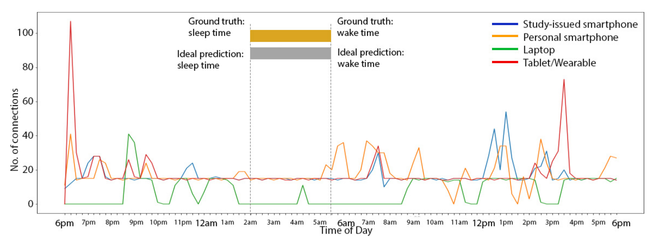

Figure 1: Low WiFi device activity from multiple devices corresponds to a sleep behavior.

Figure 1 shows an example of the network connection frequency for a typical user with multiple devices every 15-minutes through 24 hours between Day_1, 6 pm to Day_2, 6 pm. This example yields several critical insights: (1) The connection frequency increases with active online device utilization and decreases with less utility. For example, we can observe the user displaying the highest device use between 6-7 pm using a tablet. (2) A user also switches between four devices throughout the 24 hours, highlighting the potential for a more comprehensive behavioral monitoring by expanding device selection. (3) The user's smartphone is the last device used before bedtime and the first device used upon waking up. (4) While the frequency of connection increases with more device use, it is important to note that increased device activity does not always imply a user being active in true nature. For example, it can occur from an application on the device running automatic software updates or accepting notifications (e.g., incoming emails, received online messages), as demonstrated by the user's laptop network connection log, which peaked between 4 to 5 am.

These observations suggest that multi-device monitoring will lead to significantly better sleep prediction than using a single device. Indeed, our results in Section 5 confirm this.

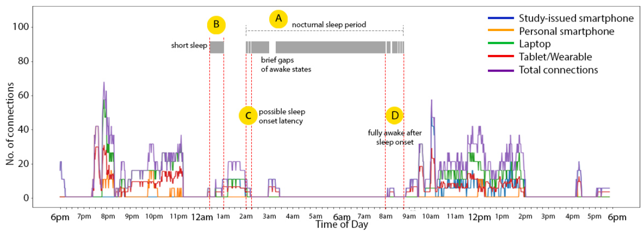

Figure 2: Challenges predicting sleep with WiFi data.

Key Systems Challenges. However, these observations also demonstrate non-trivial challenges in using device network events as a predictor for sleep. First, as shown in Figure 2, increased WiFi network activity can be caused by device updates and app notifications that do not reflect the user's actual physical state. For example, the user is asleep at 3:00 am while the device updates or apps receive notifications. Second, while it is not unusual for a person to use their device before bedtime, it is also not unusual for a person to take some time to fall asleep after putting their device away [11, 19]. This duration is known as *sleep onset latency* and is estimated to be within 20 minutes. However, it also increases progressively with age and electronic device-use [5, 15, 20]. Thus, the second challenge is determining the true start of a user's bedtime, as the first predicted sleep state may not reflect the user falling asleep. Conversely, the last predicted sleep state may not imply a user waking up (see Figure 2C and D). Third, while it is reasonable to assume a user interacting with their device before bedtime and upon waking up [11, 19], it may not always be the same device used all the time. *SleepMore* was designed to overcome these challenges.

Preserving Privacy. A key consideration in this work is to preserve users' privacy while still being functional. As such, *SleepMore* only uses network activity data without knowing what that network activity corresponds to. In addition, *SleepMore* will only provide user-specific sleep predictions if the user explicitly provides the MAC addresses of their devices. Without these MAC addresses to user device mappings, *SleepMore* can still produce sleep results at aggregate levels but without attributing results to users. Aggregated results remain helpful as they can provide a good overview of the health status of an entire home or residential dormitory without needing to identify individual users.

3 *SLEEPMORE* DESIGN

This section presents our design of *SleepMore* as a multi-device WiFi sensing approach for sleep monitoring. Our implementation of *SleepMore* is a two-step process, which first classifies users in *sleep* or *awake* states in intervals, and second, takes the sequences of these states to define when the user is sleeping. In doing so, *SleepMore* estimates the user's bedtime and wake time, T_{sleep}, and T_{wake}. Figure 3 presents the system overview. We describe the implementation details as follows.

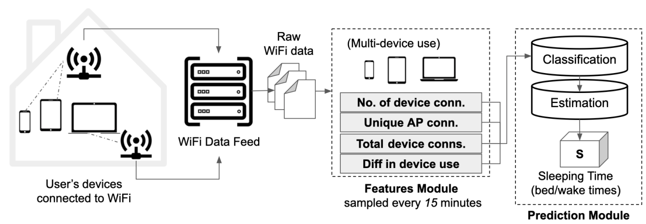

Figure 3: *SleepMore*, a two-step process in predicting sleep states, and estimating bed and wake times.

3.1 Collecting WiFi Data Using RTLS

SleepMore requires WiFi data indicating when a user's device(s) communicate on the network. To obtain this in a scalable way, we leverage the real-time location system (RTLS) data feed available on commercial access points. For this paper, we collected all our data from a university dormitory that uses Aruba WiFi APs that support RTLS [25]. The RTLS feed provides reports from every AP that sees a device communicating on the network. Each RTLS message contains the following information:

```
Timestamp, Packet Age, Data Rate, DeviceType, Channel,
DeviceMACAddress, AssociationStatus, RSSI, Noise
Floor, BSSID, MON BSSID
```

SleepMore only uses data from associated client devices as those clients have legitimate access to the network. In particular, we discard all data from unassociated devices as they are primarily WiFi probe requests from clients roaming for a usable network. Note: while RTLS feeds can be used to track the location of WiFi devices, *SleepMore* is only using this data as a proxy for device activity.

The RTLS reports are generated by all APs every 5s in our test environment. However, these reports will only list the WiFi devices that were active and seen by that AP (i.e., they were using the same WiFi frequency as the AP and close enough to that AP) in that 5s interval. In particular, if a device has gone to sleep (e.g., because the user is sleeping and the device is charging), the device will not be emitting any network packets. It will not be seen and reported by any RTLS report generated by any AP.

3.2 Pre-processing Module

The first step in *SleepMore*'s pre processing pipeline is to clean the noisy WiFi RTLS data. In particular, we remove data with invalid timestamps or RSSI values too weak (indicating spurious transmission). We only retain records of WiFi device activity on days that users provide their Oura data (see Table 1 for data summary).

SleepMore's features are generated from the RTLS *Timestamp* (records time at which a data message is received), *Device Type* (client or AP station), *Device MAC Address* (client or AP station), and *Association status* (associated or unassociated device) fields. Because a user could own multiple devices, each with their own MAC address, *SleepMore* generates device features primarily based on the number of associated connection events per device (i.e., *networkEvents*) and the unique AP (i.e., *uniqueAPs*) to which these devices were connected. We use the AP connected to as a feature to avoid cases where a user is moving with a phone in power saving mode – in this case, the phone will not generate many network events, but the change of APs show that the user is not sleeping. We assume that a device connects to just a single unique AP when a user is sleeping – this assumption can be relaxed easily (by adding a set of APs into the feature set) in the rare cases where this assumption is not true. In Section 5.1.2, we show that generating these features every 15 minutes, similar to prior work [21], struck a good balance between accuracy, computational overhead, and resilience to noise.

