DeepSleep

A sensor-agnostic approach towards modelling a sleep classification system

by

Shashank P. Rao

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday November 15, 2018 at 2:15 PM.

Student number:	4624017	
Project duration:	Oct 1, 2017 – November 15, 2018	
Thesis committee:	Prof. dr. Alan Hanjalic,	TU Delft, Professor
	Dr. H. S. Hung,	TU Delft
	Dr. P. S. Cesar Garcia,	TU Delft, supervisor
	Dr. Abdallah El Ali,	CWI, supervisor

This thesis is confidential and cannot be made public until December 31, 2018.

Abstract

Sleep is a natural state of our mind and body during which our muscles heal and our memories are consolidated. It is such a habitual phenomenon that we have been viewing it as another ordinary task in our day-to-day life. However, owing to the current fast-paced, technology-driven generation, we are letting ourselves be sleep-deprived, giving way to serious health concerns such as depression, insomnia, restlessness, apnea and Alzheimer's. Polysomnography (PSG) studies are used for diagnosing and treating sleep-related disorders. Although the PSG studies are considered as the gold standard, they are obtrusive and do not allow for long-term monitoring. Various wearables have been manufactured to help people monitor their sleep-health. However, these devices have been shown to be inaccurate.

The ubiquitous sensor technology employed by the wearables provides large volumes of data, recorded in the most natural setting of the user. There is an opportunity to make use of the highly available sensor data to model a sleep scoring system that could help individuals monitor their sleep-health from the comfort of their home. In this thesis, we aim to alleviate this problem by attempting to bridge the gap between the highly accurate but obtrusive medical diagnosis (PSG) and the non-intrusive yet inaccurate wearables.

In this work, we propose *DeepSleep*, a deep neural net-based sleep classification model using an unobtrusive BCG-based heart sensor signal. Our proposed model's architecture uses the combination of CNN and LSTM layers to perform self-feature extraction and sequential learning respectively. We show that our model can classify sleep stages with a mean f1-score of 74% using the BCG signal. We employ a 2-phase training strategy to build a pre-trained model to tackle the limited dataset size and test the transferability of the model on other types of heart-signal. With an average classification accuracy of 82% and 63% using ECG and PPG based heart signal respectively, we show that our pre-trained model can be used in the transfer learning setting as well. Lastly, with the help of a user study of 16 subjects, we show that the objective sleep quality metrics correlate with the perceived sleep quality reported by the subjects with a correlation score of $r = 0.43$.

Although our proposed model's performance is not yet comparable to the medical standards, we show that it is possible to monitor our sleep-health using the wearable signals with the least domain knowledge and preprocessing techniques. The prediction and performance of our *DeepSleep* model shows that it is able to learn the biological rules of sleep wherein it always follows a *Deep* or *REM* stage with a transitional *Light* stage. Our model treats the classification problem sequentially, thus, identifying important sleep parameters like onset of sleep cycles and time spent in different sleep stages which are time-dependant factors. Furthermore, our user study, conducted using the SATED questionnaire, provides an insight into the difference in user's perceived sleep quality and model's estimation. It shows that an automated classification system needs to incorporate various external factors such as environmental and ambient conditions to be able to strongly correlate with the perceived or subjective quality. We further discuss the future research gaps and opportunities that could improve the model's performance and also extend it to other domains like irregular heart-beat and apnea detection. We consider this work to be a starting point for research into sleep and heart health using non-intrusive wearable sensors and deep neural network-based architectures.

Preface

Two years ago, when I started the Computer Science MS programme at TU Delft, I was only carrying passion and enthusiasm for machine learning but with almost no theoretical or technical skills in this field. Yet the extraordinary teaching skills of some of the professors here at the Delft University of Technology have managed to steer my interests in directions I could not have foreseen. Not only did they do an amazing job of explaining the intuition and nuances behind topics, making even the most complex of them seem simple, they also somehow managed to make those topics interesting, challenging and, at times, even fun. They showed me how important it is to have a thorough understanding of a subject from a research and technical point-of-view. But most importantly, they gave me the confidence that I could learn and master things that I could not even imagine before.

My special thanks go out to Pablo Cesar and Abdallah Ali, my thesis supervisors, for guiding me through the overwhelming number of possible directions in which this topic could have taken me. Without your constant guidance, patience, support and feedback I would certainly have lost sight of the big picture. But more importantly, I would want to truly thank you, Pablo, for putting up with me during the entirety of the thesis year and acknowledging the human side of writing a thesis. Working at CWI with you all has been one of the most memorable days for me during this period. The stress-free environment at the CWI-DIS group provided me with a different outlook towards cross-domain researches and teamwork. My thesis journey has been a series of ups and down days. Thanks to yours and Abdo's constant moral support, I can surely say that I will be ending it on a high.

I would also like to give special thanks to my friends at TurtleShell Technologies (Dozee) and NIMHANS. It would have been very difficult to complete this work without your continuous help with the dataset and surveys. Thank you Gaurav and Mudit for bearing my sudden (and sometimes, outrageous) demands for new sleep devices and data. During the course of my thesis, I have realized how important and challenging it is to build a product in the healthcare domain where precision and quality is the highest priority. I feel really proud to have been a part of your team and I am looking forward to witnessing Dozee creating revolution in the healthcare sector.

Finally, I would like to thank my friends, family and faculty of MMC group who helped me through those moments in which nothing made sense. Thank you for patiently listening to my rants about my work, eventually putting you all to sleep! I would be nowhere without you all.

Shashank P. Rao
Delft, November 2018

Contents

List of Figures

List of Tables

Acronyms

AASM American Academy of Sleep Medicine. 8, 9, 28, 32

BCG BallistoCardiography. ix, xi, 2, 3, 5, 21, 27, 29, 34, 41–44, 47, 50, 51, 59

bi-LSTM Bidirectional Long-Short Term Memory. 5, 16, 37, 63

CDC Centers for Disease Control and Prevention. 1

CNN Convolutional Network. 5, 39

ECG Electro-Cardiogram. ix, xi, 2, 4, 5, 16, 19, 29, 41, 42, 47, 51, 59

EEG Electroencephalogram. 2, 3, 8, 9, 16, 50, 51, 58

EMG ElectroMyogram. 2, 16, 50, 58

EOG Electro-Occulogram. 2, 8, 16, 50, 58

HRV Heart-Rate Variability. xi, 4, 10–12, 16, 44, 51, 52

NIMHANS National Institute of Mental Health and Sciences. 5, 28

PPG Photo-plethysmography. 2, 3, 5, 41, 42, 47, 62

PSG Polysomnography. 1–4, 7, 50, 59

R&K Rechtschaffen & Kales. 3, 8, 9, 28, 32

RIP Respiratory Inductive Plethysmography. 16

RNN Recurrent Neural Network. 61

1

Introduction

Sleep is the most naturally learned activity for any living being. It just comes natural to us. We have been studying a lot about sleep and kept improving our sleeping environment from grasslands to foam mattresses. Interestingly, sleep has also been culturally nuanced. Every culture has had its own interpretation of sleep. Italians have been known to take siestas to have better sleep, Indians believed in sleeping in a specific direction and posture and Japanese believe in the art of power naps called Inemuri. We understand that sleep is important which is also evident from the time we spend sleeping. We spend almost one-third of our lives sleeping - which is around 26 years for an average human being. At the same time, we spend almost seven years just trying to sleep [27], tossing and turning in bed. We understand the importance of sleep and yet we fail to understand how we can improve it and lead a more healthy lifestyle.

The increasing use of technology, and other habits have driven us towards an unhealthy lifestyle. In today's fast-paced, always-connected, and sleep-deprived world, our need for a good night's sleep is more important – and elusive – than ever. *Sleep deprivation* has become synonymous to our generation. Ariana Huffington quite eloquently describes the importance of good restorative sleep in her book, *The Sleep Revolution* [57]:

> *These two threads that run through our life — one pulling us into the world to achieve and make things happen, the other pulling us back from the world to nourish and replenish ourselves — can seem at odds, but in fact they reinforce each other.*

According to a study conducted by the Centers for Disease Control and Prevention (CDC) in the USA in 2009 [41], around 35.3% of a total of 74,751 people reported having less than 7 hours of sleep on average during a 24-hour period. 48% reported having snoring problems, 37.9% reported unintentionally falling asleep during the day, and 4.7% reported nodding off while driving. These numbers should not be underestimated because studies on the effect of sleep deprivation on our health reveal some disturbing findings. A recent study by UCLA [94] showed that sleep deprivation disrupts our brain cells' ability to communicate with each other, causing temporary mental lapses, affecting memory and visual perception. Another study [7] revealed that two consecutive nights of less than 6 hours of sleep could make us highly inefficient and unproductive for a whole week. Hence, the importance of a good amount of sleep in our lifestyle cannot be ignored. These are just the lighter consequences of inadequate sleep. Severe cases of sleep deprivation have shown to cause depression, insomnia, restlessness, and Alzheimer's [36, 72, 124].

Increased occurrences of sleep-related disorders have led users to resort to consult doctors and sleep experts for diagnosing their disorders. Sleep clinics use Polysomnography (PSG) to diagnose and treat sleep disorders [77]. With the help of the data recorded from a set of electrodes attached to a subject, an expert scores the sleep stages visually and analyzes the sleep pattern to confirm the prognosis. Sometimes, the expert monitors the subject's movements through a camera during the study. This way of diagnosing sleep has raised privacy and comfort-related concerns among the. The complexity of recording, inconvenience, lack of privacy and monetary concerns has led to a growing reluctance from considering medical help for sleep problems [26, 99].

This lack of simplicity and flexibility has led people to embrace wearable technology. The rise of ubiquitous computing [133] has introduced many new wearable devices in the healthcare domain. Commercially available products, like *Beddit*[1], *Emfit*[2], and *Nokia Sleep*[3] use pressure-sensitive sensor sheets to monitor sleep, while, devices like *Fitbit*[4], *Apple Watch*[5], and *Garmin*[6] are wrist-worn sleep trackers. The sensor sheets use the *piezo-electric* sensors to capture the pressure differences caused by the ballistic forces of the heart. This technique of measuring heart forces is defined as *Ballistocardiograph* and the method of representing it graphically is defined as *Ballistocardiography* (BCG) [43]. Whereas, the wrist-worn devices are based on the principle of Photo-plethysmography (PPG), where the smartwatches measure the changes in the blood flow using low-intensity infrared (IR) light diodes [5, 120]. As per the Statista Consumer Surveys, over 30% of U.S consumers own a fitness device [38]. Various studies have proven that it is possible to monitor the heart rate and measure the sleep quality, alertness, and respiration of the user using such BCG and PPG-based devices [33, 55, 89, 103, 131, 132].

Most of the individuals have not only begun to use sensor-enabled devices to quantify their health but also for tracking chronic medical conditions. Consequently, wearables have begun to be considered as a ”secondary” diagnostic tools [135]. For instance, sleep apnea was reported to be identified using the heart rate, breathing volume and body movements recorded by a sensor mattress working on a BCG technique for recording cardio-respiratory signals [53]. Restlessness during sleep was identified using sleep patterns and a quick feedback mechanism from the device enabled positive habit formation and behavioural changes in users.

Despite the positive changes, there is an attrition in the actual usage of wearables for clinical diagnosis by the users. According to some studies [19, 81], wearable users are concerned about the reliability of the health report generated by these wearables. For instance, in sleep tracking, the lack of reliability stems from the fact that the accuracy of the sleep classification models employed by the current sleep sensors is not promising enough to be used for medical applications. A recent comparative study of various fitness wearables showed significant variations in accuracy among different devices — with error margins of up to 25% [37]. The current state-of-the-art sleep classification systems [53] report an accuracy of 65-70% when it comes to classifying BCG data. This means that there is a failure rate of around 30% which could prove to be quite fatal. It is highly necessary that we address the shortcomings in the accuracies of sleep classification models and attempt to make them be considered for medical diagnoses as well.

This chapter gives an overview of current practice of sleep monitoring in the medical scenario, then describes the current trend in usage of wearables & sensors as an alternative for clinical sleep monitoring, and explains the issues & inadequacies of the current systems in classifying and analysing sleep patterns. Finally, it is followed by the research questions that this thesis aims to address and our contributions towards the work.

1.1. Current sleep monitoring settings

PSG studies are used in sleep clinics to diagnose and treat sleep-realted disorders. Electroencephalogram (EEG), ECG, Electro-Occulogram (EOG) and ElectroMyogram (EMG) electrodes are attached on the subject’s body and the data is viewed live by the sleep expert. Since it is necessary to ensure that the electrodes are attached at the specific locations and have not detached during the sleep, the sleep expert may sometimes monitor the subjects through a live camera feed. The expert then annotates the stages during the duration by following the

[1] https://www.beddit.com
[2] https://www.emfit.com
[3] https://www.withings.com/mx/en/sleep
[4] https://www.fitbit.com/nl/home
[5] https://www.apple.com/lae/watch/
[6] https://sleeptrackers.io/garmin-vivosmart-hr/

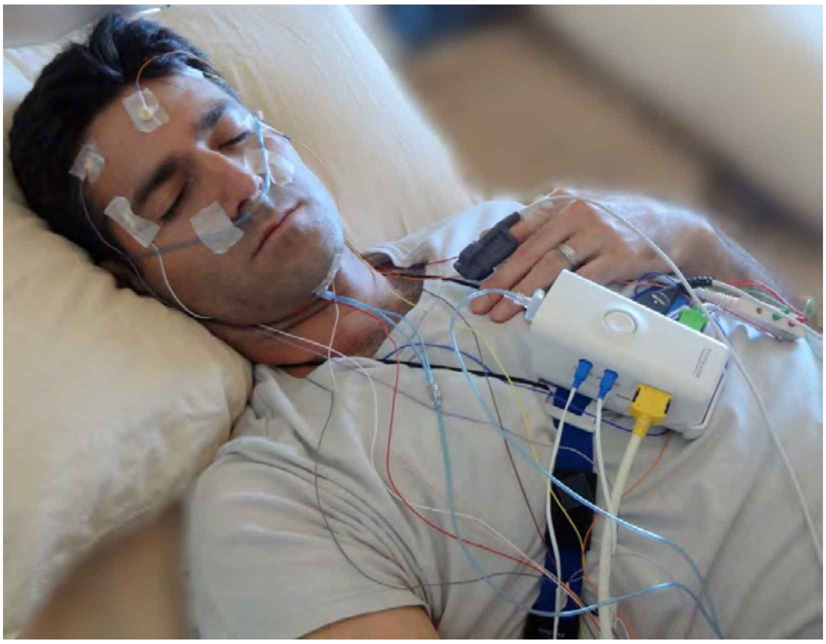


Figure 1.1: A typical sleep study conducted using PSG where the subject has to sleep with over 64 electrodes attached to his body (*Source: WebMD [118]*)

Rechtschaffen & Kales (R&K) guidelines [106] for sleep scoring. R&K scoring rules are based on EEG signals due to which it is critical that the EEG electrodes are perfectly placed on the subject's head and that they do not get detached during the duration of sleep. In addition to that, the subject is advised to try not to move a lot during his sleep to avoid disturbing the electrodes.

Although sleep monitoring using PSG is still considered as the gold standard, this kind of controlled setting create some issues for the user. Inconvenience and privacy concerns have led people to gradually shun away from sleep clinics and instead, believe in the *quantified self* movement wherein the individuals track their sleep from the comfort of their home using trackers.

Sleep tracking using sensors have been made possible due to the advancements in ubiquitous computing. *Ubiquitous computing* or *pervasive computing* is the growing trend of embedding sensors into everyday devices to gather data from the user and use it to minimise user's need to interact with the computer [133]. At present, the sleep tracking devices use BCG [4] and PPG [6] concepts to record heart data unobtrusively. BCG is a method to obtain cardiorespiratory signals by noting the difference in pressure values caused by the ballistic force of our heart during the pumping of the blood. It is widely used in smart mattresses and other similar sensor sheets where the body's contact or proximity with the sensors enables noting of pressure differences in the movements. Trackers based on PPG work on the principle that a heartbeat can be noted by the amount of light reflected or absorbed when blood flows through our wrists. This concept is used in smartwatches like Apple Watch and FitBit. There are wearables like *Dreem*[7], based on EEG sensors for sleep tracking. Although they report a better accuracy than the BCG-based sensors, they are not popular enough since they need individuals to wear the headbands while sleeping.

Due to their small form factor and their non-invasive nature for collecting data, the smart sleep sensors have been viewed as an alternative to PSG studies in the hospitals. Current trackers use various signal processing and machine learning algorithms to classify sleep stages and predict sleep quality. The high sampling rate of the sensors further helps in automating sleep classification due to the generation of large volumes of data.

Despite the advances in collecting, storing and processing of volumes of data, the accuracy reported by these trackers has not been at par with current clinical methods [51]. They

[7] https://dreem.com/en/

Figure 1.2: Various sleep trackers commonly found and used in the market. (*Source: Sleeptrackers.io [1]*)

have been reported to have a high failure rate which makes them less suitable for medical diagnoses [62].

1.2. Issues in sleep classification

The low performance of wearables in classifying sleep can be attributed to the nature of the data generated by the sensors. Data recording using sleep sensors is contaminated by the high amount of noise from movement, powerline disturbances and interferences from other sensors. This makes it challenging to filter the signal. Current signal processing techniques rely on fixed set of parameters to filter the signal. These parameters are usually non-adaptive and do not account for different signals and noises arising from different users. Due to this, extracting HRV features from BCG introduces noise and makes the classifier prone to misclassification.

In addition to that, the current machine learning pipelines for sleep classification rely on their domain knowledge about the correlation between heart activity and sleep quality to engineer features. These kinds of features are called HRV features. However, the problem with this kind of approach is that the HRV features can be accurately identified using the ECG-based heart signals. Although the BCG signal could also provide heart rate information, it differs from the characteristics of ECG. The current features for classifying sleep using heart signal is too *domain specific* and is not easily *transferable* to other similar domains.

The large volumes of data generated by the sensors are difficult to label since they are generated in an uncontrolled setting. Large amount of data is required to train a classifier. Additionally, to obtain labelled data, PSG studies need to be conducted with the help of an expert. However, conducting experiments with PSG is expensive, restrictive and not suitable for long-term monitoring.

Hence, a sleep classification system should be modelled in such a way that it learns to differentiate normal signal from the noisy data and should be able to filter it out. The model should be adaptive and be able to identify the underlying effects of heart activity on sleep and extract features accordingly. Lastly, different training strategies should be used to make the model learn from the limited amount of data.

1.3. Research goal

The lack of reliability of sleep classification from sensor data needs to be addressed. In order to provide the users with other tools for long-term sleep tracking, there is a need to improve the current classifying systems. Through this work, we aim to build a sleep scoring system that can be compared to medical standards by using home-based sleep trackers. Therefore, the underlying research question of this thesis is as follows:

How can a sleep classification system be modelled, using BCG sensor data, in order to *achieve a performance comparable to medical standards?*

The above research question raises further subquestions:

1. Can we model a subject-independent system that is robust to noise?
2. Can we transfer the model, developed using BCG sensor signal, to perform classification on other types of sensors like ECG and PPG?
3. Does the perceived sleep quality of a user correlate with the objective quality metric measured by the model?

1.4. Contributions

To address the research questions posed above, this work proposes a hybrid deep neural net model that performs feature extraction and emulates the time-variant nature of the sleep patterns. A stack of 1D Convolutional neural network is used for self-supervised feature extraction. Sequential stacks of Bidirectional Long-Short Term Memory (bi-LSTM) layers is used in combination with the CNN layers for performing sequential classification. Additionally, in this work, we adopt a 2-phase training strategy wherein we first *pre-train* the model and then *fine-tune* it to achieve better results with limited data. In combination with the subsampling nature of the Convolutional Network (CNN), the 2-phase training strategy further helps in avoiding the vanishing gradient [100] issue faced by the glsbilstm layers. We further test the transfer learning capability of our model by using the pre-trained layers from ConvNet on other types of heart sensors like ECG and PPG and verify if the network can predict sleep stages with comparable performance.

For the sleep classification task, we use a dataset generated from a sleep sensor sheet called Dozee[8]. The sensor sheet works on the principle of BCG. The dataset has been annotated and validated by sleep experts from the National Institute of Mental Health and Sciences (NIMHANS)[9] in India, with the PSG studies forming the reference or the ground truth for the BCG data.

Based on the performance analysis of our *DeepSleep* model on the BCG data, we show that our model achieves an overall precision & recall score of around 74% when performing a 4-class classification, and a score of 82% with 2-class classification task (REM, NREM). Our model's performance has been shown to perform slightly better than the other studies that have used BCG data. Additionally, a user study based on the SATED framework, conducted using 16 users of the Dozee device, showed that the objective sleep quality measurement correlates with the subjective score reported by the users ($r = 0.43$). This shows that even though our model does not achieve performance comparable to the gold standards, it is still able to identify important sleep parameters that determine the overall sleep quality. This shows that our model can be used for general sleep tracking and monitoring, if not diagnosing sleep disorders. To the best of our knowledge, we believe that the *DeepSleep* model is the first model to utilise the power of deep neural nets to perform sleep classification on the BCG data.