Figure 4 shows the amount of correlation between sets of features (highly correlated features are darker in color and have scores close to or exactly 1.0). We observe that for all device types (smartphone, laptop, tablet), the number of unique AP associations (uniqueAPs) is highly correlated with the number of WiFi connection events (networkEvents). In Section 5.2, we use these results to remove all highly-correlated features (i.e., unique APs for all devices) and adopt a standard machine learning pipeline of feature selection. *SleepMore*'s final feature selection uses importance weights, which is the number of times a feature is used in the fitted trees inside the XGBoost classifier.

3.3 Sleep Prediction Module

SleepMore's final sleep prediction module runs both the machine learning classification and estimation models.

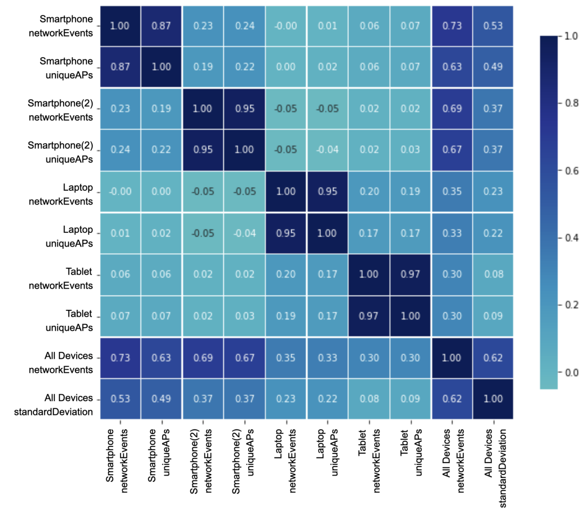

Figure 4: Feature correlation heatmap.

Machine Learning Classification Model: *SleepMore* produces a binary predictions of whether the user is currently in a *sleep (1)* or *awake (0)* state. As per Figure 3, *SleepMore* first derives a set of attributes from the user's multi-device WiFi device activity logs collected every 15 minutes. Using the features above as input, we run different machine learning techniques to produce the required binary prediction.

Section 5.1 compares several machine learning techniques, including Decision Tree (DT), k-nearest neighbors (k-NN), Random Forest (RF), and Extreme Gradient Boosting (XGBoost) algorithms for this problem. We found that the XGBoost classifier was the best. XGBoost is a gradient boosting approach that uses an ensemble machine learning model algorithm, well known for its computational efficiency and model performance. The ensembles are based on decision trees, giving a straightforward interpretation of features.

Implementation: Our ML model is trained offline, and the inference engine was implemented using the standard scikit-learn framework [28]. The model is trained with 30% user data, thus semi-personalized. The RTLS data is processed by the binary classification engine using XGBoost to predict the *sleep* or *awake* state of a user every 15-minutes between $Day1$ 6:00 pm to Day_1 and Day_2 6:00 pm. Finally, our model's critical parameters, namely the number of trees, tree depth, and learning rate, are tuned to avoid model overfitting.

Estimation Model: For *SleepMore* to accurately predict a user's sleep patterns, it will have to overcome the challenges described in Section 2.1. Specifically, by putting all binary state predictions together, we can conceivably determine a user's nocturnal sleep period for that 24-hour cycle by choosing the most extended occurrence of sleep states (Figure 2A).

This nocturnal sleep period could include brief awake states as it is not unusual for a user to wake up once or twice during the night. However, our consideration toward the most extended sleep period, specifically during the night, may perversely overlook short sleep states that can occur between long awake states as in Figure 2B.

Thus, the system component requires solving an estimation problem to accurately identify both a user's bedtime (T_{sleep}) and wake times (T_{wake}) using the series of predicted binary states derived in Section 3.3. We compare two different estimation techniques for this problem: predicting with moving averages (i.e., MVA) and smoothed aggregation (i.e., AGG). Both prediction methods use the hypothesis that a user is more likely to be predicted in a sleep state when their past events assert that they were sleeping and vice versa. In Section 5.3, we compare the performance of these two different prediction and smoothing methods and found MVA worked best, thus was chosen for the final solution.

(1) *Predicting with Moving Averages:* When using MVA, *SleepMore* will provide sufficient weight to past states in the predictor to reduce the probability of predicting an awake state when a sudden burst of WiFi activity occurs amid a nocturnal sleep period (e.g., a background app notification). Instead, it will require a series of steadily rising WiFi network device activities to change the prediction from asleep to awake. Similarly, valuing the past awake states as an additional predictor would reduce the model predicting a sleep state when WiFi device activity briefly dips (e.g., the user leaves home for a short time).
(2) *Smoothed Aggregation:* Another approach is by applying smoothed aggregation (AGG) whereby we total the sum of predicted sleep states over a larger observation window. For example, by considering predicted sleep states over a 30 minutes window, we produce a range of prediction states that will likely contain a 95% confidence level of the user's state we are interested in. We smooth out the representation by applying a Savitzky-Golay (SG) filter [34] to determine the most extended sleep period. With this smoothing, even brief sleep episodes, as shown in Figure 2B, will be considered as part of the user's nocturnal sleep period. Then, we can determine the start of the most extended sleep period as T_{sleep} and the end of the period as T_{wake}.

Implementation:The outcomes from our ML model are fed into the estimation model that uses a 15-minute moving average window to identify the start and end of the sleep period, denoted as T_{sleep} and T_{wake}. The estimation model currently assumes only one sleep event per every 6 pm to 6 pm 24-hour period per user and selects the longest predicted sleep period within those 24 hours as the user's sleep duration.

With these implementations, Section 5.4.2 presents how *SleepMore* can run both the training and inference components on relatively cheap and resource-constrained Raspberry Pi embedded devices.

3.4 Anomaly & Noise Detection Module

Prior work has discussed how noise in WiFi-sensing, such as ping-ponging of AP connections, can negatively impact prediction performance [21]. In our case, the AP ping-pong effect is largely minimized by limiting WiFi network data collection to within users' residential premises. Our system will still be susceptible to noise-affected errors arising from background app running updates or receiving in-app notifications when the user is asleep. However, the estimation technique of predicting with moving average, as described in Section 3.3, efficiently handles these brief occurrences by assessing past states to correct the following states. Separately, there can be situations where a user is not at home for most of the evenings, resulting in sparse WiFi device activity. The repercussions of not having enough data for prediction will bring about spurious outcomes. These occurrences are uncommon but can happen every so often.