1.5. Thesis Outline

The structure of this thesis is as follows. Chapter 2 gives an overview of the physiology of sleep and elucidates previous works on sleep classification systems. In Chapter 3, we describe the datasets and the architecture employed in this work. The model's design choices, training methodology and the details of the implementations are further discussed in detail in this chapter. We present our results later in Chapter 4. Finally, we discuss the learnings from this thesis and provide an outlook for future research opportunities in Chapter 5.

[8] https://dozee.io
[9] https://www.nimhans.co.in

2

Related Work

Sleep is primarily governed by brain activity. Over time, many biologists have tried to understand the internal state of our brain during sleep. Initial works in sleep classification systems have relied on this domain knowledge to understand features that described the sleep stages. Hence, before proceeding to build a classification model, it is important to understand the physiology of sleep and how the brain and the heart activities are affected by sleep. In this chapter, Section 2.1 briefly explains the biological aspect of sleep and describes the various terminologies used in this domain. The effect of sleep on heart activity, heart-based features used for identifying stages, the concept of ballistocardiography (BCG), and the sleep scoring rules for PSG studies in clinics are further explained in this section. Section 2.2 describes the deep learning algorithms that are used in this work. It is followed by Section 2.3 which discusses prior works in sleep classification. Lastly, Section 2.4 describes the various methods of quantifying subjective and objective sleep quality.

2.1. Physiology of Sleep

According to Tabuchi and Wu [123], sleep is a period during which the brain is engaged in number of activities which are closely linked to the quality of life. Human sleep is characterised by two basic stages, *REM & non-REM (NREM)*. Each of these stages is associated with the specific brain-wave patterns and neuronal activity. Two internal biological mechanisms define our sleep patterns. These mechanisms are *Circadian rhythms* and *Sleep-wake homeostasis*. *Circadian rhythms* control various functions ranging from fluctuations in body temperature, metabolism, and the release of hormones. They control the timing of our sleep and wake by causing us to be sleepy and awake at particular times without the need of an alarm. This is what we call as body's biological clock. Whereas, *sleep-wake homeostasis* tracks the need for sleep and keeps a check on the intensity of sleep. It reminds our body to sleep after a certain time. It is usually influenced by medical conditions, medications, environment, diet and mental stress [95].

Sleep is characterised by 2 basic stages, REM & non-REM (NREM). Each of these stages is associated with specific brain-waves and neuronal activity. NREM is further divided into 3 stages [95]. The stages are describes as follows:

1. **NREM 1**: This NREM stage indicates the transition from wakefulness to sleep. During this stage, the heartbeat, breathing and eye movements gradually decrease and the muscles get relaxed with occasional twitches. The brain waves begin to go slow when compared to daytime wakefulness patterns.

2. **NREM 2**: This NREM stage is another light sleep period which indicates the transition of our sleep into deeper sleep. The heartbeat, breathing and eye movements further slows down and the body temperature begins to drop. Most of the human sleep patterns spend more time in this stage than in any other stages.

3. **NREM 3**: This stage indicates the period of deep sleep. This stage is important for an individual to feel refreshed in the morning as the body restores, relaxes and repairs muscles and releases hormones like growth hormones. The heartbeat and breathing go to their lowest range during this period. This stage occurs for longer time during the first half of the night.

4. **REM**: *Rapid Eye Movement* or REM stage occurs about 90 minutes after the onset of sleep. Our eyes move rapidly from side to side behind closed eyelids and the brain waves fluctuate with the frequency as seen in wakefulness. Our breathing becomes faster and irregular, the heart rate and blood pressure increase to near waking levels, and the arm and leg muscles become temporarily paralyzed. The muscles are paralyzed since most of our dreaming occurs during REM sleep and this prevents acting out during our dreams. REM sleep increases in duration over the night and it is important to consolidate and learn memories assimilated by the brain over the course of the day.

Usually, for sleep studies, sleep experts combine *NREM 1* and *NREM 2* into a single stage called *Light* sleep, since, the brain wave activity is nearly indistinguishable between these two stages. This transition between the *REM* & *NREM* occurs in 4-5 cycles over a night of sleep. A *sleep cycle* is defined as the complete transition of sleep from wakefulness to deep and then, to REM; light stage being sprinkled in between these transitions. Since REM and NREM occur in tandems after every 60-90 minutes, it is often advised to sleep around 7 hours to get a minimum of 4 cycles of REM and NREM. However, this number varies with age, gender, genes, environment, and medications.

The brain activities are analyzed to identify different sleep stages. These activities are record using an EEG test. In sleep clinics, a polysomnography (PSG) machine is used to record physiological signals and to analyze sleep patterns.

2.1.1. Polysomnography-based Sleep Scoring

Polysomnogram (PSG) is a study employed in sleep labs or hospitals to measure and record the physiological activities during sleep. A sleep expert then reads the brain wave activity along with the activities of eye movements and body movements to determine the sleep stages. The rules for scoring sleep using an EEG and EOG has been developed by *Rechtschaffen and Kales* (R&K, 1968) [106]. The sleep scoring guidelines have been recently modified by American Academy of Sleep Medicine (AASM).

In 1968, a group of researchers under the chairmanship of Allan Rechtschaffen and Anthony Kales convened to develop guidelines for staging and scoring sleep in normal human beings [106]. These guidelines were published in a manual subsequently and have been the reference for sleep staging since then. Apart from setting rules for sleep staging, R&K had set criteria for recording sleep data using PSG. The manual specifies that a minimum of one channel of central EEG, chin EMG, and two channels of EOG (electrodes to be placed above and below the eye, laterally). They also recommended an epoch-to-epoch approach for scoring sleep. The recommended epoch length was set as either 20 or 30 seconds.

R&K manual categorised sleep stages primarily into 6 stages: *Wake*, *Stage 1*, *Stage 2*, *Stage 3*, *Stage 4* and *REM*. The scoring criteria for these stages is as follows:

1. **Wake**: More than 50% of the epoch should consist of alpha wave (8-13 Hz) activity or low voltage, mixed (2-7 Hz) frequency activity. Alpha activity is mostly seen during quiet alertness with eyes closed [23]. There is a decrease in amplitude when the eyes open.

2. **Stage 1**: 50% of the epoch should consist of relatively low voltage mixed activity (2-7 Hz), and less than 50% of the epoch should contain alpha activity. Slow rolling eye movements can also act as scoring indication.

3. **Stage 2**: Sleep spindles and/or K-complexes should appear for more than 0.5 seconds. Also, the EEG activity may begin to consist of slow activity (<2 Hz).

4. **Stage 3**: 20-50% of the epoch begin to contain slow activity indicated by low frequency (<2 Hz).

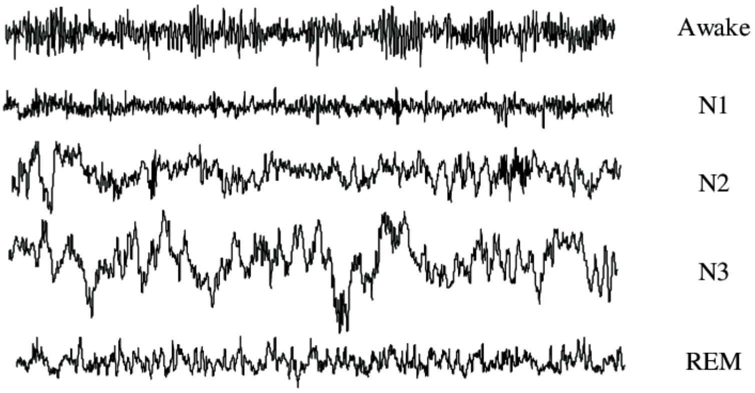


Figure 2.1: EEG wave patterns during different stages of sleep (*Source: National Institutes of Health [95]*)

5. **Stage 4**: More than 50% of the epoch consists of delta activity (amplitudes of 75μV).

6. **REM**: The EEG voltage should be relatively low with mixed activity (2-7 Hz). This should be coupled with rapid-eye movements indicated by EOG and reduced or absent chin EMG activity. The low mixed activity resembles the activity during wakefulness, however, the muscles are paralyzed which is indicated by EMG signal.

Fig. 2.1 shows the patterns formed by the EEG signals during the different sleep stages. Ideally in sleep hospitals, the experts score the sleep stages by visually checking for the EEG wave patterns as defined in each of the stages.

Although these guidelines have been followed for decades by sleep scientists, limitations of R&K scoring system [118]. Firstly, the dependence on an epoch-based system often needs the scorer to rely on the preceding and following epochs before scoring the current one. This considers sleep staging to be discrete rather being continuous. Hence, the scoring of abnormal sleep or highly fragmented sleep becomes quite challenging since it is difficult to keep track of previous epochs when there is a lot of wakefulness or movement activity. R&K system fails when the subject has breathing disorder or inconsistent sleep patterns. Hence, to tackle the limitations posed by the R&K system, AASM in 2007 modified and set new guidelines for sleep scoring. To make the scoring easier and more accurate for sleep experts, AASM decided to reduce the number of stages. The team set up by AASM to review the guidelines reviewed a study of 8 subjects and changed the rules for placing the electrodes for EEG and EOG. The team came to a consensus on the terminology of sleep stages and decided upon the following:

1. Stage W (Wakefulness or Wake)
2. Stage N1 (NREM 1 or *light* sleep)
3. Stage N2 (NREM 2 sleep)
4. Stage N3 (NREM 3 sleep)
5. Stage R (REM sleep)

2.1.2. Identifying Sleep Stages using Heart signals

Section 2.1 and 2.1.1 explained how sleep is associated with various physiological activities. Heart rate gradually slows down as the sleep transits into Deep sleep and it increases, close to its wakeful state, during the REM period. These changes in heart activity suggests that heart rate can be used to correlate with sleep stages. However, the heart activity is subtle and is not very distinguishable unlike the brain wave activity. It is almost impossible for a human scorer to visually score the sleep stages based on just the heart activity recorded by ECG electrode. The heart signal itself is not very useful to visually score the stages. However, the subtle changes in the heart activity can be captured by certain set of features called *Heart rate variability* features.

Heart rate variability (HRV) is a physiological phenomenon of variation in the time interval between heartbeats. It is basically a measure of the variation in the beat-to-beat interval [113]. HRV helps in studying the relation between the autonomic nervous system (ANS) and cardiovascular system. Hence, HRV does not only find its value in sleep stage identification but also helps in identifying heart-related disorders like arrhythmia (irregular beats), cardiac arrests and myocardial infarction [17, 113].

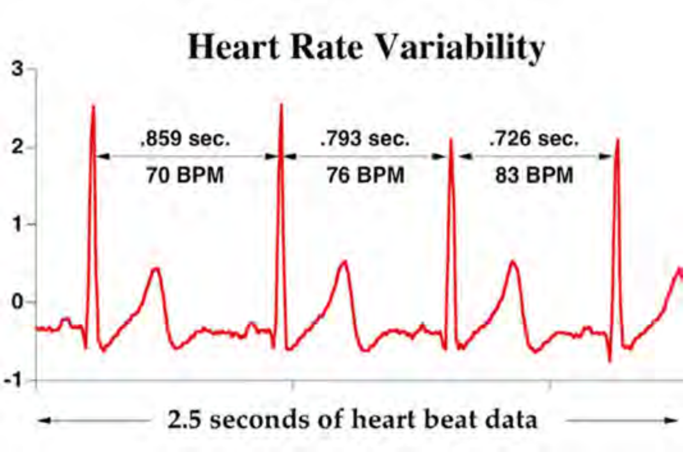


Figure 2.2: Depiction of heart-rate variability phenomenon on an ECG waveform.

An ECG signal is mainly characterised by the shape of the wave form when a heart beat occurs. The shape of the wave is defined by the *PQRS* waveform which contains most of the information of the beat. The P-wave characterises the beginning of the beat, followed by a trough (Q-wave) formed by the contraction of the heart muscle and then, the peak is formed which indicates the beat (pumping of blood from heart chambers). This peak is called as the *R-wave* or *R-peak*. Usually, the waveform of interest for heart-related tasks is defined in the QRS wave. And, HRV is the study of the variation between the adjacent R-R peaks, or simply called as R-R intervals. Hence, HRV features are usually extracted from the R-R intervals in the ECG signal.

Time Domain

The basic and simplest features that can be extracted from the R-R intervals are the time-domain features. These features are mean RR interval, mean heart rate, and mode of RR values. Table 2.1 shows the time-domain and statistical features used in the HRV analysis.

Statistical Methods

When analysing long duration of ECG recording and analysis, statistical methods prove to important to study the heart's activity. One of the feature extracted using this method includes the standard deviation of the RR intervals (*SDNN*). Since, variance is mathematically equal to total power of a spectrum, SDNN helps in studying the short-term variations and removes the lowest frequency components. However, SDNN fails when the recordings are of longer cycle lengths.

Other features extracted using statistical methods include standard deviation of the average of RR interval (*SDANN*), root mean square of the differences of successive RR intervals (*RMSSD*), the number of RR-interval differences where the differences in the interval length is ≥50ms (*NN50*), and the ratio of the NN50 and the total number of RR intervals (*pNN50*). All of these features measure the short-term variations of the high-frequency estimation in heart rate.

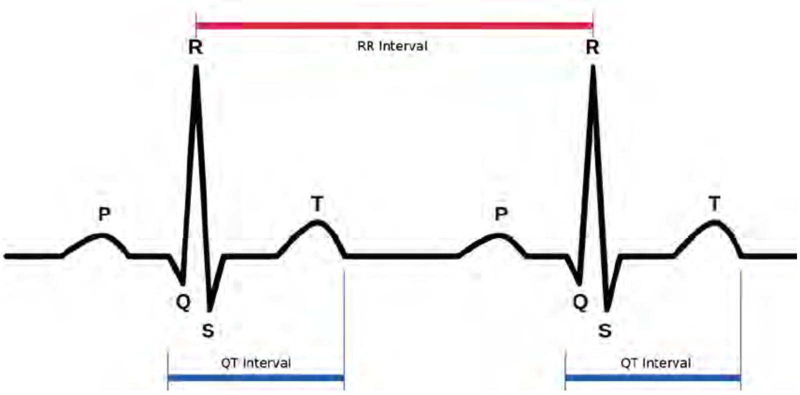


Figure 2.3: Typical waveform characteristics of an ECG heart signal (*Source: National Institutes of Health [95]*)

Geometric Methods

The RR intervals can also be converted into a geometric pattern and the variability on the basis of the geometric or the graphical properties of the resulting pattern can be studied. Conversion to geometric patterns include sample density distribution of RR interval durations, sample density distributions of differences between adjacent RR intervals, and Lorenz plot of RR intervals. These features have been proven to be influenced more by lower frequencies [113].

Frequency Domain

Spectral analysis of the RR intervals depends on knowing the *power spectral density* (PSD) of the signal. PSD can be calculated either using Fast Fourier transform (FFT, non-parametric method) or by using wavelet transforms (parametric method).

In terms of HRV, three main spectral components are identified in heart signal: VLF (very low frequency), LF (low frequency), HF (high frequency). LF and HF features, in their normalised form, represent the two branches of the autonomic nervous systems. Hence, the relative power distribution and the LF/HF ratio helps in determining the state of sleep; wakefulness or deep sleep. Other activities like sleeping posture and breathing rate can be determined using these features.

The HRV features help in parametrizing sleep in terms of the heart activity. However, the mode of measurement of these features introduces some flaws in the system. Firstly, the extraction of HRV features rely on the RR intervals. If the identification of R-peaks is not accurate enough or if one of the R-peak is mis-identified or not identified at all, then the extracted features fail to capture the variations of the heart. Secondly, spectral analysis on a longer signal length (24 hr) does not consider the ”stationarity” effect. When the heart beat modulations keep fluctuating, the PSD features fail to encapsulate these non-stationary frequency components. Hence, the interpretation of these features for sleep stage identification becomes poorly defined.

Having said that, heart signals can be used for predicting sleep stages since there are notable changes in the heart activity during sleep. Moreover, heart signals are easier to record unobtrusively with the help of wearable devices. Hence, this provides an opportunity to base the sleep classification task using the heart signals.

2.1.3. *Ballistocardiography* - Unobtrusive recording of Heart activity

It has been known for long that our body moves with every heartbeat. These subtle movements are attributed to the body's recoil to cardiac expulsion of blood into the arteries. *Ballistocardiography* (BCG) is an unobtrusive method based on the measurement of these ballistic forces. Formally, ballistocardiography is a non-invasive method based on the measurement of the body motion generated by the ejection of blood at each cardiac cycle. It relies on the detection of the cardiac and cardiovascular-related mechanical motions [68].

Name	Description	Formula
mRR	mean of RR intervals	$\frac{1}{N}\sum_{i=1}^{N} RR_i$
HR	mean Heart rate	$\frac{60000}{mRR}$
SDNN	Standard deviation of normal R-R intervals	$\sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(RR_i - mRR)^2}$
CV_RR	Coefficient of variation	$\frac{SDNN}{mRR}$
RMSSD	Square root of the mean squared differences of successive RR intervals	$\sqrt{\frac{1}{N}\sum_{i=1}^{N-1}(RR_{i+1} - RR_i)^2}$
pNN50	Percent of normal R-R intervals that are greater than 50ms	$\frac{num(RR \geq 50)}{num(RR)}$
RR_mod	Mode of RR (most repeated value in RR)	$mode(RR)$
RRdif_mod	Mode of RR's first order of difference	$mode(\partial RR)$

Table 2.1: List of Time-domain and Statistical HRV features.

The BCG waveform is difficult to understand as it constantly varies from person-to-person and sensor-to-sensor. Most of its understanding are based mainly upon empirical correlations with other measurements like electrocardiogram [112]. However, recent works like that of Kim et al. [68] have successfully managed to explain the important BCG waveforms using mathematical models. This has led to better understanding of the BCG waveforms, making their associations with ECG-related waveforms and cardiovascular disorders, more meaningful.

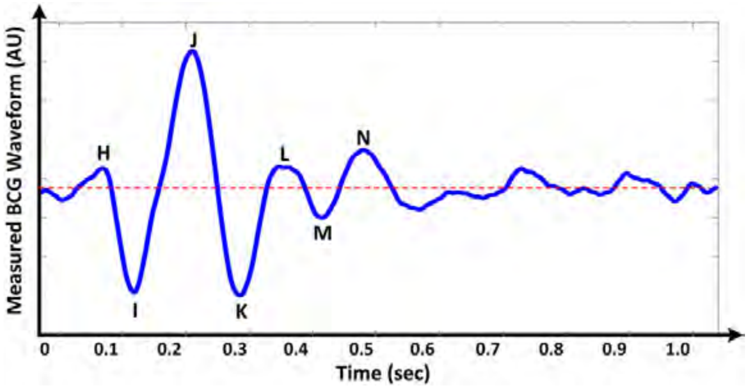


Figure 2.4: General signal characteristics of a BCG waveform

Just like how the *PQRS* waveform generated by an ECG recording represents the structure of a heartbeat, BCG generates *IJK* waveform for the same. Fig. 2.4 shows the structure of the BCG waveform. The J-peaks in BCG correspond to the R-peaks in the ECG. This understanding helps us in extracting HRV features mentioned in Section 2.1.2 by treating it as a domain-transferable task.

As the BCG method is used to record the ballistic movement forces, devices or wearables using this concept employ *piezoelectric* sensors to record the BCG data. Piezoelectric sensors use piezoelectric effect, to record changes in pressure, acceleration, temperature, or force by converting these changes into an electric charge [116]. For pressure sensors (BCG-like use cases), a thin membrane and a massive base is used so as to transfer the force to the sensor elements. Due to their nature of design for detecting pressure forces, these sensors are sensitive to vibrational values, thus introducing varied amounts of noise in the BCG signal acquisition. This has been one of the main limitation of the BCG for medical diagnosis since the signal itself is contaminated with movement and breathing artefacts which are difficult to filter due to its non-stationarity nature. It makes detection of J-peaks a challenging task as the peaks are not highly distinguishable unlike the peaks generated in ECG. And, since the HRV features are extracted from the R-R or J-J intervals, missed or wrong peaks would in-turn give rise to error in HRV feature extraction. Hence, the usage of BCG-based sleep-monitoring system have not been reliable enough for medical usage.

Having said that, there have been some notable studies [22, 88, 93] that have attempted at improving the heartbeat detection in BCG waveform by adopting different adaptive signal processing techniques. The review on the advances of BCG systems by Inan et al. [59] gives a positive statement that the current advances of computational tools in signal processing algorithms has led to reduced BCG measurement noise, thus improving the physiological studies performed on human subjects. Giovangrandi et al. [43] reiterate the statement that although the current BCG systems still face issues like lack of standardization and proliferation of measurement methods, BCG can still be very well used in places not reachable previously — people's home. They believe that better processing methods can enable the BCG to be used as a diagnostic tool.

Most contributions towards the automation of sleep classification comes alongwith the advancement in the field of machine learning. Earlier works on sleep classification involved studying EEG signals, extracting features pertaining to EEG and using classical machine learning models for classification. With time, the focus shifted towards ECG, respiratory and movement signals as they are easier to obtain. These signals can be collected non-invasively or with much lesser electrodes on a subject's body. The ability to correlate *heart-rate variability* (HRV) with human sleep led to further developments of sleep scoring using *Ballistocardiogram* (BCG) signals. BCG signals allowed researchers to study sleep patterns using non-invasive data acquisition techniques.

2.2. Deep learning

There are many definitions for *deep learning* but all of them lead to this common understanding of the field: Deep learning is a subclass of machine learning which is concerned with algorithms that were inspired by the structure and function of the brain called artificial neural networks. Various algorithms and architectures trying to depict the working of a human brain have been studied but only few of them obtained success in recent times. Two such algorithms are *convolutional neural networks* and *recurrent neural networks*. This work uses these algorithms as well to build the classification model and hence, attempts at giving a brief explanation about them.

2.2.1. Convolutional neural networks

A convolutional neural network (CNN) is a class of feed-forward neural networks that has been widely used for analyzing images [48, 75]. Given a set of input and weights, a traditional feed-forward network computes outputs using an activation function. The working of a perceptron, a type of feed-forward neural network, is given as follows:

$$y = f(W_{ij}x_i + W_{ij+1}x_{i+1} + \dots) \tag{2.1}$$

where, y denotes the output, W_{ij} being the weight parameter, x_i denotes the inputs and f is the activation function.

In a matrix form, the above equation is represented as:

$$y = f(Wx + b) \tag{2.2}$$

where, b is the bias term. When large data formats like images are fed into a perceptron, the image is flattened before giving it as an input. A simple $20*20$ image when flattend introduces 400 inputs, thus introducing 400 bias and weight parameters that need to be optimized. Hence, a convolutional network was developed to deal with such kind of problems.

A CNN layer processes localized patches of an image. The size of the patch is determined by a set of parameters that define the CNN layer. The parameters are described as follows:

1. **Filter size**
   The filter size (or kernel size) (F), determines the size of the patch that needs to be convolved in one forward pass. For an image, the filter size is a 3D tensor, (L, W, C), denoting the length, width and the number of channels of an image's patch. Hence, for an image of size $32 * 32$, if a filter size of $5 * 5 * 3$ is chosen, a $5 * 5$ filter is randomly intialized with weights and then performs the convolution operation (element-wise matrix multiplication) over the whole image.

   $$y = W \otimes x \tag{2.3}$$

   Thus a feature map, y, of size $28 * 28$ is produced after the complete convolutional process.

2. **Stride**
   The stride parameter (S) determines the amount of translation that a filter has to make after every convolutional operation. The stride parameter helps in subsampling the image and also, to control the number of parameters to be examined.

3. **Pooling**
   Pooling layers are used after obtaining a feature map to reduce the size of the feature map by removing duplicate or commonly occurring convolved values. Pooling layers are used to prevent the CNN layer from overfitting by removing redundant values from the feature map. There are two variants of pooling layers that are commonly used:

   (a) Max-pooling: The feature values are pooled using the *max* function. Only the maximum values in the feature map are retained.

   $$maxpool = max(f_i) \tag{2.4}$$

   where, f_i denotes the feature value for i^{th} pixel.

   (b) Average pooling: The feature values are pooled by computing their mean values.

   $$avgpool = avg(f_i) \tag{2.5}$$

Hence, for an image of size $L * W * C$, new convolved feature maps of size $L_1 * W_1 * C_1$ is produced. These are given by:

$$L_1 = \frac{L - F + 2P}{S} + 1$$
$$W_1 = \frac{W - F + 2P}{S} + 1$$
$$C_1 = k$$

where, P = zero padding and k = number of features.

An activation function basically converts the input into a specific range of output. The range of the output depends on the type of the activation function. The selection of the activation function depends on the nature of the problem (classification or regression) and the intended nature of the output. *ReLu* function is one such activation function which has been shown to perform well during the intermediary training process. *ReLu* introduces *non-linearity* into the domain space and has been to proven to be a better activation function when dealing with the weights of the neural networks. A $ReLu$ is based on the following function [92]:

$$R(x) = max(0, x) \tag{2.6}$$

i.e

$$R(x) = (0, x < 0; x, x \geq 0)$$

Eq. 2.2.1 shows that $ReLu$ is a simple and efficient function. Due to it's non-linear property, a $ReLu$ unit helps in producing smoothened gradients in the back-propagation & gradient descent models. However, a $ReLu$ function can only be used as an activation function for the hidden layers of the networks where gradient calculation is required. The weights are learned and readjusted by following the back-propagation algorithm [82] and stochastic gradient descent [13].

2.2.2. Recurrent neural networks

Recurrent neural networks (RNN) were designed to process sequential information, especially, the problems that require persistent noting of earlier occurrences. RNNs help in reasoning the next sequence based on the knowledge about the previous state. RNNs are designed in a looped-chained format where the same type of information is passed over and again till it reaches a optimum solution. An RNN consists of:

1. x_t, input at timestep t
2. s_t, state of the hidden layer at timestep t
3. o_t, output at step t

These three parameters are related as follows:

$$s_t = f(Ux_t + Ws_{t-1}) \tag{2.7}$$

$$\begin{aligned} \text{where}, f &= \text{activation function} \\ U, W &= \text{layer weights (parameters)} \end{aligned}$$

These layer parameters are shared across every RNN cell which reflects the fact that the same task is performed across each cell of an RNN. As can be seen in Eq. 2.7, the current state of the RNN cell depends on the current input and the previous state. This dependancy allows for performing tasks like language modelling and machine translation. However, as the length of dependancy increases, RNNs find it difficult to connect the current state to a very old state. This problem is defined as the vanishing gradient problem.

To prevent the vanishing gradient issue and to enable long-term dependancies, Long-Short term Memory networks (LSTM) were introduced. LSTM network is a variant of RNN network with the ability of storing state information in its so-called *memory* unit. The core idea behind an LSTM network is the working of its internal gates. The gates choose to either store or forget state information based on their internal activation function. Hence, the internal states of an LSTM cell are given as:

$$\begin{aligned} i_t &= \sigma\left(x_t U^i + h_{t-1} W^i\right) \\ f_t &= \sigma\left(x_t U^f + h_{t-1} W^f\right) \\ o_t &= \sigma\left(x_t U^o + h_{t-1} W^o\right) \\ \tilde{C}_t &= \tanh\left(x_t U^g + h_{t-1} W^g\right) \\ C_t &= \sigma\left(f_t * C_{t-1} + i_t * \tilde{C}_t\right) \\ h_t &= \tanh(C_t) * o_t \end{aligned}$$

where, i, f, o are the input, forget and the output gates.

There are some variants of LSTM network like Bi-directional LSTMs (bi-LSTM) and Gated recurrent networks which were designed to perform sequential tasks with lesser parameters. bi-LSTM are primarily used in applications where the future state information is also needed along with the past information.

2.3. Sleep stage classification

The cumbersome nature of the PSG studies do not allow long-term sleep monitoring and tracking. This led people to use sleep wearable devices to track their sleep. The form-factor of the sensors and the easy collection of sensor data provided large volumes of data. This had increased the interest in automatic sleep tracking and sleep stage prediction.

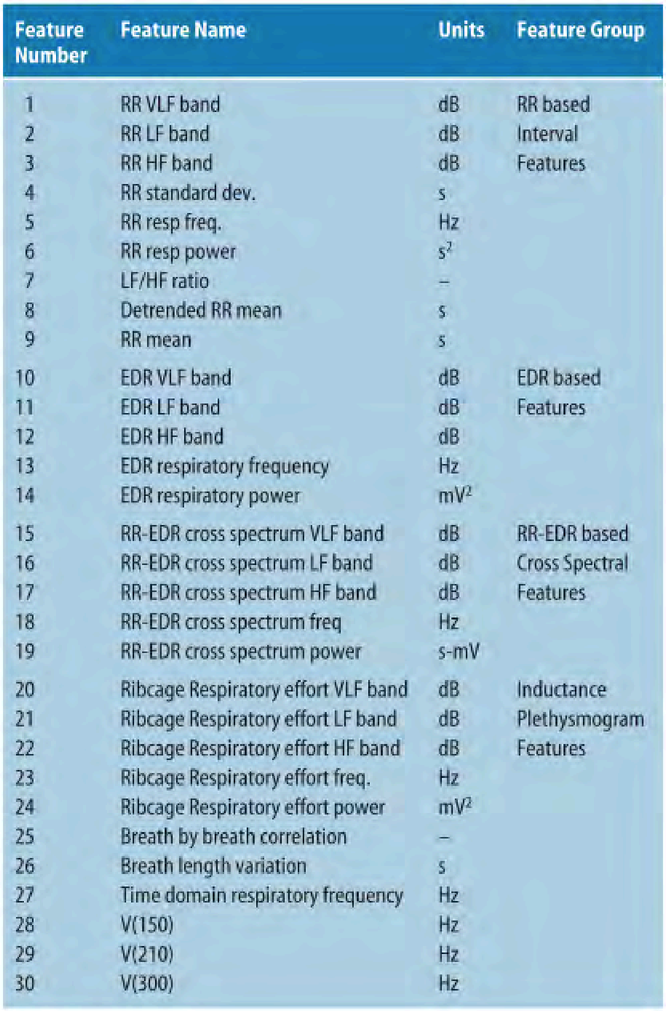
Earliest studies in the domain of sleep stage prediction used signal processing techniques to extract, filter and process the signals such as EEG, ECG, EOG, and EMG to study the science behind sleep. With the increased availability of sensor data, machine learning-based methods for sleep stage prediction became more popular. HRV features were extracted using the domain knowledge and machine learning algorithms were used to perform classification. Most recent works in this domain are based on deep learning algorithms wherein least domain knowledge and preprocessing is required to perform the same task. In this section, various prior works employing classical machine learning techniques using ECG and unobtrusive wearable signals (Section 2.3.1, 2.3.2) and deep learning-based models (Section 2.3.3) are discussed.

2.3.1. Supervised classification using ECG

Earlier studies on sleep classification involved rigorous feature engineering and the usage of classical machine learning paradigms to obtain good results. Redmond et al.'s [107] work on sleep staging using cardiorespiratory signals is one of the notable study in this field. Their work focussed on determining if sleep can be classified between *Wake*, *REM* and *Light* by just using the cardio and respiratory signal. They obtained the signal from the subjects suffering from *Obstructive Sleep Apnea.* Redmond et al. conducted the experiment on 31 subjects and obtained their complete polysomnography data. The sleep stages were scored by a single sleep expert using all the PSG signals, however, the authors considered only the ribcage respiratory effort signal for their classification algorithm. The respiratory signal was then preproccesed by identifying the R-peaks in the ECG component of the signal, centering and removing the variance in the data sample and extracting the respiratory component from the cardiorespiratory signal. The complete signal was divided into a non-overlapping epochs of 30 seconds and features were extracted for each of these epochs. Over 30 distinct features were extracted for the purpose of sleep staging. The features represented the frequency-domain of ECG (*VLF, LF, HF*), time-domain (*RR-interval, peak correlation, maximum peak etc.* and spectral power domain (*power density, respiratory frequency etc.*). The authors ensured that the features captured the individual signal characteristics of the ECG and the respiratory wave components in addition to their correlation features. Redmond et al. notably pointed out that sleep staging is not a random phenomenon but it rather depends over the duration of sleep. Hence, instead of assigning equal probabilities to all the classes, time-based class weights were incorporated.

The authors reported an accuracy of 69% upon training the data for 3-class classification (W, REM, Sleep) using a *quadratic discriminant classifier.* The accuracy improved to 82.5% when the same classifier model was applied on 2-class classification (W, Sleep). Although the study showed that sleep staging can be performed using cardiorespiratory signals, the accuracy scores suggest that the features extracted do not fully encapsulate the problem's dimensionality space. Time-based class weights certainly increased the performance, however, the feature points are still considered as discrete points rather than as a continuous time-series signal.

Another study conducted by Fonseca et al. [39] focuses on classifying sleep stages into 4 classes - *Wake, REM, Deep, Light* by using ECG and Respiratory Inductive Plethysmography (RIP) signals for cardio and respiratory data respectively. This work employs some

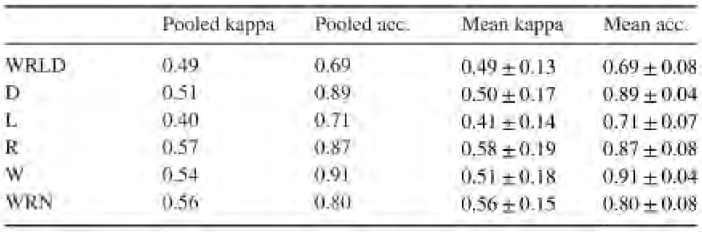


Feature Number	Feature Name	Units	Feature Group
1	RR VLF band	dB	RR based Interval Features
2	RR LF band	dB	
3	RR HF band	dB	
4	RR standard dev.	s	
5	RR resp freq.	Hz	
6	RR resp power	s^2	
7	LF/HF ratio	–	
8	Detrended RR mean	s	
9	RR mean	s	
10	EDR VLF band	dB	EDR based Features
11	EDR LF band	dB	
12	EDR HF band	dB	
13	EDR respiratory frequency	Hz	
14	EDR respiratory power	mV^2	
15	RR-EDR cross spectrum VLF band	dB	RR-EDR based Cross Spectral Features
16	RR-EDR cross spectrum LF band	dB	
17	RR-EDR cross spectrum HF band	dB	
18	RR-EDR cross spectrum freq	Hz	
19	RR-EDR cross spectrum power	s-mV	
20	Ribcage Respiratory effort VLF band	dB	Inductance Plethysmogram Features
21	Ribcage Respiratory effort LF band	dB	
22	Ribcage Respiratory effort HF band	dB	
23	Ribcage Respiratory effort freq.	Hz	
24	Ribcage Respiratory effort power	mV^2	
25	Breath by breath correlation	–	
26	Breath length variation	s	
27	Time domain respiratory frequency	Hz	
28	V(150)	Hz	
29	V(210)	Hz	
30	V(300)	Hz	

Figure 2.5: HRV features used by Redmond et al. in their work [107]

novel feature extraction, feature selection and post-processing techniques to obtain better results than that of Redmond et al. [107]. Fonseca et al. conducted the study using data from 48 healthy subjects. Instead of using a combined cardiorespiratory signal for feature extraction, the authors extracted the respiratory signal using a 10th order Butterworth filter [12]. They extracted over 142 features comprising of cardiac, respiratory, and power spectral features. In order to reduce the bias introduced by the inter-subject variability, each feature was Z-score normalized by subtracting their mean and dividing by their standard deviation. One important contribution the authors make is treating the sleep staging problem as being dependent on the sleep cycles which in turn depends on the duration of sleep. Thus the authors treated sleep classification as being time-variant but occurring in a batch cycles of around 90-110 minutes.

To address this issue, a *cubic spline* fitting method was employed. Cubic spline further smoothens the signals and makes it usable even if there was a presence of motion artefacts. A multi-class *Bayesian* linear discriminant with time-varying class weights was used as the classifier. Since the total number of features being used is too high and can invariably overfit the model, the authors considered *Sequential Forward selection* [61] as their feature selection method. The least number of features, *S*, that can maximize the Cohen's kappa criterion was set at 80 after multiple fitting rounds on the training data. The study reported an accuracy of 69% for classifying *W, R, D, L* and 80% for the classification of *W, R, D.* Although their performance fares among the other state-of-art results in 4-class classification, their classifier fails to predict *Wake* stage well and has a high misclassification error when it comes to



	Pooled kappa	Pooled acc.	Mean kappa	Mean acc.
WRLD	0.49	0.69	0.49 ± 0.13	0.69 ± 0.08
D	0.51	0.89	0.50 ± 0.17	0.89 ± 0.04
L	0.40	0.71	0.41 ± 0.14	0.71 ± 0.07
R	0.57	0.87	0.58 ± 0.19	0.87 ± 0.08
W	0.54	0.91	0.51 ± 0.18	0.91 ± 0.04
WRN	0.56	0.80	0.56 ± 0.15	0.80 ± 0.08

Figure 2.6: Sleep classification performance report as given in Fonseca et al.'s work [39]

differentiating *REM* and *Deep*. The cubic spline fitting helps in improving the classification between *REM* and *Deep* as it smoothens the motion signal. It penalizes abrupt and short changes, thus making identifying *Wake* stage difficult. Even after reducing the number of features to 80 from 142, there are cases of overfitting in the performance since *Light* stage was the most predicted class by the classifier.

Another study on sleep classification by Xiao et al. [134] focuses on using HRV features and random forest classifier to classify sleep. The research goal of the study was to improve the performance of sleep classification using subject-independent data. A total of 41 HRV features were extracted and random forest was used to note the importance of each feature. The study performed their experiment on a PSG data comprising of 41 subjects. Only the RR sequences of the PSG data were used since the HRV feature extraction only requires the RR intervals instead of the entire ECG data. In comparison to the study by Redmond et al. and Fonseca et al., this work introduces a new feature called *Zero Crossing analysis*. This feature introduced non-linearity in the feature space. Other HRV features like time domain, frequency-domain and non-linear features (detrended fluctuation analysis (DFA), autocorrelation coefficients and Mutual Information (MI)) were extracted for the experiment. A novel contribution from this study was the way the authors dealt with signal normalisation and feature selection. The non-linear features were normalised by first decomposing the signal into 6 components using discrete wavelet transform (DWT) and db6 mother wavelet [67]. The sum of the standard deviation of the component was later used to reconstruct the RR sequence.

The study used two ways of evaluating the classifier - accuracy scoring and Cohen's kappa statistic (κ). The random forest classifier was trained on two sets of the data: 1) Subject-specific classifier and 2) Subject independent classifier. The study reported an accuracy of 88.67% and $\kappa = 0.7393$ for subject-specific classification and an accuracy of 72.58% and $\kappa = 0.46$ for subject independent classifier. The accuracy of the subject independent classifier was reported to be improved by 18% when fractile-based normalisation was used. The authors placed the features into confidence intervals of 5% and 95% and normalised only those features that were between the two confidence intervals. Although the accuracies reported by Xiao et al. seem encouraging enough, it is clear from the kappa statistic that the agreement of predicted classes between the subject-independent classifier and PSG classes is moderate to low. Moreover, the authors simply discard the NREM data samples from the training data so as to address the class imbalance problem. This leads to loss of information for the classifier. Lastly, the sleep classification can have possibly imporoved if the authors considered weighting the classes according to time rather than considering them as equally probable of occurring at any time of the sleep duration.

A more recent work in 2017 by Fonseca et al. [40] considers the limitations of their previous work [39] and the other above works. The authors agree that sleep is structured process and that it cannot be treated as independent of time. Hence, this work is based on a generalisation of Hidden Markov Model (HMM) called *Conditional Random Fields* (CRF). CRF models do not assume that the features are independent and the states are discrete. They compute the probability of a possible output y given an observation x. This way the CRF model calculates the correlation between the features and also takes the previous state into account when predicting the current one. Unlike HMM, CRF would need not know the distribution of the observations from the data, since computing joint probabilities from the dependent vari-

Study	Sensor type	#Features	Classifier	#Classes	Accuracy
Redmond et al. [107], 2007	ECG, RIP	30	LDA	W, Sleep W, REM, Sleep	82.5% 69%
Fonseca et al. [39], 2015	ECG, RIP	142	LDA	W, REM, Deep W, REM, Deep, Light	80% 69%
Xiao et al. [134], 2013	ECG	41	Random Forest	W, REM, Deep	72.58%
Fonseca et al. [40], 2017	ECG	114	CRF (HMM)	W, Sleep	78%

Table 2.2: Performance report of supervised sleep classification studies using classical machine learning methods on ECG data

ables requires large amount of data. This work achieved an average accuracy of 78% with $\kappa = 0.50$. However, it should be noted that this work performs binary classification between the stages due to which the accuracy scores are reported higher than other studies. Secondly, the dataset used contains over 342 recordings which is the largest dataset used so far for sleep classification. Even after that, the authors cite lack of data samples for their lower accuracies. Moreover, over 81 features from ECG and 33 features from respiratory signal were used for classification and yet no feature selection method was employed. This could have possibly overfit the model and caused the high binary classification scores. Having said that, Fonseca et al. have rightly approached the sleep classification problem by modelling it as a sequential state problem. They have shown the importance of employing a sequential model in dealing with transitional and non-separable features.