Thus *SleepMore* will benefit greatly from an anomaly detection module. By definition, an anomaly is a data point significantly different from the predictable pattern of data over a certain period of time. Two patterns can occur from monitoring a user's sleep; regular and irregular sleep patterns. A regular sleeper refers to those who habitually sleep and wake up at the same time (e.g., from midnight to 08:00 for 7–8 hours) [33], while the irregular sleeping pattern will change their sleep and wake up times by about 2 to 4 hours [37]. With this understanding, our anomaly detection must detect data anomalies that are significantly different from the expected prediction series of the user.

One method is to keep a simple moving average by calculating the average of the previous N data points. However, we should consider users changing their sleep-wake cycle due to everyday factors such as deadlines and parties. Thus, a more feasible approach is implementing an exponentially weighted moving average, which places more significance on the most recent bedtime and duration. Given this threshold, any prediction that is significantly larger than the confidence band (two standard deviations) of previous data will be regarded by *SleepMore* as an anomaly. Section 5.3.2 presents our results with these anomalies removed.

4 DATA COLLECTION STUDY

To evaluate *SleepMore*, we conducted a detailed data collection study, IRB approved, with all participants provided written informed consent before participating.

4.1 Procedure and Participants

This study involved 68 undergraduates in on-campus university dormitory housing and lasted for four weeks between March and May 2021. Participants met at the start of the study to collect various monitoring devices to collect baseline data, provide the MAC addresses of all their personally owned devices (phone, laptop, tablet), and attend a one-time in-lab clinical assessment to assess their sleep health. At the end of the study, the devices were returned, and each participant received compensation of up to USD 7.40 weekly in cash if they fulfilled all study requirements. These requirements include: (a) regularly wearing various monitoring devices and logging their sleep and health practices, (b) installing the required data collection apps on their smartphone, and (c) using the campus WiFi while in their dorms.

4.2 Baseline Sleep Data for Comparison

Each participant wore an Oura sleep sensing ring as baseline sleep data to compare against *SleepMore*. The Oura ring estimates sleep using heart rate, HRV (heart rate variability), body temperature, and movement via infrared photoplethysmography (PPG), temperature sensor, and a 3-D accelerometer signals [3]. Participants were instructed to wear the Oura ring (both during day and night) and sync the data to the Oura app daily. Before the study started, participants were asked to select an Oura ring size that was most comfortable for them. The Oura app was also installed on their mobile phones, and they were asked to run it daily to sync their sleep data to the cloud. From this cloud data, daily sleep measures such as bedtime, wake time, time-in-bed (TIB), wake after sleep onset (WASO) as well as thirty-second epoch by epoch sleep stages data (wake, light, deep, and REM) were extracted using Oura Health's cloud API. Note: we only use the Oura as a baseline and not as ground truth for medically approved sleep studies requiring highly invasive and expensive polysomnography.

4.3 Data Summary

WiFi Sensing of Multiple Devices: Throughout the study, participants were encouraged to utilize the campus WiFi frequently. We recorded all the MAC addresses of their personal smartphone (Android or iOS), laptop, and tablet. We use these MAC addresses to extract the RTLS data only for devices owned by study participants. Note: by default, the MAC addresses in the RTLS data are hashed. Thus, we only identify individual users after explicitly providing us with their MAC addresses.

Table 1 summarizes the data for our experiments in the next section. Two researchers reviewed the Oura baseline data and identified days with incorrect data to ensure data reliability. We then extracted WiFi network activity data for all user-owned devices for the days when their Oura baseline data was correct. Note: the distribution of sleep and awake states for each user is highly imbalanced as users spent only up to 20% of the day sleeping.

Users	68
Demographics	Age: mean: 22.5, std. dev.: 1.7 58 undergraduates, 10 graduates
Study Duration	4 weeks per user
Oura baseline data	14 - 24 days (mean: 17 days)
Oura sleep population summary	Bedtime: 12:00 am - 5:30 am (mean: 1:47 am) Wake time: 5:30 am - 1:15 pm (mean: 8:29 am) Sleep duration: 195 - 660 mins (mean: 401 mins)
% of time spent sleeping	≈ 12-20% of the day per participant

Table 1: Study data summary

5 EVALUATING *SLEEPMORE*

In this section, we evaluate the performance of *SleepMore* using the dataset collected from the data collection study. We ran to leave one out cross-validation for all our results and present accuracy, recall, precision, and F1-scores (weighted average of precision and recall).

5.1 Efficacy of ML Models

Our first evaluation compares the efficacy of different classification algorithms at predicting sleep and awake states in 24 hours. Specifically, we compare the performance of K-Nearest Neighbor (k=3), Decision Tree (DT), Random Forest (RF), and Gradient Boosting (XGBoost). As shown in Table 2, XGBoost yields the highest accuracy of 92.1% in classifying sleep states.

Algorithm	Accuracy	Precision	Recall	F1	p-value
KNN - gen	0.903	0.840	0.850	0.842	p>.1
KNN - per	0.904	0.842	0.850	0.844	
DT - gen	0.823	0.742	0.624	0.658	p<.01
DT - per	0.857	0.795	0.715	0.745	
RF - gen	0.848	0.774	0.707	0.731	p<.01
RF - per	0.886	0.821	0.806	0.811	
XGBoost - gen	0.905	0.861	0.827	0.839	p<.01
XGBoost - per	0.917	0.865	0.862	0.862	
XGBoost - tune	0.921	0.873	0.870	0.870	p<.05

Table 2: Model efficacy with different approaches.

5.1.1 Model Personalization. We then compared the difference between using generalized and personalized models where the personalized models was trained using 30% of the test user's data. Overall, the personalized models improve model performance by a small percentage. Next, we compared the efficacy of personalization with varying amounts of data, as shown in Table 3. Using 10% training data to build a per-user model improves model accuracy by 0.5% compared to a generalized model (p<.05). Increasing the training data to 20% improves the model accuracy by a slight 0.7% (p<.05). We found that using 30% of users' data achieves a good balance between training time and accuracy improvement as no significant improvement was seen beyond 30%.

Training Data	Accuracy	Precision	Recall	F1	p-value
0 (general)	0.905	0.861	0.827	0.839	-
10%	0.910	0.862	0.845	0.850	p<.05
30%	0.917	0.865	0.862	0.862	p<.05
50%	0.919	0.871	0.866	0.866	p>.1

Table 3: Training data for model personalization.

Model personalization also helps to improve recall – the probability that a sleep state is correctly predicted. Figure 5 illustrates the prediction outcomes for user P23 over a night. With 30% training data, recall and overall accuracy improved by 3% and 90% compared to a generalized approach (88%).