It is quite evident from the above studies that the classical machine learning techniques failed to produce convincing classification results. Most of them suffer from the fact that the classifier is too simple to separate the non-linearity among the 4 main classes of sleep staging. Better performances have been reported after combining *REM* and *Deep* into a single stage or by reducing the number of classes. This can be attributed to the fact that the features extracted were not fully encapsulating the problem space. Even after the extraction of large number of features in [39] and [107], the non-separability still exists. Either the number of features were not enough or the quality of features was not good enough to represent the feature space. Furthermore, the feature engineer step requires good amount of domain knowledge to determine which feature is significant enough for the problem. Lastly, it is clear from the studies that sleep is highly dependent on time and sleep staging varies over the night. However, apart from using time-based a-piori class weights and using HMM, none of the studies attempted to treat sleep staging as a time-variant problem. The classifiers used treated the features as a heap of feature points, randomly selecting them for classification.

Table 2.2 gives an overview of the performances obtained by various studies using classical machine learning methods on ECG and other invasive sensor data.

2.3.2. Supervised classification using non-invasive, wearable sensors

Devot et al. [32] conducted their study on sleep classification using textile-based ECG sensor. The ECG sensor was stitched into the mattress and the pillow case and data was recorded for 30 subjects in reference to the signals from PSG data. The goal of the study was to show that the ECG signal coverage from textile sensor is as good as the ECG sensor from the PSG. The authors employed hidden Markov model as the sequential classifier to classify between REM and NREM sleep. They followed the standard procedure of extracting RR sequences from the ECG data and then extracting the spectral features - VLF, LF and HF. These spectral HRV features were shown to able to differentiate the REM and the NREM sleep. The study achieved a mean coverage of 81.5% with REM stage and NREM for 18.5%. However, these accuracies do not give a complete picture of the sensor's usability since the movement artefacts in textile sensor was not removed or dealt with.

Another similar study was performed by Kortelainen et al. [73] using a commercially available bed sensor called *Emfit*[1]. The Emfit sensor works on the principle of BCG. Since

[1] https://www.emfit.com

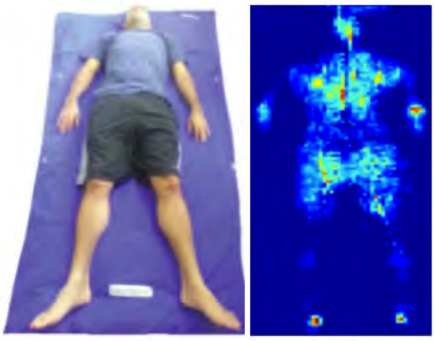


Figure 2.7: Experimental setup based on the pressure-sensor sheet as employed by Samy et al.'s work [111]

the authors aim to extract HRV features, they incorporated a different algorithm to detect the RR peaks from the signals since the nature of the peaks from the BCG sensor differ from the peaks found in ECG. The authors used Fourier transform based Cepstrum method to identify the exact periodicity of the signal and performs convolution with the BCG signal. Due to this, the lag between each peak is clearly visible and makes it easy to identify the true peaks from the false ones. However, even after performing peak-detection, the study employs a semi-manual method of removing false peaks from the signal. For extracting features, the authors used time-variant autoregression model (TVAM) so as to take the non-stationarity of the signal in consideration. Later, standard HRV-based spectral features were extracted from the parameters of the autoregressive model. HMM model was used as the classifier and the classification was performed on 18 recordings. As a result, the study achieved an accuracy of 70% for detecting REM and 74% for detecting NREM . With the addition of movement signal, the results were shown to improve for wake detection at 81%. This study very well displays the potential for a home-based sleep monitoring tool given the state-of-the-art algorithms and accuracies obtained for detecting RR peaks and thereby, sleep stages. However, the experiments were conducted in a closed-form where the subjects were asked to try not to move a lot. To deal with the high amount of noise generated by the sensors, the authors fixed a threshold for the movement signal activity. Using such non-adaptive technique to reduce noise may or not may not work in every cases. If the bed-sensor be used for pathological prevention and diagnosis then real-world scenario should be taken into consideration.

To make the data acquisition process more non-invasive, Samy et al.[111] conducted their study on sleep staging using an unobtrusive pressure-sensitive bed sheet. For the study, the authors used a fully lined pressure-sensitive sensors in a sheet. The bedsheet is a matrix of 8192 pressure sensors and is sandwiched by fabric to prevent the external conductance. As the sheet covers the entire body of the subject, the authors make use of movement data as one of the features for classification. Fig. 2.7 shows the experimental setup used in this work.

The features used for classification are: *Respiratory variability*, *respiration rate*, *leg movement* (helps in determining the restlessness of the subject), *body movement* (helps in differentiating between REM and Deep stage), and *posture and body orientation*. The authors extracted over 32 features in relation to the geometric position of the body on the sheet over the sleep duration. The signals were split in to epochs of 30sec before extracting the features. This study compares their BCG data against a 40 hour worth gold data from PSG studies which was performed simultaneously. *KNN*, *SVM* and *Naive-Bayes* classifiers were considered for classification. The *Naive-Bayes* classifier reported the highest classification performance of 72% followed by SVM's 70%. However, it should be noted that the the classes are highly imbalanced, with *Light* stage comprising almost 70% of the data. Hence, a simple *Naive-bayes* classifier, without making any underlying probability assumptions, ends up classifying *Light* more than the other stages. The performance of the unobtrusive bed-sheet in terms of precision and recall is unsurprisingly low.

Lin et al.'s study on studying sleep is based on an entirely different method than the above

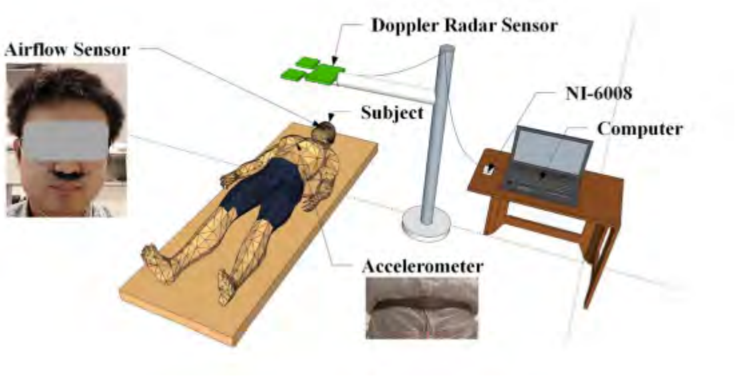


Figure 2.8: Experimental setup of Lin et al.'s [85] Doppler-based sleep sensor

Study	Sensor type	#Features	Classifier	#Classes	Accuracy
Devot et al. [32], 2007	Textile-ECG	3	Hidden Markov Model (HMM)	REM, NREM	81.5%
Kortelainen et al. [73], 2010	BCG	5	HMM and Autoregression	REM, NREM	74%
Samy et al. [111], 2014	BCG	6	KNN, SVM, Naive-Bayes	W, REM, L, Deep	72%
Renevey et al. [108], 2017	PPG, 3D accelerometer	5	HMM	REM, NREM	81.35%

Table 2.3: Performance report of studies on sleep classification using BCG and other non-invasive sensor data

mentioned studies [85]. Lin et al. developed a new sleep-tracking system called SleepSense, which was based on a *Doppler radar* sensor. The Doppler radar sensor can measure the target displacement remotely using the Doppler effect. The sensor placement and the setup of the experiment employed in this work is shown in the Fig. 2.8

The study presents a whole signal acquisition and processing system using the Doppler radar sensor [83]. New features like root mean square, mean crossing rate, energy, MFCC, and sample entropy were extracted from the signal. Also, breathing signal was extracted from the Doppler radar sensor and, together with movement and extracted features, sleep classification was performed. However, this study does not classify the standard sleep stages. Instead, on-bed time, bed exit and breathing are the classes used for predicting the sleep quality. The results achieved by this study cannot be compared with other state-of-the-art results as they do not perform the standard sleep classification as this thesis focuses on, however, this work shows that unobtrusive sleep monitoring can be performed and they have high potential for medical applications if explored more.

Another study on sleep classification using wrist-worn devices for sleep monitoring was recently performed by Renevey et al. [108]. The authors used data from wearable wrist-worn devices like fitness bands and smartwatches to classify sleep stages. The signals used for this study were *photoplethysmography* (PPG) and a 3D accelerometer. The signals were obtained from 10 subjects using a heart monitor smartwatch. An ECG sensor was used as a reference to compare the HRV features. After performing signal processing on the PPG signal, the study achieved an average mean absolute error (MAE) of 9.51 ms in comparison to the ECG sensor. As for the sleep classification, this study achieved an average accuracy of 81.35%. However, the recall score of NREM is around 75%. The lower performance of detecting NREM can be attributed to the insufficient amount of data which has caused class imbalance. Moreover, the performance score of this study shows that wearable data can be used for sleep tracking but they cannot be relied upon for medical applications since the study just focuses on classifying REM and NREM and not the transitional stage of *Light* (NREM1 and NREM2). The reason for the difficulty in identifying the *Light* stage can be because of the lack of robust features that are not sensitive to noise and outliers and have the ability to define *Light* stage.

Table 2.3 gives an overview of the performances obtained by various studies using different machine learning methods on BCG and other non-invasive sensor data.

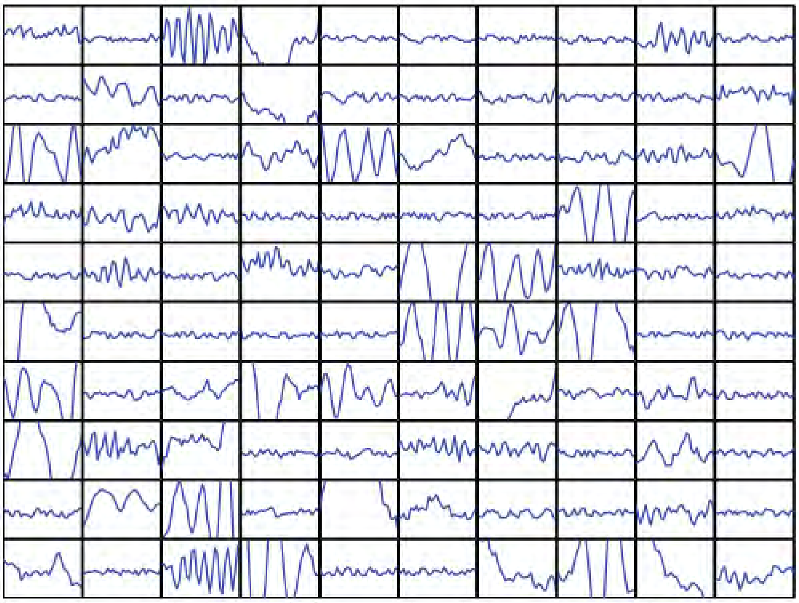


Figure 2.9: By visualizing the intermediate layers of the DBN network, Längkvist et al. [79] show that some of the features learnt by the network correlate with the K-spindles which are usually seen in an EEG waveform.

2.3.3. Deep Neural Networks-based sleep classification

More recently, machine learning techniques, involving artificial neural networks and deep neural networks, attempt to address the above issue of feature engineering. *Self-supervised learning* attempts to build a deep neural net that can extract features by itself with the help of the *Stochastic Gradient Descent* algorithm.

One of the notable studies in unsupervised learning for sleep staging was conducted by Längkvist et al. [79] where EEG, EOG and EMG signals were fed into a *Deep Belief Network* (DBN) for feature generation. The DBN architecture consists of stacks of *restricted Boltzmann machines* which calculate and propagate losses at each layer and thus learn new features. These features are passed to a *Hidden Markov Model* for sequential classification. The authors compare their performance against pre-computed feature extraction methods. The signals were obtained from 25 subjects and each of these signals were preprocessed by filtering out power line disturbances. The signals were downsampled so as to match the number of samples from each of the other signals. The authors reported an accuracy of 67.4% for a DBN architecture which was fed with raw signal values. An accuracy of 72% was reported when 28 handmade features was given as seed features to the DBN.

This study clearly demonstrated the advantages of employing an unsupervised classification pipeline as the results suggested that sleep classification can be performed with the least amount of domain knowledge and feature engineering effort. Fig. 2.9 shows the features learnt by some of the intermediate layers of the DBN network. Having said that, the architecture suffered from uncorrelated features introduced by the parallel DBN layers. DBN is not capable to correlate, say, EOG and EEG signal in this case since it treats them independent of each other. Also, the HMM classification architecture becomes large with higher number of data points and hence overfits the data. Although it is a sequential classifier, it does not have the ability to process longer sequences and remember the states of previous sequences.

To address the problem of sequentially classifying the signals, Dong et al.[34] incorporated a subclass of *Recurrent Neural network* called *Long-Short term Memory* (LSTM). An LSTM network processes data sequentially. The architecture of their model can be seen in the Fig. 2.10. The memory unit in an LSTM makes it able to store the state of the previous states. This accounts for the vanishing gradients over sequences. This study incorporated a Multilayer perceptron (MLP) unit for preprocessing before passing on the processed inputs to the LSTM for classification. Using the MLP layer most of the sparsity in the data and the

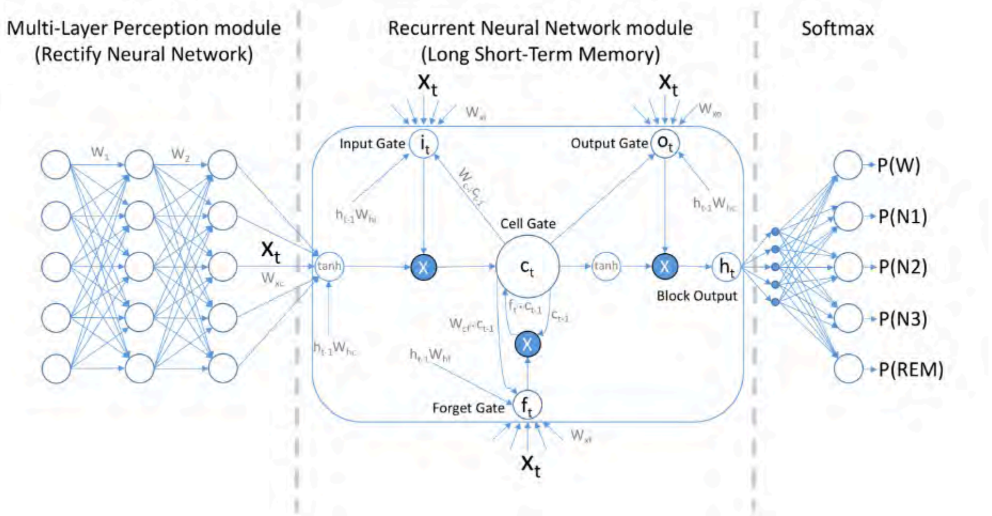


Figure 2.10: Model architecture as used by Dong et al. [34]

noise is reduced and only those features are passed on that have higher activations. This Mixed Neural network architecture was reported to achieve an accuracy of 85.92% compared to other simpler models like SVM and Random forest (79% & 81%, respectively). Although the results seem impressive, it should be noted that the signals used were EOG and EEG signals obtained by placing a single led conductor on the right eye. Firstly, sleep staging being an EEG-based study, it is not difficult to obtain good features and performances using these signals. The characteristic of EEG signal during sleep is well studied and has distinct patterns. Secondly, the study aims at achieving a level of *comfortness* for a subject using this method of signal acquisition. However, attaching conductors, one on the top of eye and the other on the bottom area, does not truly define *comfortness*.

The work by Zhang et al. [139] is based on using one's ECG data as a biometric representation of that subject. To make the neural network understand the subject-specific ECG features, the authors used a 1D-Convolutional network (1D-CNN). Given a particular length of sequence and features associated with each data point in that sequence, a 1D-CNN is able to extract deeper and better features that may not be easily discernible by humans. 1D-CNN is able to process longer sequences as it convolves the sequences with various kernel sizes (or filters). Hence, the authors give a 2 second window of ECG signal as the sequence to be processed. Instead of passing just the time-domain signal, the authors extracted the frequency-domain of each window using *Discrete Wavelet transform* (DWT) thus incorporating blind signal processing. Stack of 1D-CNNs are trained parallelly using these frequency-domain features. This CNN architecture yielded a performance accuracy of 94% in identifying the QRS samples and ID of the subject.

Supratak et al.'s *DeepSleepNet* [121] architecture uses similar concept as that of [139]. The architecture involves two 1D-CNN layers for preprocessing and self-learning of features from EEG signal, followed by a bi-directional LSTM layers. This architecture makes the classification highly robust to noise and generalizes well for different subjects as the bi-directional LSTM layers are able to learn the temporal features specific to the signal rather than learning features that can vary with subjects. DeepSleepNet achieved an overall accuracy of 86% and was comparable to other state-of-the-art-results working on EEG-based EDF dataset.

Chambon et al. [21] performed temporal sleep stage classification using multimodal data from the PSG studies. The authors explored the dependence of sleep on time by modelling their classification network as time-distributed network. In addition to that, they used various sensor signals from the PSG studies as input to their network. This method of using multimodalities for training reduces the feature extraction and separation problem. Multiple convolutional layers are trained on EEG, EOG and EMG sensors and their features are later concatenated to obtain a single classification rule for sleep stages. Using this method

Study	Sensor type	#Features	Classifier	#Classes	Accuracy
Langkvist et al. [79], 2012	EEG, EOG, EMG	Unsupervised	DBN, HMM	W, REM, NREM, L	72%
Dong et al. [34], 2018	EEG, EOG	Unsupervised	LSTM	W, REM, NREM, L	85.92%
Supratak et al. [121], 2017	EEG	Unsupervised	1D-CNN + LSTM	W, REM, NREM, L	86%
Chambon et al. [21], 2018	EEG, EOG, EMG	Unsupervised	1D-CNN	W, REN, NREM, L	87%

Table 2.4: Performance report of studies on sleep classification systems using self-supervised learning (deep learning methods)

of automatic feature learning and multimodal network, Chambon et al. have achieved a mean accuracy of 85% without incorporating time-dependency. After incorporating time-dependency, they achieved an accuracy of 87-90%. Fig. 2.11 shows how the introduction of different time-context values affected the classification accuracy.

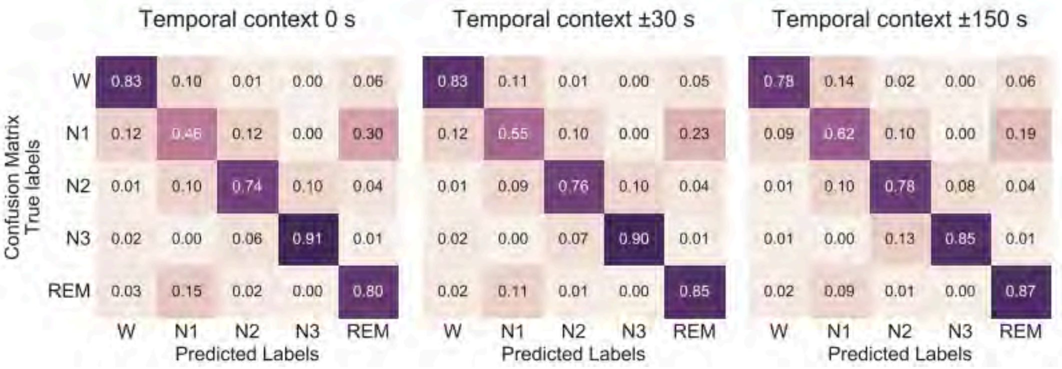


Figure 2.11: Classification accuracy regarding different values of time-contexts as used in Chambon et al. [21]

Although the accuracy reported is comparable to the ones in [121] and [128], the individual accuracies of the stages during transitions is low. This can be due to the fact that simple 2D CNN layers were used to capture the temporal nature, but, more complex networks like LSTM can capture the temporal context in a much better way.

Other similar works have used EEG signals for sleep classification and have achieved comparable results as Supratak et al. [121]. Tsinalis et al.[128] employed similar preprocessing steps as of Supratak et al. but their work dealt in classifying 5 classes instead of 4. Their analysis on CNN filter maps show the features learned by the CNN model. Biswal et al. [11] used spectrogram features and expert-defined features in addition to the raw EEG signals. Their feature and classification sensitivity analysis showed that around 85% of their prediction was in agreement with the PSG stages.