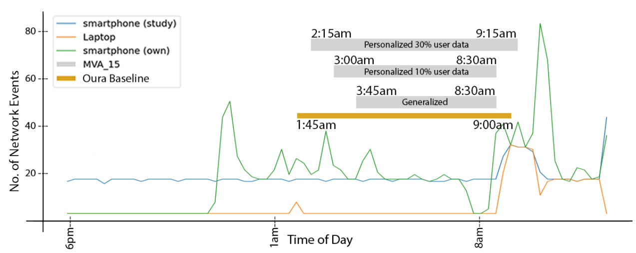

Figure 5: Personalization improves sleep prediction.

5.1.2 Sampling Frequency. Another key input to *SleepMore* is the sampling rate of WiFi network activity data. In Section 3.1, we explained that the RTLS data frequency ranges from every 5 seconds to several minutes depending on device use.

Frequency	Accuracy	Precision	Recall	F1	p-value
30-min	0.915	0.872	0.853	0.86	p>.1
15-min	0.917	0.865	0.862	0.862	-
1-min	0.920	0.871	0.869	0.868	p>.1
30-sec	0.918	0.867	0.865	0.865	p>.1

Table 4: Prediction sampling frequency for ML model.

Guided by prior work [21], we investigated WiFi data sampling rates of 30s, 1 min, 15 min, and 30 min. Our analysis, shown in Table 4, found that neither increasing nor reducing the sampling frequency affected the overall model performance. Increasing the sampling rate to either 1 min or 30s has a small statistically insignificant improvement over sampling every 15 or 30 mins. Thus, to reduce the amount of data *SleepMore* needs to collect (this helps make it easier to deploy), we used a WiFi sampling rate of 15 mins.

5.1.3 Model Tuning. The inner workings of our chosen classification algorithm XGBoost uses decision trees. Several essential parameters for tree-based classification algorithms are reducing the tree depth (i.e., max_depth) and lowering learning rate (i.e., eta) to improve overfitting - although doing this using results in more computationally expensive boosting. Increasing the number of trees (i.e., n_estimator) can improve performance. We iteratively optimized the hyperparameters of our model, which resulted in the following: *max_depth = 8*, *n_estimator = 200*, and *eta = 0.3*. Fine-tuning our model yields a 92.1% accuracy, which is statistically significant compared to the default XGBoost model (see Table 2).

Key Takeaway. In its present form, *SleepMore* predicts users' sleep states by employing a fine-tuned personalized (using 30% of a user's data) XGBoost classification model. We collected, on average, 23 days of data per user in our data collection study – thus, 1 week's worth of user data would be used for training. Our approach samples WiFi-generated features from multiple user devices every 15 mins. In practice, *SleepMore* does not need to consistently sample WiFi data for the entire day; instead it just needs to sample during likely sleeping periods every day (i.e., at the very least from 8 pm to the following morning on weekdays with afternoons added on weekends).

5.2 Benefit of Using Multiple Devices

A key hypothesis of this work is that using data collected from multiple devices owned by a user will lead to better sleep prediction that using just a single device. To test this, we compared the accuracy of using multiple devices versus just a single preferred primary device. A preferred device is the device (usually a smartphone) that is used the most [40].

Device Type	# Users with Device (%)	% WiFi activity/day
Smartphone	68 (100%)	47.0%
Smartphone (study)	25 (55.3%)	29.0%
Laptop	51 (75%)	36.7%
Tablet	20 (36.8%)	34.8%

Table 5: Multi-device connection to WiFi per day.

Table 5 summarizes the percentage of time different user-owned devices were connected to the WiFi network. In particular, smartphones were owned by 100% of our users and connected to the monitored WiFi network 47% of the time. Note: the monitored network was only in the dorms; users were probably elsewhere the rest of the time. Laptops (owned by 75% of our participants) were connected 36.7% of the time.

5.2.1 Performance of Multi-Device Features. Figure 6 shows the importance of all features based on our model's fitted trees. The F-score for feature importance is measured in terms of weight, that is, the number of times the feature is used in a tree. Intuitively, a high F-score (1 being the max) reflects how important the feature is independent of other features. This analysis revealed that using the sum of network events across all user-owned devices had the highest F-score compared to using events only from smartphones or tablets.

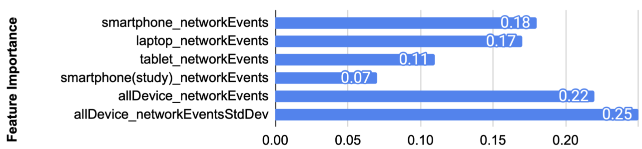

Figure 6: Feature importance.

Table 6 compares the model performance with different device features. Using just smartphones yields a 78.9% recall and 89.3% accuracy. However, adding other device features increases recall by ≈ 10% and overall accuracy by 3% ($p<.01$).

Feature	Accuracy	Precision	Recall	F1	p-value
Smartphone	0.893	0.860	0.789	0.812	p<.01
Multiple devices	0.921	0.873	0.870	0.870	-

Table 6: Performance with different device features.

Key Takeaway. The results strongly suggest that using multiple devices will improve *SleepMore*'s accuracy and recall.

5.3 Efficacy of Sleep Estimation

With the classification component determined, *SleepMore* now has to interpret the classification results. In particular, given a continuous prediction of sleep and awake states, *SleepMore* must now accurately predict the start and end of a user's sleep period (T_{sleep}, T_{wake}), and the sleep duration?

# Devices	T_{sleep} mins	T_{wake} mins
Smartphone	42 (p<.01)	38 (p<.01)
Smartphone + Laptop/Tablet	37 (p>.05)	31 (p>.05)
Multiple devices	36	28

Table 7: Errors from predicting with moving average.

Table 7 tabulates the mean error in estimating a user's sleep period. *SleepMore* achieves a mean error of 36 mins and 28 mins in predicting bedtime and wake time with multiple device sensing. Using multiple devices over a single device achieves better estimations of up to 10 mins in wake time ($p<.01$). Using two devices compared to a single device reduces the prediction error for bedtime by 5 mins, and wake time by 7 mins ($p<.05$), but is marginally significant compared to multi-device ($p=.09$). We now investigate factors that affect these results.

5.3.1 Effect of Window Size. In Section 3.3, we described two techniques to estimate sleep timing: predicting with moving averages (MWA) and smoothed aggregation (AGG). MVA determines a 'sleep' or 'wake' state by gradually moving the window, defined by the threshold interval, over the predicted outcomes in single increments. Then, we determine the most extended sleep period from the revised outcomes. In AGG, we total the sum of every few sleep states, defined by the threshold interval, and smooth out the representation by applying a Savitzky-Golay (SG) filter [34]. Figure 7 charts the errors of T_{sleep} and T_{wake} in minutes, calculated as the difference in time between our prediction and the users' Oura ring sleep baseline reference.