Zhang et al.'s work [139] gives an implication over how a suitable deep neural network such as a 1D-CNN, with frequency-based features, can be used as a preprocessing step for sleep classification. These preprocessed features can be passed on to a sequential-based classifier to further learn spatio-temporal features and perform sequential classification. This kind of architecture mimics the kind of approach employed by an expert in scoring sleep stages. The works by [79], [121], [128], [11] show that deep learning can be useful in incorporating blind signal processing and unsupervised feature learning. The visual analysis of CNN filters, using the internal weights and activation values, in [128] shows that the black-box nature of the deep neural nets can be reduced. The features learned by the network at various activation layers can be visualized and be preprocessed accordingly. Table 2.4 gives a summary of all the studies related to sleep classification performed using deep learning methods.

2.4. Measuring Sleep Quality

”Sleep quality” is a widely used term in sleep medicine and yet its definition has never been established formally. Sleep, being an unconscious activity, is difficult to measure manually and is influenced by many subjective factors like mood, ambient conditions, personal habits, and personality traits [20, 119].

Sleep clinics have attempted to objectively measure the sleep quality of a subject using some key observations from the PSG studies. The most commonly used observations are sleep onset latency (how much time one takes to fall asleep), total sleep duration, sleep efficiency in terms of time spent in REM and NREM sleep, and number of disruptive events like apnea, awakenings or movements. The objective parameters set by Littner et al. [86] for the PSG studies have been the gold standard of objective sleep quality measurement. However, the PSG studies do not take into consideration the subjective factors of the subjects.

The *Pittsburgh Sleep Quality Index* (PSQI) was one of the foremost attempt to quantify the quality of sleep subjectively [16]. Buysse et al. initially developed PSQI to provide the clinicians with a standardized way to categorize users as either ”good” or ”poor” sleepers. PSQI is a self-rated questionnaire which assesses sleep quality over a 1-month time interval. It consists of 19 items or questions that help in evaluating the quality. Each of these items are then combined into 7 clinically-derived component scores, weighted equally from a scale of 0-3. The 7 different components are sleep quality, sleep latency, sleep duration, habitual sleep efficiency, sleep disturbance, use of sleeping medication and daytime dysfunction. These 7 component scores are added to obtain a global score ranging from 0-21. The higher scores in the PSQI indicate having worse sleep quality. The psychometric properties of PSQI have been formally evaluated by Buysse et al. [16], Carpenter and Andrykowski [18] and Cole et al. [25], showing that PSQI has a sensitivity and specificity of 89.6% and 86.5% for identifying sleep disorder cases. Due to its high sensitivity and its ease of administering and scoring, the PSQI has been considered the gold standard for subjective measure of sleep quality.

The other widely used subjective measure includes the *Epworth Sleepiness Scale* (ESS). ESS consists of 8 self-rated items that measure a subject’s habitual likelihood of falling asleep in common situations of daily living. Unlike PSQI, ESS has no specific time frame of data collection. The items are scored on a scale of 0-3, which adds to a scale of 0-24 for a global score. ESS score values greater than 10 are considered to indicate significant sleepiness [63]. Upon testing the reliability and psychometric validation of ESS, it was found to be sensitive towards changes in clinical status, and hence helped in identifying sleep apnea [64, 65].

Although both these methods, PSQI and ESS, are widely used for subjective measurement of sleep quality, they have shown significantly less correlation with the sleep quality reported by PSG studies. Buysse et al. [16] suggested that the lower agreement of PSQI measure with the PSG score can be due to the fact that PSQI score is a culmination score over the past one month. However, the PSG score is only based on a single night’s study. Assuming that a single night of recording can represent a whole month of subjective scoring limits the agreement significantly. Moreover, the invasive nature of the PSG does not allow for long-term or multi-night studies as the subject needs to stay overnight in a sleep laboratory. On the other hand, ESS scoring focusses more on daytime sleepiness than the overnight sleep quality.

With the advances in home-based monitoring devices, long-term sleep monitoring has become a possibility and a good alternative to the PSG studies. Kushida et al. [77] and Kosmadopoulos et al. [74] provided alternative methods to PSG and compared the wrist-worn device’s sleep quality assessment with the PSG’s, proving that sleep wearables can be used as an alternative to PSG for calculating objective sleep quality. However, PSQI’s time restriction introduced bias in terms of clinical and habitual changes by the time objective score was computed. Landry et al. [78] further confirmed in their study that the subjective scores and the objective scores are weakly correlated. As their study was focussed on older adults, they reasoned that, along with the common subjective factors, the cognitive ability of the users also played a significant role in biasing the study.

To overcome some of the problems posed above, Buysse introduced a 5-item self-rated questionnaire called *SATED* [15]. SATED is an acronym for satisfaction, alertness, timing, efficiency and duration. Hence, the SATED framework measures the subjective sleep quality by covering the aforementioned 5 dimensions. Unlike PSQI, SATED does not require any specific time-frame for score collection. Moreover, the focus is on general sleep health rather than identifying serious sleep disorders. Although the SATED framework is not formally validated yet, it has proven to be a simpler method than PSQI and ESS in the Catalan Health Survey conducted in 2017 [31].

3

Datasets and Architecture

The prior works described in Section 2.3.3 show that most of the work on sleep classification using deep learning has been performed on EEG signals than on ECG (heart signal). Since the sleep labelling rules are specifically designed with respect to the waveform of EEG signals, modelling a classification system with EEG signals is easier than with the heart signal. However, EEG signals are difficult to obtain non-intrusively and is done at the expense of the subject's *comfort*. There is a dearth of studies that employ ECG or BCG signal as the input data for performing sleep classification. If a classification system be modelled to learn, extract and classify features from BCG signals, such architecture could prove to be useful for home-based sleep monitoring system.

The goal of this thesis is to design a deep neural network model that could learn the features and classify the sleep stages by using the BCG signal. To design such a classification system, the model should be robust to noise in the input signal, be able to learn new features from the signal, and be able to capture the temporal and sequential nature of the sleep stages. In this chapter, the datasets used for this work are described. Furthermore, the architecture of the model employed in this work is explained along with the implementation details.

3.1. Dataset

This thesis is based on four datasets: *Dozee*'s *BCG* dataset, *Dozee*'s *ECG* data, the *MIT-BIH Polysomnographic* dataset [45, 46] and the *PPG*-based Fitbit data [91] provided by *Fitabase*[1]. In this work, the model is trained using the *Dozee*'s BCG dataset while the *Dozee ECG*, *MIT-BIH PSG* data and the *Fitbit*'s PPG data is used for the transfer learning setting.

3.1.1. Dozee BCG and ECG data

This dataset has been provided by an Indian-based startup company called *TurtleShell Technologies*[2]. *Dozee* is the name of the sensor sheet product designed and manufactured by the company. The sensor sheet contains four pairs of piezo-electric sensors with varying impedance. This sheet is then placed under the mattress, aligning with the chest area of the subject (shown in Fig. 3.1). The pressure differences created by the expansion and the contraction of the subject's lungs (or chest) is captured by the piezo sensors. These pressure values are sent to their server every 4 minutes, where preprocessing and signal processing of the pressure signal is performed.

The dataset used in this study consists of 25 subjects, each of them having 1 or more nights of recordings. A recording, here, means a collection of BCG data of a subject over a sleep duration of one single night. The duration of sleep can range from 6-8 hours. As the

[1] Fitabase: https://www.fitabase.com/research-library/

[2] Dozee: https://www.dozee.io

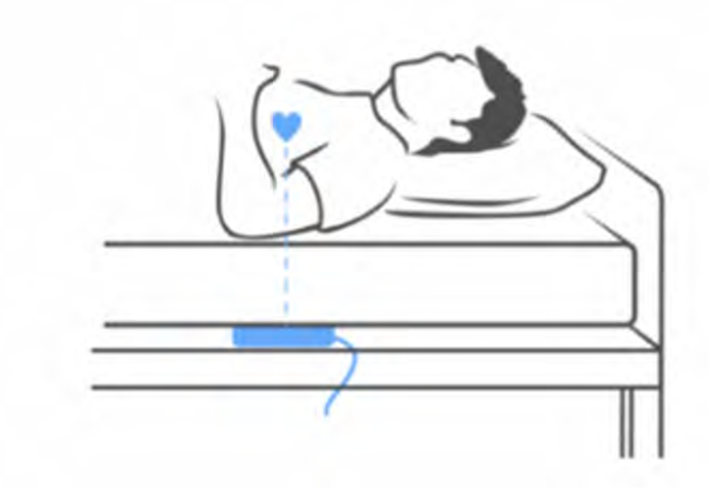


Figure 3.1: Placement of Dozee device for recording heart activity.

BCG signal is obtained at a sample rate of 250 Hz, the number of data samples from a single recording of, say, 7 hours is:

$$250 * 7 * 3600 seconds = (\mathbf{6.3M\ samples}) \tag{3.1}$$

The dataset contains a total of 51 recordings. These recordings have been annotated by 2 sleep doctors from NIMHANS[3] with a kappa agreement of $\kappa = 0.80$. NIMHANS is a premier national research institute based in Bangalore, India alongwith whom these sleep studies have been conducted. For a sleep study, the Dozee device is placed under the mattress in the NIMHANS's sleep lab. The PSG-based electrodes (EEG, ECG, EOG and EMG electrodes) are attached onto the subject. The data from the PSG (Polysomnography sensors) is used by the sleep doctors to visually annotate the sleep stages according to the AASM and R&K scoring rules [106]. The ground truth labels along with the ECG signal data from the PSG study are mapped to the Dozee data using the recording timestamps. The ECG signal from the PSG and the BCG signal from the Dozee sheet are then compared and correlated to verify if the signals have been mapped correctly, in accordance to the timestamps. The Dozee's ECG and BCG data have a heart-rate correlation score of 0.87 as can be seen in Figure 3.2

All the 25 subjects who have participated in the study fall in the age range of 24 - 40 years, with normal to healthy lifestyle. The *Dozee BCG* data is highly imbalanced as the device is started for recording the moment the subject is on bed and is stopped when he is awake. This is in contradiction to the way the PSG recording is conducted by the sleep doctors. In case of the PSG recordings, the recordings start as soon as the electrodes are attached to the user. Hence, there are more number of *Wake* instances in the *Dozee ECG* data than in the *Dozee BCG* data. The complete set of preprocessing steps followed to build the *Dozee BCG* dataset is described in Section 3.2.

For this work, the sleep study was conducted to include only healthy and normal subjects in terms of heart and sleep related disorders. Since it is tough to recruit subjects for sleep studies in clinical settings, some of the subjects were called for multiples studies, with at least a day gap between each study.

In this study, we wanted to understand the subjective sleep quality and wanted to study how it correlated with the objective sleep quality measurement. Hence, to obtain the perceived sleep quality scores, we used the SATED framework. A total of 16 subjects (M=34.5 [range:26-40], Male=10, Female=6) volunteered for the study. Each of these subjects were called to NIMHANS sleep clinic for the study. Once the study is completed the following morning, the subjects were instructed about the SATED scoring and were handed the questionnaire. The questionnaire used to collect the perceived sleep scores in this work is shown

[3] National Institute of Mental Health and Sciences: https://www.nimhans.ac.in

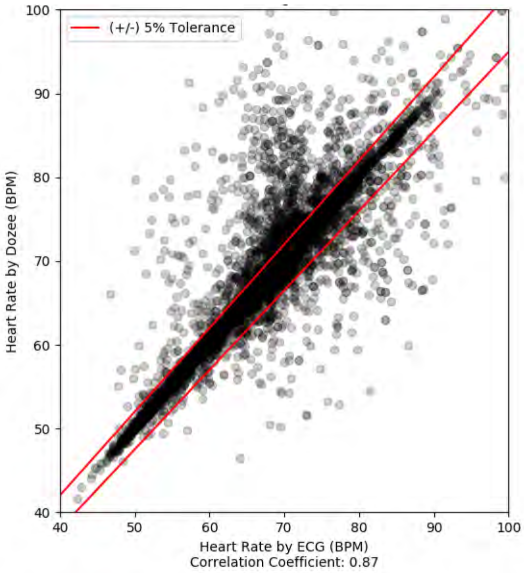


Figure 3.2: Correlation between ECG and *Dozee's* BCG signal

in Figure 3.3. Since the SATED dimensions include questions about alertness and satisfiability, we instructed the subjects to fill in the same questionnaire after a gap of one day from their study. This way we attempted to collect better alertness scores which may not be perceived immediately after waking up. We used the mean SATED scores to test its correlation with the PSG's and *DeepSleep*'s sleep score.

3.1.2. MIT-BIH Polysomnographic data

The MIT-BIH Polysomnographic Database [46] is a collection of recordings of multiple physiologic signals during sleep. The study was conducted at the Boston's Beth Israel Hospital Sleep Laboratory for evaluation of chronic obstructive sleep apnea syndrome, and to test the effects of constant positive airway pressure (CPAP). The database contains over 80 hours' worth of four-, six-, and seven-channel polysomnographic recordings, each with an ECG signal annotated beat-by-beat, and EEG and respiration signals annotated with respect to sleep stages and apnea.

In this study, 16 unique subjects were selected from a population of 60 subjects. The subjects selected consisted of both, sleep apnea affected and non-sleep apnea subjects. The mean age of the subjects was 40 (range:32-56).

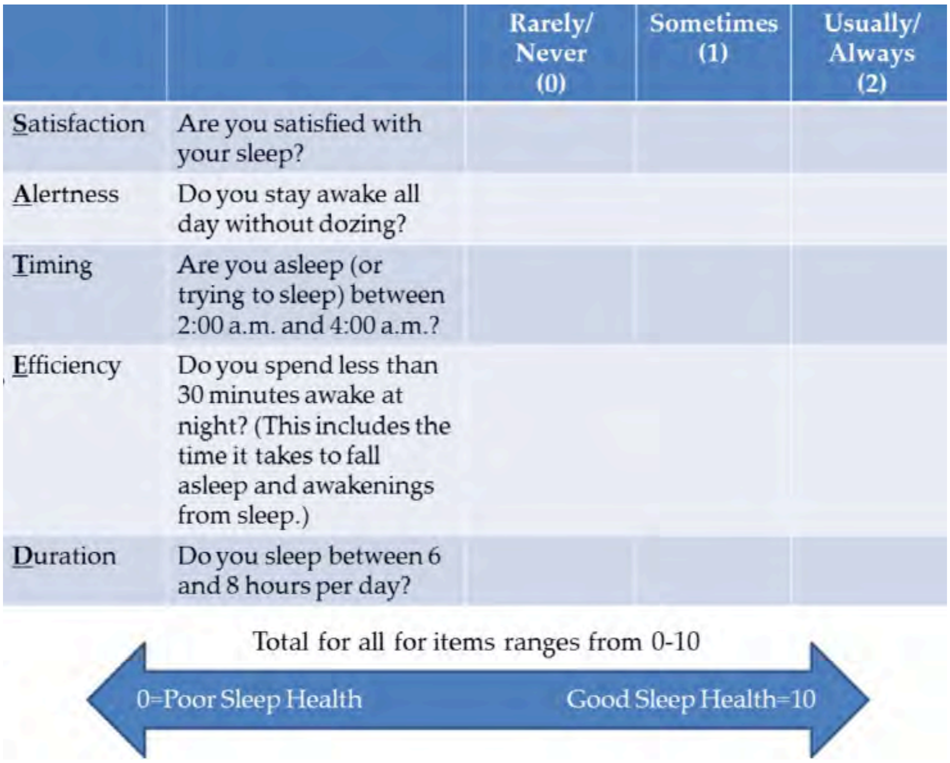
The recording time for each of the subject varied between 2 and 7 hours. The signals recorded were electroencephalogram, electrooculogram, electromyogram of the chin muscle, invasive blood pressure, oxygen saturation, two respiration signals and an electrocardiogram. These physiological signals were digitized at a sampling interval of 250 Hz. A total of 80 recordings were made available by this study [46] for research in sleep monitoring, sleep apnea detection and to understand sleep physiology.

3.1.3. Fitbit-PPG data

The study was approved by the West Virginia University Office of Research Compliance. Participants were 24 healthy adults (40% female, M = 26.1 [range = 19–41] years, 92% white) with no history or symptoms of sleep disorders.

Participants wore a Fitbit device in the product's velcro cuff adjacent to an Actiwatch 64 (AW-64)[4] on their non-dominant wrist during a single night of standard PSG study. Computers used for recording PSG, Fitbit, and AW-64 were time synchronized. Epoch-by-epoch data

[4] Actiwatch 64: https://www.usa.philips.com/healthcare/product/HC1046964/actiwatch-spectrum-activity-monitor



		Rarely/ Never (0)	Sometimes (1)	Usually/ Always (2)
Satisfaction	Are you satisfied with your sleep?			
Alertness	Do you stay awake all day without dozing?			
Timing	Are you asleep (or trying to sleep) between 2:00 a.m. and 4:00 a.m.?			
Efficiency	Do you spend less than 30 minutes awake at night? (This includes the time it takes to fall asleep and awakenings from sleep.)			
Duration	Do you sleep between 6 and 8 hours per day?			

Total for all for items ranges from 0-10

0=Poor Sleep Health

Good Sleep Health=10

Figure 3.3: Sample SATED framework questionnaire used for collecting perceived sleep quality scores.

Dataset	Sensor type	No. of Recordings	Sample rate (Hz)	Wake	REM	Deep	Light
Dozee BCG	BCG	51 (25)	250	8%	22%	25%	45%
Dozee ECG	ECG	51 (25)	250	8%	22%	25%	45%
MIT-BIH	PSG-ECG	80 (16)	250	15%	25%	30%	30%
Fitabase-Fitbit	PPG	12 (4)	120	20%	20%	15%	45%

Table 3.1: Overview of the class representation in each of the dataset used in this work. The ”Number of Recordings” indicates the total number of data that we have for that particular dataset. The number in the brackets correspond to the number of unique subjects from which the recordings were obtained.

were extracted from the outputs of both devices and compared to PSG. Additionally, three participants also wore the two Fitbit devices simultaneously (on the same wrist) overnight at home to assess inter-Fitbit reliability. This dataset was primarily collected by *Fitabase* which specializes in crowd sourcing and collecting Fitbit data using their API. Since the data is directly collected from the user’s device, the sample rate (120 Hz) of the data is much lesser than the above mentioned datasets. *Fitabase* aggregates heart rhythm values for every 30 second time period, thus reducing the quality and the information of the signal. Although the participants for this study were 24 in total, we were given access to only 12 of them due to *GDPR* restrictions.

3.1.4. Summary of datasets

In this work, we use the *Dozee BCG* for training our classifier. *Dozee ECG, MIT-BIH ECG* and *Fitbit-PPG* datasets are used for testing the transferability of our *DeepSleep* model. The distribution of sleep stages in each of these datasets are shown in the Table 3.1 and Fig. 3.4.

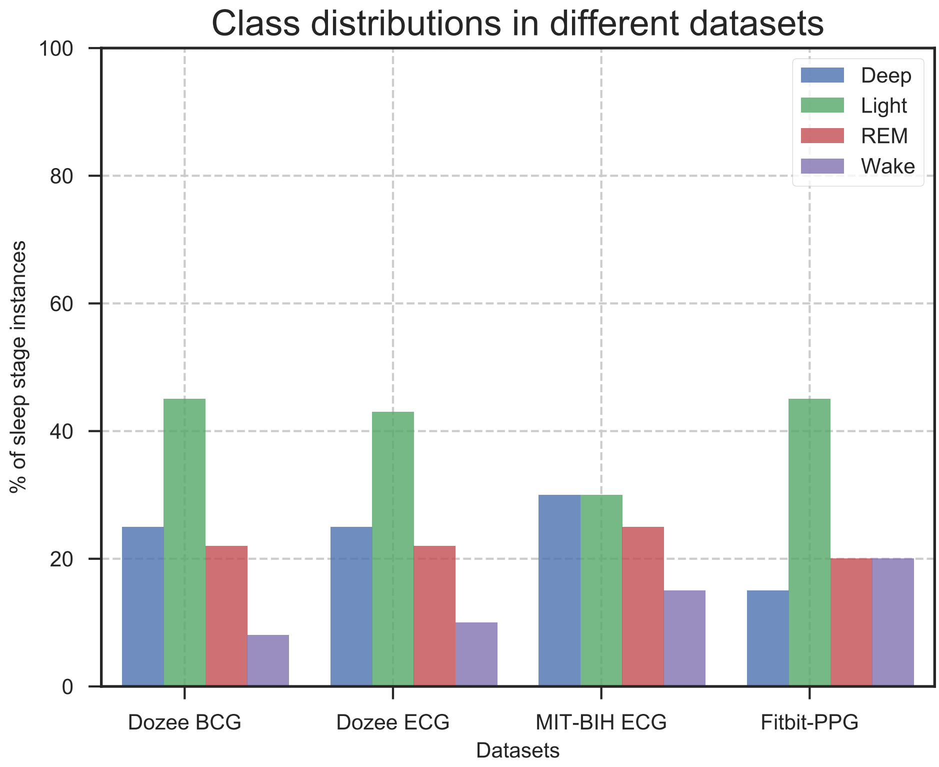


Figure 3.4: Distribution of data in terms of sleep labels

3.2. Preprocessing Pipeline

The *Dozee BCG* dataset is gathered by following these set of preprocessing steps:

Data acquisition
The sensor data recorded by the *Dozee* system is stored in the embedded storage device. The data is collected and sent to the processing server on an interval of 4 min. ASCII data format is used during transmission to reduce the size and the overhead on the server. The ASCII data from the server is downloaded, converted to unicode format and then used for signal processing. The raw sensor data obtained from the device consists of timestamp values, piezo values (pressure values) and ambience values like temperature and humidity.