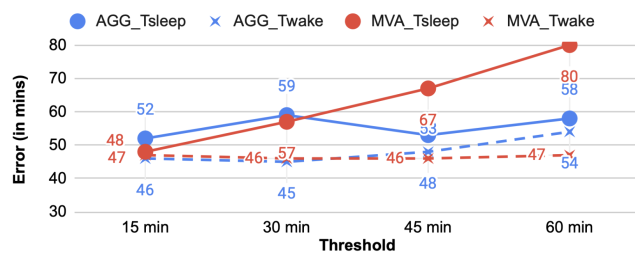

Figure 7: Error at varying thresholds/technique.

We observed that using MVA with a rolling window size of 15 mins yielded the best results with a 48 and 47 mins error in estimating T_{sleep} and T_{wake}, respectively, with other techniques and window sizes producing higher errors.

5.3.2 Error Analysis. To better understand the source of these prediction errors, Table 8 provides the full statistics for the MVA and AGG prediction errors using a 15 mins window size. We observed that the max values and stdev. in all cases were very high (up to 450 mins) suggesting that outlier detection and removal would significantly improve the prediction results. A probability density function view of the sleep time errors (Figure 8 also suggested that the outliers were a small fraction of the predictions.)

From removing the outliers, specifically those errors that fell beyond the Upper Inner Fence (UIF) as the threshold, we removed about 17% of the predicted data and reduced our go-to-bed and wake time errors to 36 and 28 mins.

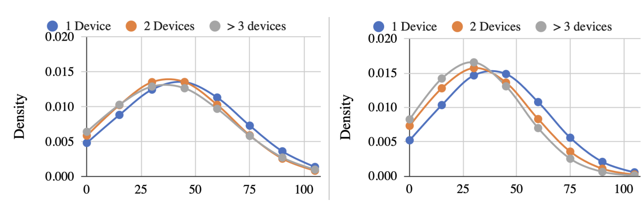

Figure 8: PDF of bedtime and wake time errors.

	Pred. w Moving Average		Smoothed Aggregation	
	T_{sleep}	T_{Wake}	T_{sleep}	T_{wake}
Median	30	30	30	30
Mean	48	47	52	46
Max	330	345	450	345
Min	0	0	0	0
Mode	15	15	15	15
Stdev.	47	50	56	48
Q1, Q3	15, 60	15, 60,	15, 75	15, 60
UIF, UOF	127.5, 150	105, 150	165, 255	127.5, 195

Table 8: Summary statistics of sleep time error (mins).

	Oura	*SleepMore*		Oura	*SleepMore*
Median	405	420	**Mean**	401	410
Max	660	570	**Min**	195	180
Mode	420	405	**Stdev.**	67	55
Q1	360	375	**Q3**	450	450
UIF	585	562	**UOF**	720	675

Table 9: Sleep duration descriptive statistics (mins).

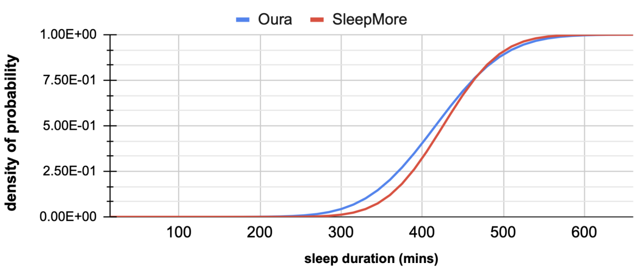

Figure 9: Sleep duration CDF by Oura and *SleepMore*.

Table 9 summarizes the average descriptive statistics for the sleep durations measured by both the Oura ring and predicted by *SleepMore* for the data collection study participants. The differences in these two distributions was statistically insignificant (with p>.1). In particular, both modalities measured that 50% of the participants sleep for at least 400 mins per day while 10% of them slept for at least 500 mins (also shown in Figure 9).

5.3.3 Characterizing the Outliers. We further analyzed the outliers to understand what caused them to appear. First, we found many outliers caused by predictions made with sparse WiFi data. For example, Figure 10 illustrates the sleep estimation outcomes for participant P59 over two different days, where Day 1 represents an extreme outlier and Day 2 represents the desired monitoring outcome. *SleepMore* uses only WiFi data as its sole data source to make predictions. Thus, it is possible that users may not be present in their residential premises or not always be connected to the WiFi network, and this can lead to sparse/inconsistent data on some days.

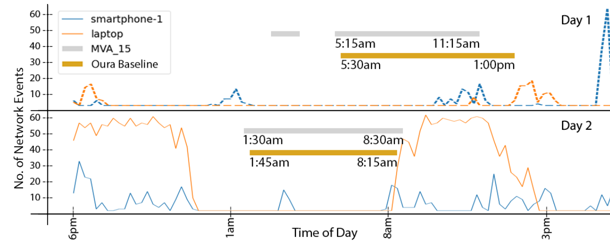

Figure 10: Larger errors from sparse WiFi data.

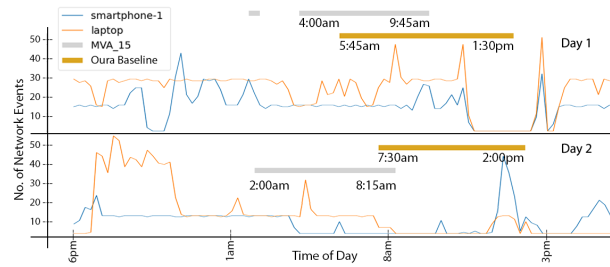

Figure 11: Larger errors from the lack of training data.

Another reason for outliers is the lack of training data. For example, as shown in Figure 11, *SleepMore* produces significantly inaccurate predictions, especially on Day 1 and 2 for P11. On these days, P11 displayed an entirely irregular sleep behavior – sleeping at 5:45 and 7:30 am and waking up at 1:30 and 2:00 pm, respectively. It is possible that the lack of training data, especially users sleeping at 5:45 am onwards, had affected the results of these days. We counted only 5 samples (equivalent to less than 1%) of users who slept beyond 5:45 am and 1 sample beyond 7:30 am. As a result, our model had not learned to predict these outcomes. In all (5) similar cases, bedtime error is within 105-330 mins and wake time error is within 75-345 mins.

Key Takeaway. Indeed, a significant challenge foreseen in our approach is determining a user's bedtime and wake time from monitoring coarse-grained WiFi features. Our estimation method values sleep states made in the last 15 mins to predict each user's upcoming sleep/awake state and takes the most extended sleep period as the nocturnal sleep duration. This approach assumes that users are present in their homes and connected to their home WiFi at most parts of their evenings; otherwise, sleep prediction can be negatively impacted by sparse WiFi device activity. Separately, our study currently lacks samples of users sleeping at different times of the day, thus had affected some outcomes where users were just beginning to sleep in the morning. From removing these outliers, our approach produced between 36 and 28 mins bedtime and wake time error, depending on the number of tracked devices.