Signal Processing
The data obtained from the device contains the pressure values. To obtain the heart, respiratory and movement signals from the BCG signal, signal processing algorithms are applied. Signal processing is applied on a non-overlapping epochs of 30 seconds, hence, producing a total of 8 epochs for each of the 4 min file.

A *Butterworth* filter with a low pass of 10Hz and high pass of 1Hz with an order of 4 is applied to get the heart signal. A similar filter with a low pass of 0.9Hz and a high pass of 0.1Hz is applied to obtain the respiratory signal. For the movement signal, a sliding average is applied on the BCG signal and any signal that crosses a threshold is considered a movement. The sliding window and the threshold is set heuristically upon many trial and error runs. However, for this work we use the heart signal only.

Sequence formation
The heart signal obtained from the BCG and its features are converted into sequences of 30 seconds. These sequences are non-overlapping samples of length 7500. The reason for

choosing a sequence length of 30 seconds is to replicate the sleep scoring rules stated by AASM and R&K [106].

Preprocessing and storing

As a final step, the samples are checked for missing, infinity and null values. These samples are removed from the dataset. As the data is a mixture of different subjects, the dataset is standardized to have a central mean. The *Dozee* sheet consists of 4 pairs of piezo sensors of different impedance values and hence, the data obtained in an interval of 30 seconds also needs to be normalized. These two preprocessing steps reduce the inter and intra-variance in the subjects. Finally, to reduce the computational time of this processing pipeline, the processed data along with features and labels are stored in a compressed numpy array format called *.npz* [101]. This allows for reusability of features, faster loading into arrays, persistence of data and reproducibility of the results.

3.3. Model Architecture and Design choices

To build a model that can learn features from the raw BCG signal and capture its temporal nature, we need to use the learning algorithms that have the capability to extract features and treat the input data sequentially.

Figure 3.5 shows the architecture of our *DeepSleep* model. The model consists of two important layers: *Representation learning layer* & *Sequential layer*. The *representation learning* layer consists of stacks of *1-D convolutional networks* (CNN) which have been proven to be good at extracting and learning features from temporal data [80]. Whereas, the *sequential layer* consists of stacks of *bidirectional LSTM* (Long-Short term Memory) networks. The LSTM networks explore the sequential nature of the sleep-related features [49]. By using LSTMs we aim to enable the model to learn the sleep classification problem and tackle it the way a human sleep expert does - labelling the sleep stages based on the knowledge of the previous stage. The working and the parameters of the *representation layer* and *sequential layer* are explained in detail in Section 3.3.1 and 3.3.2.

The *Dozee BCG* data is quite *imbalanced* in terms of representations of classes, as can be seen from Figure 3.4. Moreover, annotating ground truth labels for healthcare data (in this case, sleep classification) is an expensive, complicated and limiting task. Hence, to tackle these issues, one needs to approach the problem with a training strategy that is different from one-shot training. A novel strategy needs to be employed that could make use of the limited amount of labelled data without degrading the model's performance. Hence, in this study, we employ a data oversampling method on the BCG data, followed by a 2-step training process - *pre-training* and *fine-tuning*. These steps are further described in the following sections.

3.3.1. Representation learning

The *representation* layer consists of stacks of 1D-CNN layers. The architecture of this layer is inspired on the End-to-End speech recognition architecture by Graves et al. [50]. While designing the *representation* layer, some important issues were taken into consideration:

1. The layer should be able to learn and extract features on a sample-level BCG signal.
2. The model is able to learn both, temporal and frequency-based features, without much feature engineering
3. It should be designed to work for the transfer learning setting.

Considering these points, we built the representation layer consisting of blocks of CNN layers. Each block comprises of 2 CNN layers, followed by a *batch normalization* layer [60] and a *ReLu*-based activation layer [28, 92] for each of the CNN layer. Figure 3.6a shows the structure of a single CNN block in the representation layer.

The convolutional layers extract features from a localised patch of the input data rather than looking at the whole input at once [70]. That is one of the reason why a CNN has

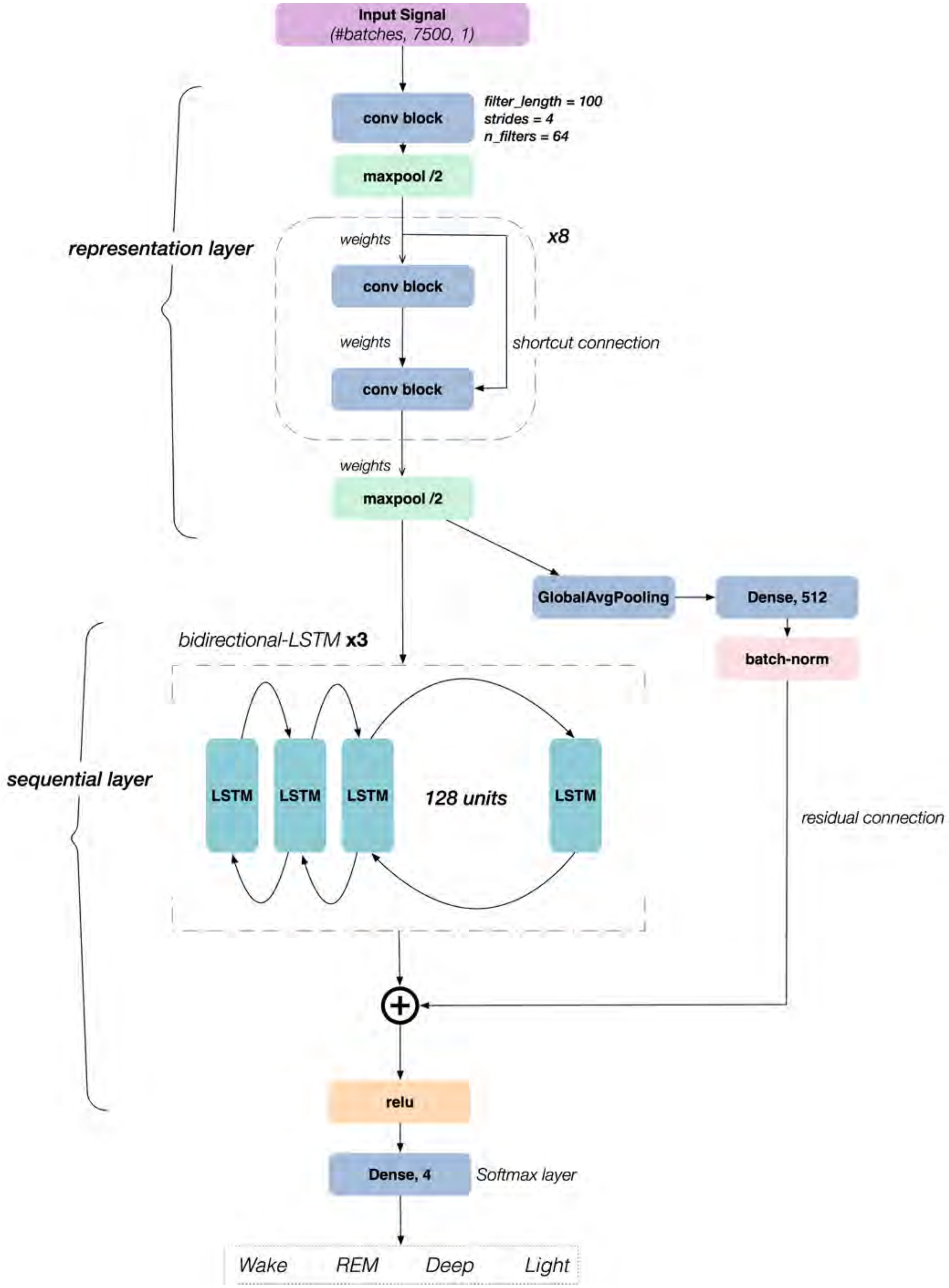
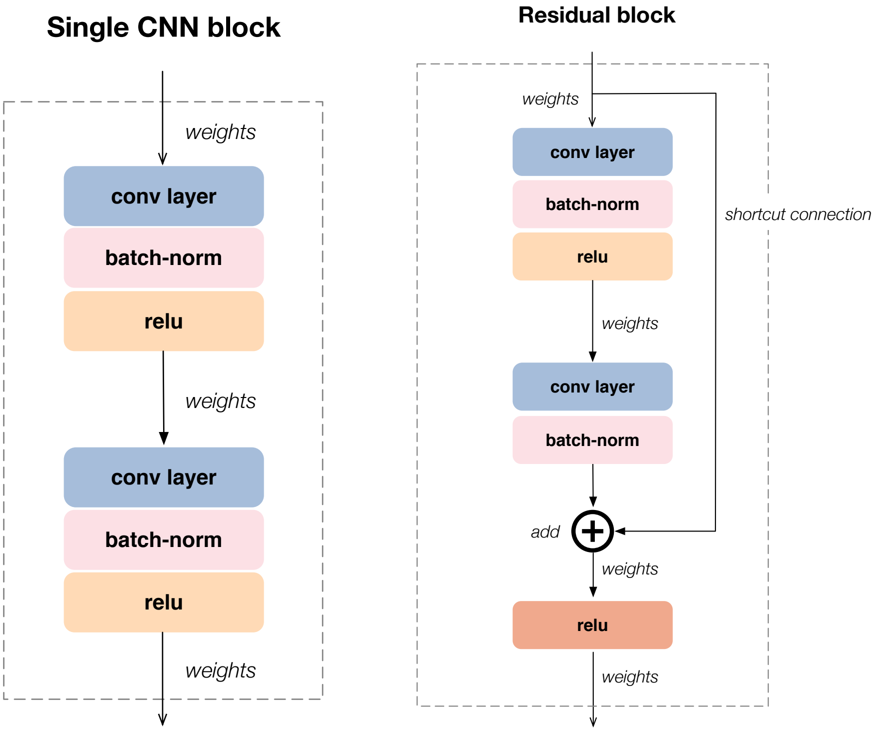


Figure 3.5: *DeepSleep* model architecture with residual connections.



(a) Structure of a CNN block.

(b) CNN block with a residual connection.

Figure 3.6: Structure of a single CNN block. Every CNN block consists of two 1D-CNN layers, batch normalization layer and an activation layer (in that sequence).

way less number of parameters to learn than compared to a multilayer perceptron. This localised patch of information is determined by the filter size and the stride length by which the filter is translated forward. This nature of the CNN allows us to apply it on a 1D data to preserve the data's temporal property. However, a CNN has no provision for storing the internal states of the sequences that it has already seen. Due to this, CNN could be used to extract features from a temporal signal but it cannot account for the sequential nature of the sleep classification. Hence, the features extracted by the CNNs are later given as input to the *sequential* layer so as to incorporate the sequential property. Given this property of the CNN layer, the convnet extracts features for every instances of a class it comes across. In simple terms, it means that the CNN extracts the same features multiple times for a particular class, thus leading to duplication and over representation of the data. If not controlled, this behaviour would lead to over-fitting of the data. To tackle this issue, we use stacks of CNN instead of just one CNN layer and a *max-pooling* layer [14] after every stack. A stack of CNN would make the network learn to differentiate between a lesser and higher correlated features among the *general* features learned in the first layer of the CNN stack. By stacking CNNs we ensure that the features extracted in each layer is further refined for higher activation [90].

Choosing an appropriate number of blocks or levels of CNN layers is an important parameter tuning task. Deciding on the number of CNN layers is dependent on factors like input signal type, size of the data, number of model parameters, training time, and convergence time. To select an appropriate number of blocks for the *representation* layer for our domain, we experimented with the depth of the *representation* layer by performing training on our BCG data. The layer depth or the number of blocks that yielded us the best performance with least overfitting was chosen for our model. Table 3.2 shows the effect of the depth of the *representation* layer on the training accuracy, validation accuracy, model parameters, and the training and convergence time. It can be seen that 8 blocks of CNN, i.e 16 layers of CNN, yielded us the best performance for our input data. To further regularize the overfitting nature of our model we introduced the *Dropout* layers, with a dropout value of 0.3, after every two blocks of CNNs.

No. of Blocks	No. of CNN layers	No. of Parameters	Output dimensionality	Mean Training Acc.	Validation Acc.	Epochs till overfit	Training time (approx.)
4-blocks	8	6M	128	57%	20%	150	4hr
6-blocks	12	11M	256	80%	40%	30	4hr
8-blocks	16	18M	512	67%	64%	100	10hr
8-blocks - residual connection	16	18M	512	69%	67%	120	11hr
10-blocks	20	36M	1024	91%	49%	75	14hr
10-blocks - residual connection	20	36M	1024	88%	60%	48	17hr
12-blocks	24	63M	1024	98%	55%	40	16hr
12-blocks - residual connection	24	63M	1024	95%	42%	37	23hr

Table 3.2: Effect of *representation* layer's depth (no. of CNN layers) on the model's performance.

Eight such blocks of CNNs are stacked on top of the other to form the representational layer. The first CNN layer is different from the rest since it determines the representational space of the input data to be calculated for the following CNN blocks. To ensure that the layer learns both the temporal and the frequency-based features, a broader *filter size* of 100 is chosen for the first CNN layer. Filter size of 100 denotes that the CNN performs convolution on every 100 samples of the input data sequence. These 100 samples translate to around 0.4 seconds of the input signal, which is normally the size of the input signal used to calculate frequency-based features that were described in Section 2.1.2. The rest of the CNN blocks have a fixed *filter size* of 16. A narrower value of filter size is chosen for the rest of the blocks so that the CNN layers extract features from the smaller and detailed sets of the input signal. This also ensures that the model learns the finer temporal features in the deeper layers.

Batch normalization [60] is used after every convolutional layer so that the weights learned in each of the CNN layers are normalized and centered around the mean. Additionally, a *rectified linear unit* (ReLu) [28] is used as the activation function for the 1D CNN layers.

By adding more stacks of convolutional blocks the input signal is represented better as the output dimensionality increases with depth. The initial output dimension (the number of filters) for the first CNN layer is set as 64. This filter number is doubled after every 2 blocks of CNN. Hence, with an 8-blocked layer, the output dimensionality comes to be 512. Although the higher dimensionality space is able to represent the signal well, it also means that there are significantly high number of layer parameters that need to trained. This causes overfitting on the data and also causes the lower blocks to receive diminished weights and gradients from the higher blocks of the layer [54]. To avoid this issue, a *shortcut-connection* or a *residual* connection is introduced as recommended by He et al. [54]. Fig. 3.6b shows the structure of the residual connection employed in this work. By using *shortcut* connection, the weights learned in the previous block are concatenated to the next CNN block. This way, no additional training parameters are introduced and the weights learned in the higher blocks are easily propagated to the lower blocks. Fig. 3.7 shows the complete expanded architecture of the representation layer. As mentioned earlier, 8 blocks of CNNs are used for extracting features. The input's sequence length, filter size, number of filters (output dimensionality) and the usage of convolutional residual connections can be seen in this figure. A convolutional residual connection simply uses a single strided CNN layer to subsample the signal from the previous block before adding a shortcut connection to the next block. This ensures that similar output dimensionality and input sequence is maintained throughout the training phase. The *Global average pooling* layer and the *Dense* layer are only added during the pre-train phase.

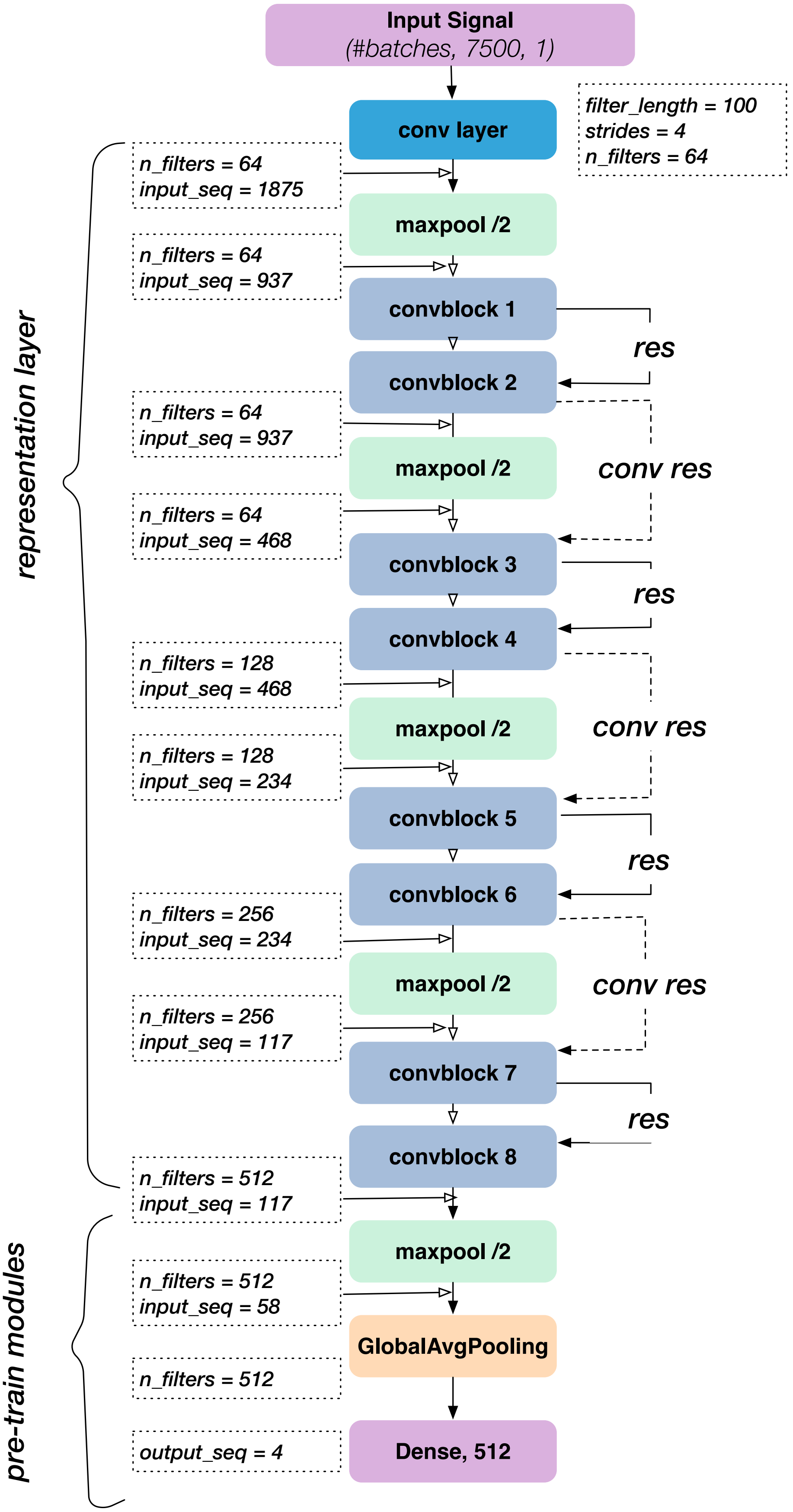


Figure 3.7: Expanded architecture of the representational layer. 8-blocks of CNNs are used in this work to perform feature extraction task.

3.3.2. Sequential learning

Since sleep staging is a temporal and sequential problem it requires a different class of learning algorithms – algorithms that treat them in their truest form. Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks are one such learning algorithms that treat the input sequentially by keeping a note of all the states that have been seen in the history [110]. An RNN unit accepts an input and generates an output for it. When processing the next sequence of input, it makes use of the output of the previous sequence to make a decision about the current sequence. This behaviour of an RNN allows it to, in theory, store infinite states of sequences. It can make predictions for the present input sequence based on the historical states that have been learned and stored. However, in practice, an RNN has been shown to perform poorly when fed with longer sequences of input. With the increase in the length of input sequences the RNN starts to ”forget” the sequences seen farther in the history. This problem is known as the *vanishing gradients* problem. To address this issue LSTM was developed as a variation to the RNN. An LSTM layer has a *forget* unit in addition to the state memory. This unit lets the LSTM to decide after every sequence if the state needs to be stored or ”forgotten”. This lets the LSTM layer to process longer sequences without any decrease in training performance. Having said that, an LSTM is highly sensitive to its initialization of the weights. When wrongly initialized it destabilizes the gradients of the LSTM units thus making them get stuck in the local minima. Also, LSTMs are good at learning the transitional features but are not good at extracting the common features. To address these shortcomings, this work performs sequential training only after loading the pre-trained weights.