5.4 Comparing Against Prior Work

This section shows how *SleepMore* compares against prior work that uses similar methods in terms of accuracy and system scalability. In particular, the prior work also uses WiFi connection data obtained directly from the WiFi infrastructure. The comparison was made against three Bayesian change point detection methods using WiFi device activity from all devices (i.e., *all_networkEvents* feature) ; normal prior [9], hierarchical prior [7], and an ensemble of normal, uniform, and hierarchical priors [21].

5.4.1 Accuracy Comparison. Prior works on sleep detection mentioned in [21] rely on Bayesian change point detection techniques that predict the bedtime and wake-up time, looking at the rate of change of values in a single variable time-series data. Techniques using only hierarchical prior or a normal prior are useful when users have regular sleep patterns, and the data is not subjected to too much noise. However, basic profiling of users is required to best decide on these priors. An ensemble model accommodates the irregular sleepers, including a uniform prior model, the normal prior models, and the hierarchical prior model.

Technique	Acc	Prec	Rec	F1	Bed/Wake Error
SleepMore	0.921	0.873	0.870	0.870	36,28
Ensemble based	0.787	0.586	0.891	0.707	139, 175
Norm. Prior	0.799	0.600	0.919	0.726	190, 109
Hier. Prior	0.814	0.622	0.914	0.740	193 ,111

Table 10: Comparison with prior techniques.

Table 10 summarizes the performance comparison and shows that *SleepMore* achieves significantly better accuracy, recall, and precision (p<.01). A plausible explanation for the change point detection models failing is the absence of location information. Specifically, the amount of WiFi network activity we calculate every 15 mins does not consider changes between places and only checks for a coarse residential location. The above change point detection techniques are suitable in situations where we have no access to the training data and want a high-level understanding of users' behavior in a residential location. However, in situations where we want to do more fine-grained analysis and have access to training data from multiple devices, *SleepMore* can provide more accuracy.

5.4.2 System Scalability. Figure 12 compares the time taken for *SleepMore* and prior techniques to predict the sleep patterns as the number of users increases. The processing time for *SleepMore* is 1 to 2 orders of magnitude lower than prior work and scales linearly as the number of users increases. In particular, *SleepMore* can predict the sleep times of 20 users in about 30s using unoptimised PyMC3 [10, 21] libraries. This suggests that *SleepMore* is a viable solution for home and dorm monitoring where the number of users per area does not usually exceed 30 people. Finally, *SleepMore* uses WiFi data that is easily accessible from many different commercial and home WiFi networks and thus does not need any custom hardware or intricate setup to become operational.

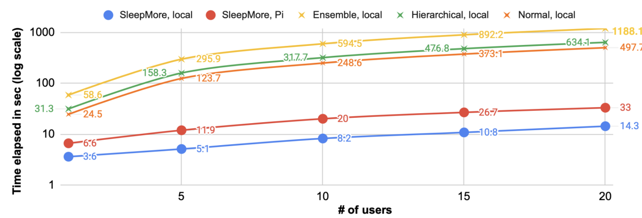

Figure 12: Prediction time for users.

Key Takeaway. While earlier work similarly utilizes WiFi connection data to predict sleep, two main differences set our work apart: First, these works relied solely on single-device monitoring through users' smartphones. Second, change point detection is employed in prior work to determine sleep from changes in WiFi event rates in different locations.

5.5 Robustness Across User & Sleep Types

Our final evaluations seeks to understand how robust *SleepMore* is to different types of users and sleep characteristics.

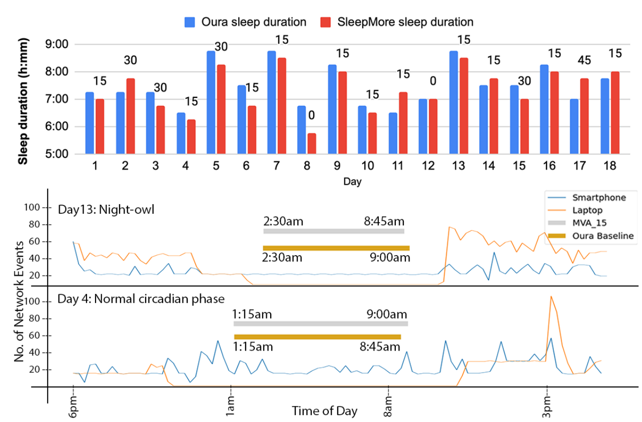

Figure 13: Comparing Oura and *SleepMore* for P4. Top charts sleep duration difference in mins.

For each participant, given their set of T_{sleep} and T_{wake} times across multiple days, we can determine the nocturnal sleep duration for that participant. Figure 13 charts participant P4, who typically slept at 1:25 am (stdev. 36 mins) every night, corresponding to a "normal" circadian phase. For the entire 18 days that P4 provided useful Oura data, *SleepMore* predicts them getting approximately 7 hours of sleep on average, thus maintaining a regular sleep pattern [33], such as on Day 4. This includes Day 13 when they significantly shifted their usual sleep time – even so, *SleepMore* predicted their sleep duration with less than 15 mins of error

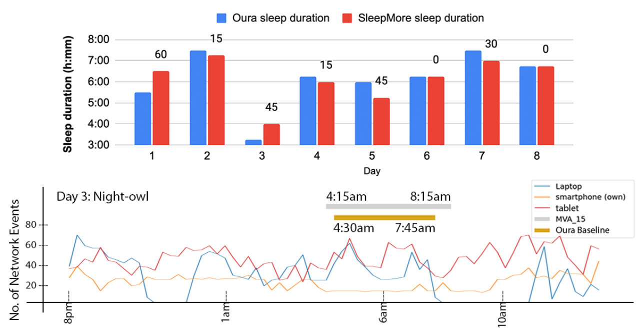

Figure 14: Night owl sleep schedule by P41, identified to maintain an irregular sleep schedule.

A separate example is participant P41, who typically slept at 2:10 am (stdev. 1h 10 mins). On average, participant P41 received 6 hours (stdev. 1 h 35 mins) of sleep during the study period. Unlike participant P4, we identified participant P41 as an irregular sleeper [37]. On Day 3, participant P41 displayed a significant shift in their sleeping pattern, and started resembling a habitual "night owl" sleep schedule [23] while receiving only 3h 25 mins of sleep. Even with this large shift, *SleepMore* predicted their sleep duration with less than 45 mins of error. Note: the 60 mins error on Day 1 is the sum of both bedtime and wake time errors.

Key Takeaway. These examples underscore the robustness of our model in measuring sleep patterns comparable to the Oura ring. The prediction outcomes of bedtime, wake time, and sleep duration are telling of sleep cycles and can provide a fundamental understanding of sleep hygiene at the aggregated and individual level.

6 DISCUSSION

Our study's objectives were to develop a sleep prediction solution that can accurately predict sleep from monitoring WiFi device activity for multi-user devices. Here we discuss the implications of our findings.