This work uses a variation of an LSTM network called *bi-directional LSTM* (bi-LSTM) [114]. The *sequential* layer consists of three *bidirectional LSTM* layers stacked on top of the other. 256 units of hidden layer are used in each direction of a bidirectional layer. Thus, each level of a bi-LSTM extracts 512 dimensions of features. Additionally, it is important to note that the first two stacks of the bi-LSTM give a *sequence* of 512-dimensional features as output to the third level instead of passing a single vector of the output state. By doing this we maintain the temporal order of the signal while it is being processed on to deeper levels. Moreover, since we concatenate the heart signal of all the subjects and pass them as a single, continuous signal to the model, it is important that the model is informed when a signal from a different subject arrives. If not informed, the bi-LSTM would continue to store the states of the sequences until the end of the data. This, in principle, does not reflect the right way to understand the sequential nature of the signal. The sequential and temporal property of sleep stages is only valid and unique to a specific subject. Hence, to tackle this, we reset the internal states of the bi-LSTM layers after it trains one whole recording. This way the states from one recording are not propagated to the other recordings. The weights of the bi-LSTM is still shared by the whole network till the end of the training.

Additionally, Fig. 3.5 indicates a sideward connection from the output of the representation layer to the output of the bi-LSTM layers. This connection is called a *residual connection* [54]. Residual connection is incorporated in our sequential layer because of the fact that the features extracted in the representation layer do not account for sequential transitions. As noted in the previous section, the representation layer extracts features from the heart signal but it does not take the signal's sequential property into consideration. However, the sequential layer does learn these transitional properties of the sleep stages. By using a residual connection we combine the features extracted in the representation layer with the transitional features learnt in the sequential layer. We use a residual connection comprising of a 512 units of a fully-connected dense network and later concatenate the result with the output from the second level of bi-LSTM layer.

The concatenated features are then passed on to a *Dense* network of 1024 units followed by a *Dropout* layer to reduce overfitting. A dropout value of 0.3 has been used through out the work. This means that the network would randomly drop 30% of the units during the training. In the end a *Softmax* layer is used to perform the final sleep stage classification.

3.3.3. 2-phase training: Pre-training and Fine-tuning

One of the most important aspect to be considered while training a model is that it should not be bias towards any particular class in a multi-class classification system. Usually such biases are introduced due to non-uniform distribution of data, and sometimes, due to the nature of the problem itself. There are various preprocessing techniques that ensure that the model does not learn such biases. Some of the commonly used techniques are oversampling or undersampling, assigning weights to classes, assigning weights to the samples and using weighted losses that give importance to lesser represented classes during the training.

Although the above mentioned preprocessing techniques help in reducing the bias in the learning model, the preprocessed data usually does not entirely enrich the original dataset. For example, the artificial samples generated after oversampling do not encapsulate all of the characteristics of the original data. Either it just duplicates the original samples or computes the closest mean value by looking at nearest neighbours. This introduces variance into the dataset. Such form of variances are highly unstable for learning models like LSTMs. A naive oversampling method disturbs the temporal and sequential nature of the heart signal. And, since the LSTM network depends on the temporal properties of the signal, its weights become highly unstable during training [44].

To alleviate this problem, a 2-phase training strategy is used in this work. Instead of training the entire model with our imbalanced data, we first randomly oversample the training data and *pre-train* the *representation* layer. The pre-trained weights are then used to extract features from our original dataset which are given as input to the *sequential* layer to *fine-tune* the model. There are two main reasons why 2-phase training is used while training a neural network model [10].

1. **To efficiently use the limited data samples**: Obtaining large amount of labelled data is always a difficult process. To make use of the limited dataset and still obtain good training accuracy, the model is first trained on a slightly oversampled dataset. Since the model is already trained once using the mixture of original and synthetic data, the network learns the weights without having to depend on randomly initialized weights. This increases the stability of models that are highly sensitive to changes in weights, especially models like RNN and LSTM.

2. **To make use of abundant dataset from different source but of the same problem domain.** In principle, a model can be pre-trained using a dataset that is available in large quantity [50]. The pre-trained weights can then be applied on the dataset that we need to predict on. It is important to note that the feature domain of the two datasets need to be highly correlated for this strategy to work. For example, for sleep classification, since the work is based on predicting sleep stages based on the heart signal obtained from the BCG data, the model could be pre-trained using another heart signal-based dataset that is higher quantity than the BCG one. As the feature domain is based on the heart signal, as long as the heart signal from BCG is correlated with the heart signal from ECG or PPG, this *transference* of weights is possible.

In this work, we use an oversampled dataset using the *random oversampling* technique to create equal distribution of all the classes.

3.3.4. Implemention

For the pre-training phase, the input signal used was the raw heart signal obtained from the BCG data. This signal was oversampled using *random oversampling* such that all the classes except the major class were oversampled to match the number of samples in the major class. This oversampled data was then divided into sequences of 30 second signal length. As the sampling rate of the *Dozee* dataset was fixed at 250Hz, the sequence length was set as 7500 samples. These sequences of the oversampled signal were pass ed on as the input for pre-training. To allow for robust and unbiased feature extraction it was necessary to shuffle the batches of sequences before every iteration of pre-training. It may be noted that although shuffling the data would disrupt the sequential order of the signal, this would not affect the

CNN's feature extraction since it does not account for the sequential ordering. The sequences within the batches itself are left intact and the ordering of the batches are shuffled. Hence, the CNN learns the temporal order within a batch, which is more important in our case. This also shows that the setting an appropriate *batch size* is an important parameter-tuning task as the gradient update and feature learning is dependant on the number of samples received by the model in a single training pass. The lack of support for sequential ordering is later accounted for during the *fine-tuning* phase. In our case, we found the *batch size* value of 128 to be giving us good results in terms of model convergence and computation time. Essentially, a batch size of 128 on the BCG data means that a single batch contains around 64 minutes of the input signal as each sequence is a 30 second epoch.

Algorithm 1 Pretraining

```
procedure initialize pretrain-step
input signal
    $data_{seq} \leftarrow input\_signal$
    $data_{seq} \leftarrow oversample(data_{seq})$
    $data_{seq} \leftarrow shuffle(data_{seq})$
initialize repr_layer
    $model \leftarrow init\_cnn$
    $lr \leftarrow 10^{-2} : 10^{-4}$
    $model \leftarrow compile(\mathit{Adam}_{lr}(repr\_layer))$
for 1:n_pretrain_epochs
    for batch in $data_{seq}$
    $model \leftarrow \mathit{Adam}_{lr}(model, batch)$
    $model_{logits} \leftarrow \mathit{softmax}(model)$
    $save(model\_weights)$
    $del(softmax\_layer)$
    return model.
```

For pre-training, a split of 80-20% for training and validation set respectively was used. As can be seen in Fig. 3.7, during the pre-train phase, the representation layer is attached with a *Global Average pooling* [56] layer that averages the model weights and features over the time dimension of the input signal. The output of this layer is then is given as an input to a *Dense softmax* layer (shown in Eq. 3.2) for performing classification. The *Dense* layer classifies the 30sec of a signal into one of the four sleep stages and the performance of the pre-training phase is monitored on an oversampled validation set. The best of the learned and adjusted weights, in each of the pre-training iterations, are stored for *fine-tuning* phase. This *softmax* layer is only used for monitoring the performance of the pre-training layer and it is removed when running the *fine-tune* model. *Dropout* layers have been used aggressively throughout the pre-training to control the overfitting of model. In our case, a dropout value of 0.3 has been found to perform well for the model. This means that after each CNN layer is processed, 30% of the units are randomly dropped, thus reducing the number of parameters are preventing overfitting.

$$S(y_i) = \frac{\exp^{y_i}}{\sum_j \exp^{y_j}} \tag{3.2}$$

An *Adam* optimizer [71] with a learning rate of 10^{-2} was used for optimizing the gradients during the pre-train phase. The pre-training is run for 100 epochs with an option to terminate the training early if there is no improvement in validation loss for a minimum of 20 epochs. The training loss and validation loss are monitored after every epoch. By monitoring these metrics we determine when to reduce the learning rate. The learning rate is reduced by a factor of 2 every time the validation loss ceases to improve for 5 epochs. A limit of 10^{-4} is set as the least value the learning rate could take during the pre-training phase. At the end of the pre-training, the model weights are saved so as to be used for the *fine-tuning* phase. The pseudocode followed for the pre-training process is given in Algorithm 1. The

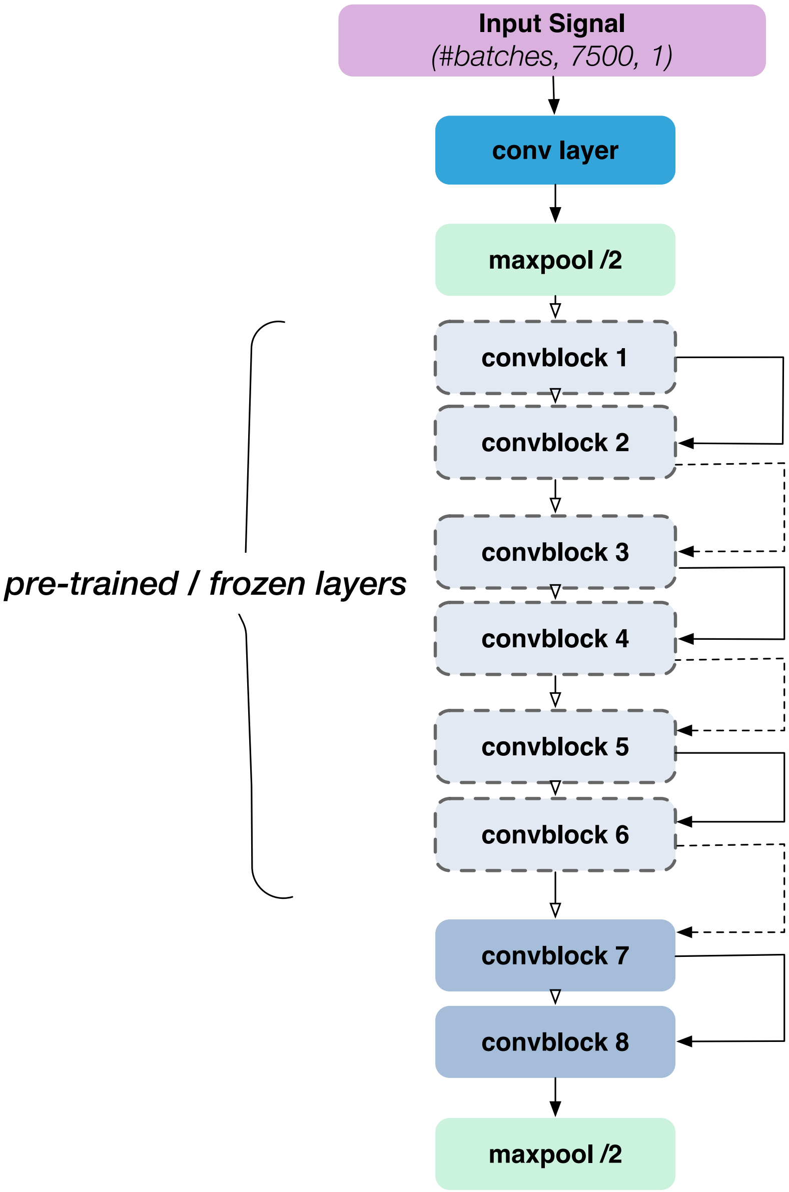


Figure 3.8: Depiction of the training strategy used for fine-tuning and performing transfer learning on our pre-trained model.

pre-training phase was run until 100-150 epochs. Our hyper-parameter tuning showed pre-training improvement only till 100 epochs though.

One important thing to note during the fine-tuning phase is that the learning rate should be much lower than the rate used in pre-training [137]. Fig. 3.8 shows the weight initialization and training strategy employed in this work for fine-tuning and applying transfer learning. We initialize our model with the pre-trained weights and make the initial 6 blocks of CNN as non-trainable (frozen layers). To fine-tune these blocks, it is important that the new weights learned while fine-tuning do not disturb the distribution of the pre-trained weights. Hence we use a lower learning rate so that the weights are updated at a slower rate and on top of the pre-trained weights. In this work we set a learning rate of 10^{-5} while fine-tuning. Additionally, we used a non-adaptive optimizer like *SGD* (Stochastic Gradient Descent) [13] instead of adaptive optimzers like *Adam* or *RMSProp* [125]. *SGD* optimizer by itself does not decay the weight and the learning rate upon reaching a plateau in the performance. Its momentum can be fixed and controlled so that a fixed learning rate is used throughout the fine-tuning phase. Algorithm 2 gives an overview of the programmatic pseudocode followed for *fine-tuning*.

Algorithm 2 Fine-tuning

```
procedure initialize finetuning
weight initialization
    $conv\_weights \leftarrow load(pretrained\_weights)$
    $lr \leftarrow SGD(lr = 10^{-5})$
initialize model
    $model \leftarrow SGD_{lr}(repr\_layer, seq\_layer)$
for 1:n_finetune_epochs
    for batch in sequence_data
    $model \leftarrow SGD_{lr}(model, batch)$

    if batch from new_recording then
        $model \leftarrow reset\_states(biLSTM)$
    return model.
```

The model was pre-trained on a GPU cluster. The model was trained parallely on two *Nvidia 1080-Ti* GPUs. This is to ensure proper management of distributed computing and memory resources. The model was programmed using the deep learning framework, *Keras* [24], with *TensorFlow* [2] as the backend.

Fig. 3.9 shows the categorical loss and categorical accuracy recorded during the pre-training phase. It can be seen that the training accuracy improves steadily over the number of epochs but the validation accuracy is quite unstable during the first half of pre-training. The initial jitter in the validation accuracy can be explained by the fact that the model is still trying to learn the right weights that could generalize well on the validation data. The second jitter in the validation accuracy occurs around the epoch 40-50. This is due to the fact that during this time the learning rate of the pre-train model is reduced as the validation accuracy and loss have stagnated. It can also be seen that the validation loss begins to diverge away from the training loss during the epochs 40-60 as the model adjusts its weights to the new learning rate. Here, it looks as if the model is overfitting but thanks to the adaptiveness of *Adam* optimizer and the implementation of dynamic learning rates, the loss functions begin to converge steadily later.

Unsurprisingly, the validation loss and accuracy is around 1.15 and 0.67 respectively, by the end of the pre-train phase. As the CNNs in the *representation* layer are very good at extracting features but not at classifying sequences, this stagnation of validation loss and accuracy scores suggest that the model could be further fine-tuned.

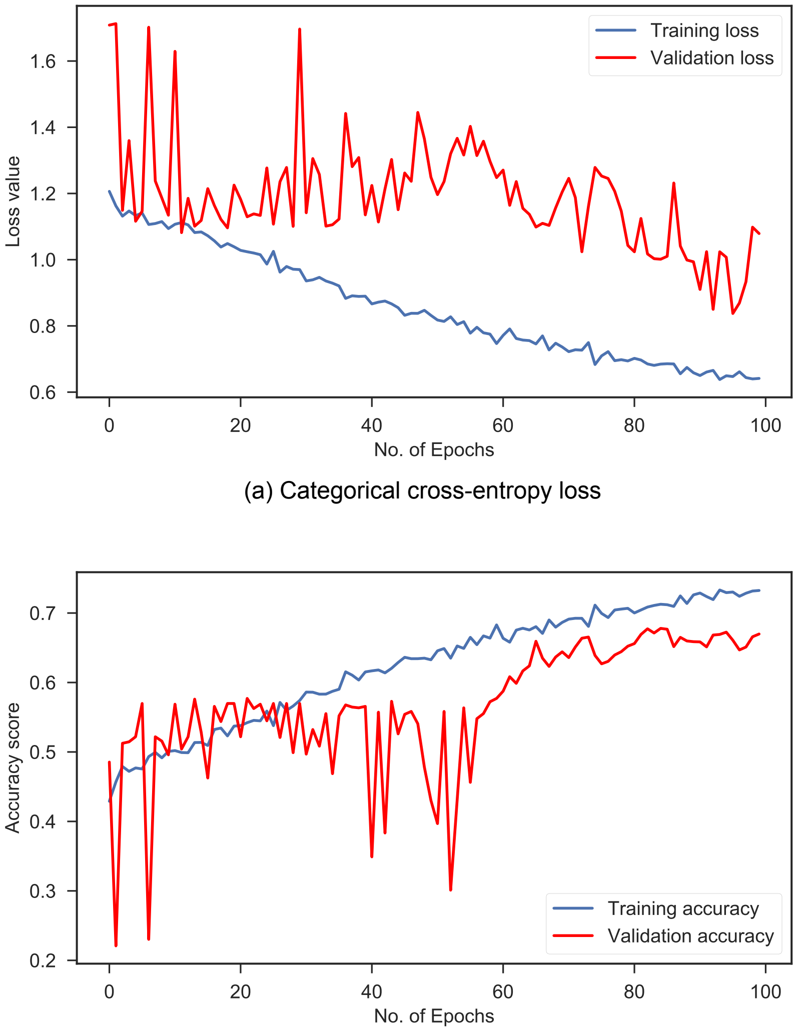
During the *fine-tuning* phase, the entire model, i.e the *representation* layer & the *sequential* layer, is trained using the original *Dozee BCG* dataset. However this time, the *representation* layer is initialized with the pre-trained weights.

Fig. 3.10 shows the the variation of model's categorical loss and accuracy during the fine-tuning phase. It is evident that the convergence of the training loss is steady which could be attributed to the initialization of pre-trained weights. The validation loss does not show signs of getting stuck in a local-minima (stagnation) and converges smoother than it did during pre-training. Also, the validation loss and accuracy has improved from the pre-training.

3.3.5. Transfer learning

Our *DeepSleep* model, trained on the *Dozee BCG* dataset, is tested on the *Dozee ECG*, *MIT-BIH ECG* and *Fitbit-PPG* dataset. The weights learned from a *source* signal (here, the BCG signal) is applied on the *target* signal to learn a similar task. Since our task of classifying sleep stages depends on a heart-based signal, we hypothesize that transfer learning could be possible in this scenario.

The pre-trained weights obtained using the BCG data were used for testing the transferability of our model to other signal types like ECG and PPG. Transfer learning setting follows similar implementation process as specified in Section 3.3.4. However, instead of pre-training the model using the ECG and the PPG, we initialize the *representation* layer



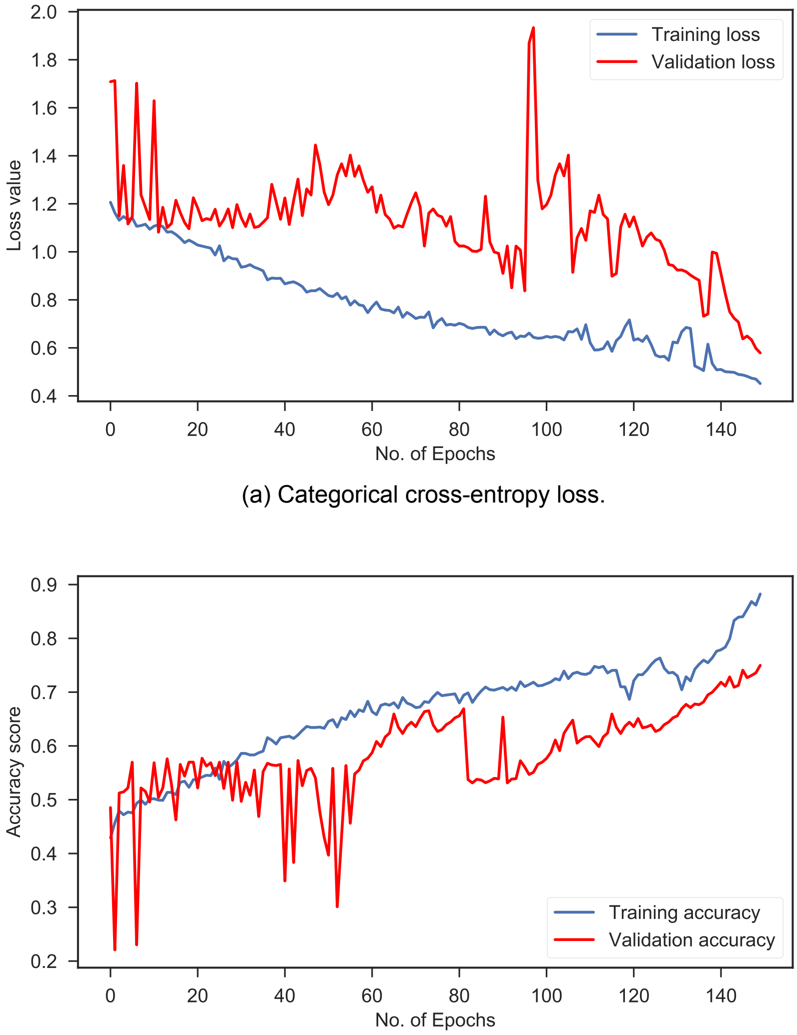
(a) Categorical cross-entropy loss

(b) Categorical accuracy

Figure 3.9: Categorical cross-entropy loss and accuracy performance during pre-training.

with the pre-trained weights learned using the BCG data. As the initial CNN layers learn the general features of the signal, these features are assumed to be common to all the heart-based signal types. Hence, the first six blocks of the 8-blocked *representation* layer are ”frozen”, i.e they are made untrainable and the last two blocks are used for re-adjusting the weights according to the input signal type, as can be seen in Fig. 3.8. This way the model uses the general heart features learnt using the BCG signal but re-learns the more specialized features that could be specific to the ECG or the PPG signals.