Improvements to Handle Corner Cases. In situations where Oura predicted a user to sleep before WiFi device activity data is monitored to decrease, our estimation technique will delay predicting the user as asleep. Figure 15 exemplifies an error our system encounters that could be attributed to multiple variables and conditions – it is possible that background activity was running when the user slept, but it could equally be possible that the Oura was producing erroneous readings. Despite removing outliers, this is one example that resulted in large errors. Our work continues to explore more sophisticated techniques to handle these corner cases better.

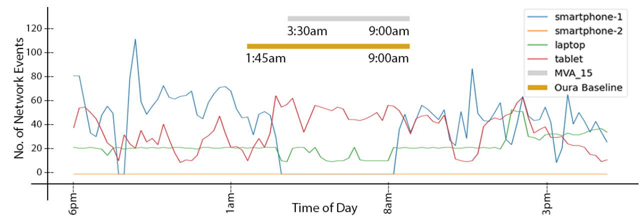

Figure 15: Outlier with large error.

Assessing Type of Sleep Characteristics. We provide evidence of our technique producing key sleep characteristics comparable to the Oura ring, such as bedtime, wake time, and sleep duration. A clear limitation of SleepMore is the inability to differentiate the four sleep stages (i.e., Deep, Light, REM, wake) as obtainable by the Oura ring. At present, the training for our model is notably restricted to nocturnal sleep cycles. With SleepMore predicting sleep states every 15 minutes, we believe the system can be extended to conceivably detect different sleep patterns, including intermittent wake-ups, sleeping in the morning, or taking daytime naps.

7 CONCLUSION & FUTURE WORK

This work proposed *SleepMore* as a promising and practical solution to provide accurate and easy-to-deploy sleep monitoring in homes, dormitories, and other environments. *SleepMore* uses a supervised learning and estimation model that leverages WiFi device activity, collected directly from the WiFi infrastructure, from multiple user devices to predict sleep. It does this in two steps; First, it determines if a user is in a sleep or awake state every 15 minutes using an XGBoost personalized model. Second, it processes these sequences of sleep and awake, using a moving average to estimate the user's bedtime and wake times. Our validation uses data collected from 68 participants living in on-campus dormitories and showed that *SleepMore* could predict sleep states with 92% accuracy, and determine the bedtime and wake times with errors within 36 minutes and 28 minutes, respectively. A detailed error analysis highlighted the importance of an anomaly detector to handle spurious data caused by inherently noisy coarse-grained WiFi data. We demonstrated the robustness of *SleepMore* at predicting sleep measures across different users with different sleep patterns and regularities. We showed that *SleepMore* significantly outperforms prior work in this area. As future work, we plan to incorporate *SleepMore* into a scalable solution for on-campus health and behavioral risk surveillance to help college students at risk of sleep deprivation and related issues.

REFERENCES

- [1] Saeed Abdullah, Mark Matthews, Elizabeth L Murnane, Geri Gay, and Tanzeem Choudhury. 2014. Towards circadian computing: " early to bed and early to rise" makes some of us unhealthy and sleep deprived. In *Proceedings of the 2014 ACM international joint conference on pervasive and ubiquitous computing*. 673–684.
- [2] Bruce M Altevogt, Harvey R Colten, et al. 2006. Sleep disorders and sleep deprivation: an unmet public health problem. (2006).
- [3] Marco Altini and Hannu Kinnunen. 2021. The promise of sleep: A multi-sensor approach for accurate sleep stage detection using the oura ring. *Sensors* 21, 13 (2021), 4302.
- [4] American College Health Association. 2021. American College Health Association-National College Health Assessment III: Undergraduate Student Reference Group Data Report Fall 2020. (2021).
- [5] K Bartel, R Scheeren, and M Gradisar. 2019. Altering adolescents' pre-bedtime phone use to achieve better sleep health. *Health communication* 34, 4 (2019), 456–462.
- [6] Zhenyu Chen, Mu Lin, Fanglin Chen, Nicholas D Lane, Giuseppe Cardone, Rui Wang, Tianxing Li, Yiqiang Chen, Tanzeem Choudhury, and Andrew T Campbell. 2013. Unobtrusive sleep monitoring using smartphones. In *2013 7th International Conference on Pervasive Computing Technologies for Healthcare and Workshops*. IEEE, 145–152.
- [7] Andrea Cuttone, Per Bækgaard, Vedran Sekara, Håkan Jonsson, Jakob Eg Larsen, and Sune Lehmann. 2017. Sensiblesleep: A bayesian model for learning sleep patterns from smartphone events. *PloS one* 12, 1 (2017), e0169901.
- [8] Massimiliano De Zambotti, Nicola Cellini, Aimee Goldstone, Ian M Colrain, and Fiona C Baker. 2019. Wearable sleep technology in clinical and research settings. *Medicine and science in sports and exercise* 51, 7 (2019), 1538.
- [9] Yassine El-Khadiri, Gabriel Corona, Cédric Rose, and François Charpillet. 2018. Sleep Activity Recognition Using Binary Motion Sensors. In *2018 IEEE 30th International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 265–269.
- [10] Daniel Emaasit et al. 2018. Pymc-learn: Practical probabilistic machine learning in Python. *arXiv preprint arXiv:1811.00542* (2018).
- [11] Liese Exelmans and Jan Van den Bulck. 2016. Bedtime mobile phone use and sleep in adults. *Social Science & Medicine* 148 (2016), 93–101.
- [12] Anja Friedrich and Angelika A Schlarb. 2018. Let's talk about sleep: a systematic review of psychological interventions to improve sleep in college students. *Journal of Sleep Research* 27, 1 (2018), 4–22.
- [13] Weixi Gu, Longfei Shangguan, Zheng Yang, and Yunhao Liu. 2015. Sleep hunter: Towards fine grained sleep stage tracking with smartphones. *IEEE Transactions on Mobile Computing* 15, 6 (2015), 1514–1527.
- [14] Tian Hao, Guoliang Xing, and Gang Zhou. 2013. isleep: Unobtrusive sleep quality monitoring using smartphones. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*. 1–14.
- [15] Jing-wen He, Zhi-hao Tu, Lei Xiao, Tong Su, and Yun-xiang Tang. 2020. Effect of restricting bedtime mobile phone use on sleep, arousal, mood, and working memory: a randomized pilot trial. *PloS one* 15, 2 (2020), e0228756.
- [16] Max Hirshkowitz, Kaitlyn Whiton, Steven M Albert, Cathy Alessi, Oliviero Bruni, Lydia DonCarlos, Nancy Hazen, John Herman, Eliot S Katz, Leila Kheirandish-Gozal, et al. 2015. National Sleep Foundation's sleep time duration recommendations: methodology and results summary. *Sleep health* 1, 1 (2015), 40–43.
- [17] Chen-Yu Hsu, Aayush Ahuja, Shichao Yue, Rumen Hristov, Zachary Kabelac, and Dina Katabi. 2017. Zero-effort in-home sleep and insomnia monitoring using radio signals. *Proceedings of the ACM on Interactive, mobile, wearable and ubiquitous technologies* 1, 3 (2017), 1–18.
- [18] Zhengjie Huang, Wenjin Wang, and Gerard de Haan. 2021. Nose Breathing or Mouth Breathing? A Thermography-Based New Measurement for Sleep Monitoring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 3882–3888.
- [19] Violaine Kubiszewski, Roger Fontaine, Emmanuel Rusch, and Eric Hazouard. 2014. Association between electronic media use and sleep habits: An eight-day follow-up study. *International Journal of Adolescence and Youth* 19, 3 (2014), 395–407.
- [20] Clete Kushida. 2012. *Encyclopedia of sleep*. Academic Press.
- [21] Priyanka Mary Mammen, Camellia Zakaria, Tergel Molom-Ochir, Amee Trivedi, Prashant Shenoy, and Rajesh Balan. 2021. WiSleep: Scalable Sleep Monitoring and Analytics Using Passive WiFi Sensing. arXiv:2102.03690 [eess.SP]
- [22] Stijn AA Massar, Xin Yu Chua, Chun Siong Soon, Alyssa SC Ng, Ju Lynn Ong, Nicholas IYN Chee, Tih Shih Lee, Arko Ghosh, and Michael WL Chee. 2021. Trait-like nocturnal sleep behavior identified by combining wearable, phone-use, and self-report data. *NPJ digital medicine* 4, 1 (2021), 1–10.
- [23] Luciano Mecacci and Alberto Zani. 1983. Morningness-eveningness preferences and sleep-waking diary data of morning and evening types in student and worker samples. *Ergonomics* 26, 12 (1983), 1147–1153.
- [24] Jun-Ki Min, Afsaneh Doryab, Jason Wiese, Shahriyar Amini, John Zimmerman, and Jason I Hong. 2014. Toss'n'turn: smartphone as sleep and sleep quality detector. In *Proceedings of the SIGCHI conference on human factors in computing systems*. 477–486.
- [25] A. Networks. [n.d.]. RTLS - integrating with the RTLS data feed. https://community.arubanetworks.com/aruba/attachments/aruba/unified-wired-wireless-access/23715/1/RTLSintegrationv6.docx. Accessed: 2021-11-07.
- [26] Maren Nyer, Amy Farabaugh, Kiki Fehling, David Soskin, Daphne Holt, George I Papakostas, Paola Pedrelli, Maurizio Fava, Angela Pisoni, Ottavio Vitolo, et al. 2013. Relationship between sleep disturbance and depression, anxiety, and functioning in college students. *Depression and anxiety* 30, 9 (2013), 873–880.
- [27] Oura Health Oy. [n.d.]. New Oura Ring Generation 3. https://ouraring.com
- [28] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, Oct (2011), 2825–2830.
- [29] Pew Research Center. April 7, 2021. Internet/Broadband Fact Sheet. https://www.pewresearch.org/internet/fact-sheet/internet-broadband/#who-has-home-broadband
- [30] Pew Research Center. April 7, 2021. Mobile Fact Sheet. https://www.pewresearch.org/internet/fact-sheet/mobile/#ownership-of-other-devices
- [31] Pew Research Center. July 16, 2021. Home broadband adoption, computer ownership vary by race, ethnicity in the U.S. https://www.pewresearch.org/fact-tank/2021/07/16/home-broadband-adoption-computer-ownership-vary-by-race-ethnicity-in-the-u-s/
- [32] Pew Research Center. June 3, 2020. Experts Predict More Digital Innovation by 2030 Aimed at Enhancing Democracy. https://www.pewresearch.org/internet/2020/06/30/experts-predict-more-digital-innovation-by-2030-aimed-at-enhancing-democracy/
- [33] Andrew JK Phillips, William M Clerx, Conor S O'Brien, Akane Sano, Laura K Barger, Rosalind W Picard, Steven W Lockley, Elizabeth B Klerman, and Charles A Czeisler. 2017. Irregular sleep/wake patterns are associated with poorer academic performance and delayed circadian and sleep/wake timing. *Scientific reports* 7, 1 (2017), 1–13.

[34] William H Press and Saul A Teukolsky. 1990. Savitzky-Golay smoothing filters. *Computers in Physics* 4, 6 (1990), 669–672.

[35] Tauhidur Rahman, Alexander T Adams, Ruth Vinisha Ravichandran, Mi Zhang, Shwetak N Patel, Julie A Kientz, and Tanzeem Choudhury. 2015. Dopplesleep: A contactless unobtrusive sleep sensing system using short-range doppler radar. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 39–50.

[36] Mary E Rosenberger, Matthew P Buman, William L Haskell, Michael V McConnell, and Laura L Carstensen. 2016. 24 hours of sleep, sedentary behavior, and physical activity with nine wearable devices. *Medicine and science in sports and exercise* 48, 3 (2016), 457.

[37] John M Taub. 1978. Behavioral and psychophysiological correlates of irregularity in chronic sleep routines. *Biological Psychology* 7, 1-2 (1978), 37–53.

[38] Roxana Tiron, Graeme Lyon, Hannah Kilroy, Ahmed Osman, Nicola Kelly, Niall O'Mahony, Cesar Lopes, Sam Coffey, Stephen McMahon, Michael Wren, et al. 2020. Screening for obstructive sleep apnea with novel hybrid acoustic smartphone app technology. *Journal of Thoracic Disease* 12, 8 (2020), 4476.

[39] Siao Hui Toh, Erin K Howie, Pieter Coenen, and Leon M Straker. 2019. "From the moment I wake up I will use it… every day, very hour": a qualitative study on the patterns of adolescents' mobile touch screen device use from adolescent and parent perspectives. *BMC pediatrics* 19, 1 (2019), 1–16.

[40] Alexander JAM Van Deursen, Colin L Bolle, Sabrina M Hegner, and Piet AM Kommers. 2015. Modeling habitual and addictive smartphone behavior: The role of smartphone usage types, emotional intelligence, social stress, self-regulation, age, and gender. *Computers in human behavior* 45 (2015), 411–420.

[41] Bohan Yu, Yuxiang Wang, Kai Niu, Youwei Zeng, Tao Gu, Leye Wang, Cuntai Guan, and Daqing Zhang. 2021. WiFi-Sleep: Sleep Stage Monitoring Using Commodity Wi-Fi Devices. *IEEE Internet of Things Journal* (2021).

[42] Shichao Yue, Yuzhe Yang, Hao Wang, Hariharan Rahul, and Dina Katabi. 2020. BodyCompass: Monitoring sleep posture with wireless signals. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–25.

[43] Bing Zhai, Ignacio Perez-Pozuelo, Emma AD Clifton, Joao Palotti, and Yu Guan. 2020. Making sense of sleep: Multimodal sleep stage classification in a large, diverse population using movement and cardiac sensing. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 2 (2020), 1–33.