As the *Fitbit-PPG* dataset was highly obfuscated and aggregated, we could not employ the same implementation process as mentioned above. For this dataset, we first filled the missing samples by computing the mean of the immediate 12 samples around the missing value as recommended in [66]. Furthermore, we randomly selected sequences of peaks in the PPG signal and duplicated them at locations where the peak data was completely missing. Lastly, instead of loading the weights of all the 8-blocks of the *representation* layer, we used only the first four blocks and froze the top two of them. This training decision was taken because the PPG dataset had lesser number of samples (owing to its lower sampling rate) and it was highly aggregated. The depth of 8 blocks of CNNs quickly diminished the signal length since we fed 30 seconds sequences as the input data.



(a) Categorical cross-entropy loss.

(b) Categorical accuracy.

Figure 3.10: Categorical cross-entropy loss and accuracy performance during fine-tuning.

3.3.6. Baseline models

To compare the *DeepSleep* model's classification capability it was necessary that we develop a baseline model specific to our BCG dataset. However, as seen in Section 2.3.1, 2.3.2 and 2.3.3, there exist few studies that have used the BCG signal type for performing multi-class sleep stage classfication using either the traditional machine learning methods or the deep learning methods. Hence, for ensuring fair comparison and understanding our model's capabilities, we reproduced some of the works that have yielded the most promising results in this domain.

For this work, we reproduced two categories of models: the traditional machine learning model and the deep learning models. For the traditional machine learning models we reproduced the works of Fonseca et al. [39] and Kortelainen et al. [73]. Although Samy et al.'s [111] work involved in classifying 4 sleep stages from the BCG signal source, the nature of the sensor values were quite different than those obtained in our work. Samy et al. relied on extracting features from the heat map formed by the sensor values as their sensor sheet was a large matrix of sensors. As this was not possible with our nature of signal values, Samy et al.'s model was not used to recreate a baseline model.

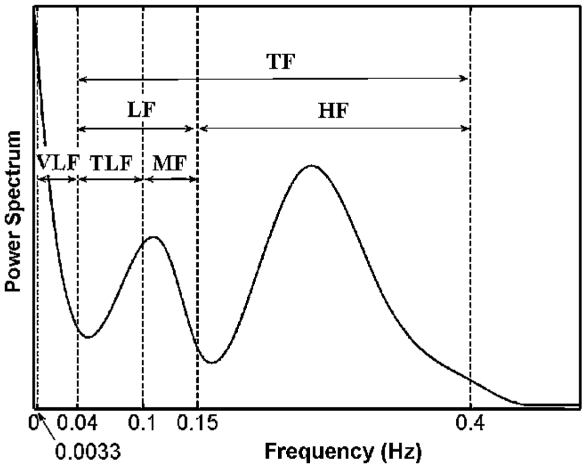


Figure 3.11: Frequency distribution of a typical heart signal.

As specified in Fonseca et al.'s study [39], over 42 HRV and respiratory features were extracted from our BCG datasets. The standard Pan-Tompkins [98] method was used for heart-beat detection. An open-source beat detection tool was used for extracting heart peaks [52]. The values between each consecutive beat (or peaks) were used to extract the HRV features extracted in Fonseca et al.'s work [39]. Time-domain, frequency domain and statistical features pertaining to the HRV analysis were extracted for this work. The features used are described as follows:

1. **Time-domain and Statistical features**:
   Features such as mean RR interval (mRR), mean heart rate (HR), standard deviation of normal RR intervals (SDNN), coefficient of variation, root mean squared differences of RR intervals (RMSSD), pNN50 and mode of RR interval values were extracted to represent the time-domain and statistical domain of the HRV. The description and formula for these features are given in Table 2.1.

2. **Frequency domain features**:
   Frequency-domain features mainly pertains to the frequency analysis of the heart signal in three different frequency bands: VLF (0-0.04 Hz), LF (0.04-0.15 Hz) and HF (0.15-0.4 Hz). Fig. 3.11 shows the spectral characteristic of the heart signal in the frequency domain.

   (a) **Spectrum Power**
   As the total spectrum power fluctuates over time and is different for each subject, a relative spectrum power is used instead of its absolute value. LF/HF, MF/LF, TLF/LF: the ratio of power within different frequency bands are computed. Hence, in total, 5 features are extracted corresponding to the power in each of the frequency bands.

   (b) **Mean frequency**
   For a band of power spectrum with energies of $P_1, P_2, ..., P_N$ at frequencies $f_1, f_2, ... f_N$, the mean frequency is defined as

$$f = \frac{\sum_{i=1}^{N} f_i P_i}{\sum_{i=1}^{N} P_i} \tag{3.3}$$

   (c) **Spectral entropy**
   Spectral entropy (SE) characterizes the complexity of a series in frequency domain. SE is defined as

$$SE = -\sum_{i=1}^{N} \frac{p_i \ln(p_i)}{\ln(N)} \tag{3.4}$$

where p_i denotes the proportion of energy, E_i, in the whole energy band.

(d) **Peak in HF spectral band**
The peak in HF spectral band is generally considered to provide information related to the respiratory modulation in the heart signal. The corresponding frequency approximately provides the respiratory rate. Normalized amplitude, $HFamp$ is calculated using this information.

$$HFamp = \frac{\text{amplitude of the peak}}{\text{total power in HF band}} \tag{3.5}$$

As stated by Fonseca et al., a cubic spline-based smoothing function is applied to post-process these features and to represent them in a sequential manner. The cubic spline method is given by the following relation:

$$v_i = h(t_i) + \epsilon_i \tag{3.6}$$

where, $i = 1, 2, ..., n)$ and $t = t_1, t_2, ..., t_n$ and $v = v_1, v_2, ... v_n$ indicates the time indices spaced at 30s and their corresponding feature values respectively. The smoothing function (h) is given by:

$$h = argmin_h \left[\sum_{i=1}^{n} [v_i - h(t_i)]^2 + \lambda \int_{t_1}^{t_n} h''(t)^2 dt \right] \tag{3.7}$$

where, λ is a smoothing parameter. Finally, A multi-class non-linear discriminant (LDA) was used as a classifier. To reproduce Kortelainen et al.'s model [73], time-variant autoregression model (TVAM) was used to consider the non-stationarity of the peak-intervals. Later, five heart-based spectral features were extracted from the parameters of the autoregression model. As described in the work, an HMM was used as the classifier.

Lastly, to form baseline models to compare the capability of deep neural networks, all the works described in Section 2.3.3 are reproduced. Since all the studies, except for Supratak et al.'s work, do not provide enough information about the parameters used in their work, we used default parameters as set by the Keras framework [24].

4
Results and Analysis

In this chapter we analyse our model's performance and present the results obtained on the test set. For a fair evaluation of our model's performance we compare it against the ground truth data obtained from the PSG studies. We also compare our model's classification scores against the most relevant prior works. As there are not many significant prior works which have used the BCG for classifying sleep stages, we attempted to reproduce some of the deep neural network-based prior works. This gives a perspective on how the design choices and the nature of the signal has contributed to the performance of our model.

This work uses the *Dozee* dataset described in Section 3.1.1 to perform the primary training of the model (pre-training and fine-tuning). A train-test split of 80-20% was used in this work for training and testing the model. Given that the *Dozee BCG* contains 51 recordings in total, 41 recordings were used for training the model. The model and its parameters were validated using 7 recordings and the remaining 3 held out data were used to test the model's predictions. Later, the pre-trained model was used for performing classification using the ECG and the PPG dataset (transfer learning setting). The evaluation metrics used in this work are the standard metrics like *precision*, *recall*, *f1-score*, and *categorical cross-entropy loss & accuracy*, that are commonly used for multi-class classification and used in prior works. We primarily report our findings in terms of *precision* and *recall* so as to give a better insight into how our model performed for each of the individual class.

4.1. Classification results

Fine-tuning of the model is performed on a calidation data consisting of 7 nights of recording. Cross-validation technique was not used in this work as it would result in a longer training time for each of the cross-validation runs. However, to fairly report the performance of the model, we tested the model on the test data consisting of 19 nights of recording (3, from *Dozee BCG* and 16, obtained for sleep quality measurement). The accuracy and the F1-score reported in this section are the mean test score. In this section, we present the performance of our model in terms of *precision*, *recall* and *f1-score* when tested on the test recording.

DeepSleep's performance

Table 4.1 shows the precision, recall and f1-scores for each class predicted by our model. The precision of the model is least for *Wake* stage. But the comparable recall score for *Wake* shows that the classifier is able to learn or detect the occurrence of the *Wake* stage. It is not able to accurately label every actual occurrence of *Wake* in its prediction. The model is able to predict the period of wakefulness but it fails to accurately predict the onset or the end of the *Wake* period. *Light* and *Deep* stage have been predicted comparatively well than the other two stages. This can be attributed to their higher number of representations. Also, the heart-signal, by itself, is enough to identify *Light* and *Deep*, but same is not the case for *Wake* and *REM*. *Wake* and *REM* stage are associated with small movements which is not

Stages	precision	recall	f1-score	support
Deep	0.74	0.76	0.75	236
Light	0.74	0.75	0.74	497
REM	0.77	0.64	0.70	193
Wake	0.59	0.70	0.64	98
avg / total~	0.74	0.73	0.74	1024

Table 4.1: Precision, Recall and F1-score of the *DeepSleep* model

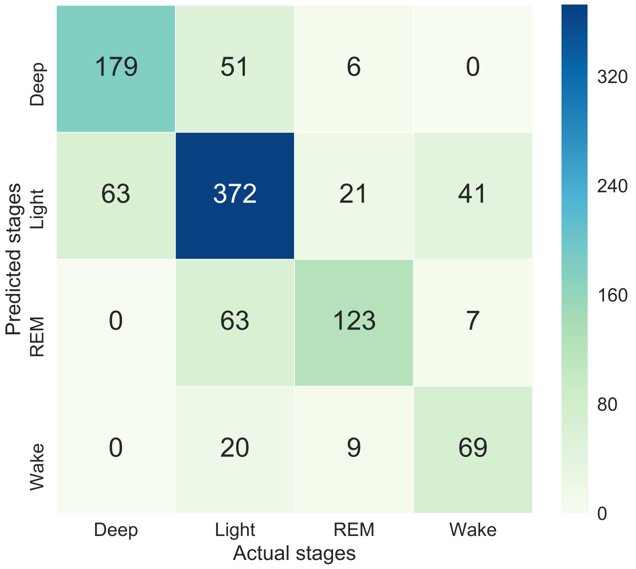


Figure 4.1: Confusion matrix for *DeepSleep*'s prediction.

picked up the model. It is possible that our model has considered the movement data as a noise in the heart signal and omitted it from the feature learning. If the movement signal is given as a separate input to the CNN layers, we can instruct the model to check for significant changes in the movement signal before making any prediction. This shows that our model may require more than one signal to perform better classification. Heart-signal in itself may not be enough to achieve higher accuracies or accuracies close to the clinical settings.

Fig 4.1 gives the confusion matrix report of the model predicted on the test data. The stages on the x-axis represent the actual or target stages and the ones on the y-axis are the predicted stages. The confusion matrix shows that the *Light* stage has been classified the most correctly followed by *Deep* and *REM*. *Wake* stage has been misclassified the most. It should be noted that none of the *REM* stages were detected as *Deep*. For a model to learn this kind of differentiation between two important stages shows the representative power of the model. The *REM* and the *Deep* stages are two completely opposite stages. This fact is largely supported by the model's predictions as well. Similarly, *Deep* was never predicted as *Wake* and vice-versa. This shows that the model's features were able to distinguish the *Deep* stage from the more movement-affected stages (*Wake* and *REM*). However, *Wake* has been misclassified as *Light* and *REM*. Similarly, around 56% of the *REM* instances were wrongly classified as *Light* stage. One reason can be the higher number of instances of *Light* sleep since there are few misclassifications for *Wake* and *REM*. However, this also shows that the model is not that capable to distinguish the movement-influenced stages. Without the knowledge about the movement signal, the model will not be able to learn these differences from the heart signal. This further reiterates the fact that multi-modality can further help improve the model.

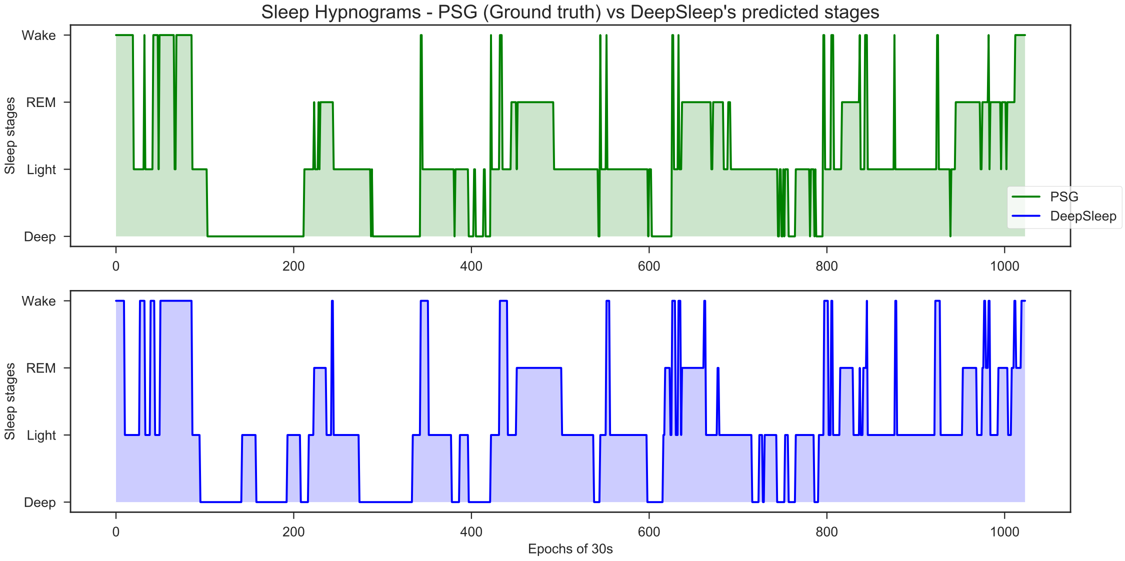


Figure 4.2: Sleep hypnogram - PSG (Ground truth) vs DeepSleep's predictions.

To get a clear picture of the *sequential learning* ability of our model, we plot the hypnograms (sleep patterns) predicted by our model and compare it against the PSG stages (ground truth). Figure 4.2 shows the sleep patterns according to the PSG data (ground truth data) and the patterns as classified by our model. From the figure it can be seen that our model identifies quite accurately the occurrences and transitions of the stages. Most importantly, the model seems to have learned the important rules of sleep stage transition wherein a *REM* stage never follows a *Deep* stage (and vice-versa). The *Light* stage always follows the *REM* or *Deep* stage, hence proving itself to be a transitional stage. Similarly, it learns the biological transition rule for the *Wake* stage as well. Another important rule that the model learns is the amount of time spent in *Deep* and *REM* stage. For any normal sleep, *Deep* stage starts higher during the beginning of the night and gradually decreases. At the same time, the *REM* stage gradually increases. This behaviour is captured by our model as can be seen from the Fig. 4.2. Interestingly, our model choses to predict smoother transitions of sleep stages unlike the stages scored by the experts. It can be seen around epoch 300 and 420 where the experts' scoring makes a sudden transition from *Wake* to *Light*, our model chooses to make a smoother shift to the *Light* stage. Lastly, Fig. 4.3 shows our model learning the occurrences of sleep cycles, which is an important factor in determining the sleep quality.

Fig. 4.4 highlights the areas that have been wrongly predicted by the DeepSleep model. We can see that most of the wrong predictions have been because of the overestimation of the *Light* stages. This is supported by the confusion matrix and the f1-score. As explained earlier, the model rightly identifies the period of most of the sleep stages. The sequence is almost always predicted. However, it fails to predict the exact onset time of a sleep stage. Other misclassifications seem to arise due to the model's nature of predicting a smoother transition. Around the 100th epoch, the ground truth hypnogram shows that the user's sleep suddenly shifted from *Wake* to *Light* and then back to *Wake*. Our model does identify this behaviour but later chooses to stick with a continuous waveform of *Wake* sleep. This behaviour is good if the model is used for general sleep tracking as users prefer to view a continuous meaningful sleep pattern. However, this level of precision would be necessary if the model needs to perform comparably with the ground truth standards. Interestingly, this can also suggest that the model avoids the human disagreement that arises when multiple experts are scoring the stages. In our work, the agreement score between the two experts is

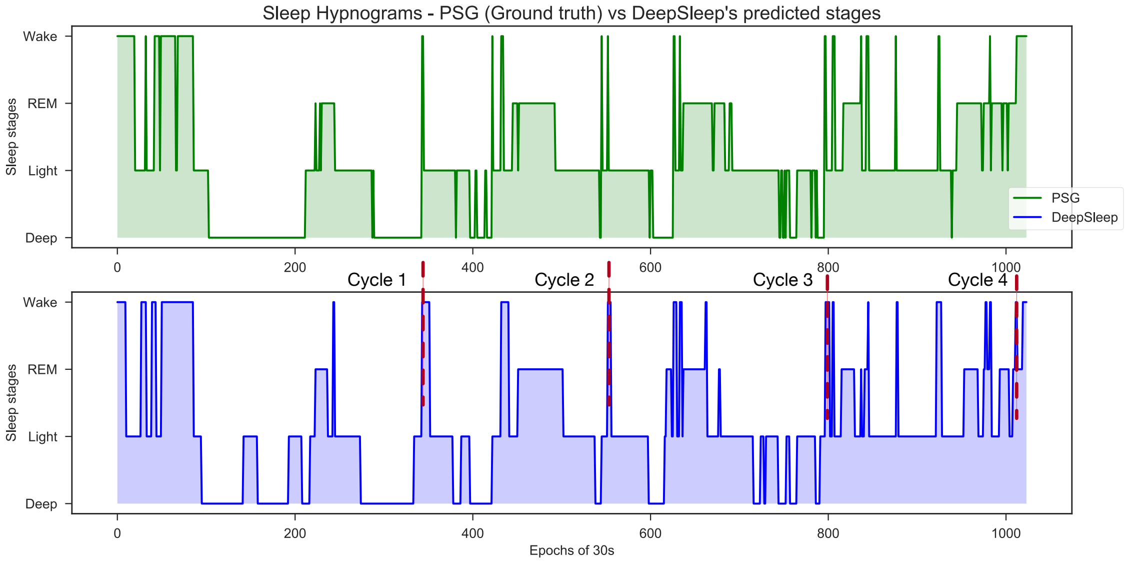


Figure 4.3: Detection of sleep cycles by the DeepSleep model.

Study	Sensor type	#Features	Classifier	Classes	Accuracy
DeepSleep (proposed)	**BCG**	**1**	1D-CNN + bi-LSTM	**REM, Deep (NREM)**	**82.43%**
Devot et al. [32], 2007	Textile-ECG	3	Hidden Markov Model (HMM)	REM, NREM	81.5%
Kortelainen et al. [73], 2010	BCG	5	HMM and Autoregression	REM, NREM	74%
Renevey et al. [108], 2017	PPG, 3D accelerometer	5	HMM	REM, NREM	81.35%

Table 4.2: Performance comparison between *DeepSleep* model and related work on 2-class classification - REM and Deep (NREM)

80% ($\kappa = 0.8$). However, we cannot prove this hypothesis conclusively since we do not have the individual scorings of the experts.

Comparison of binary classification

In Table 4.2, we compare our model with the prior works on BCG and other non-invasive sensors that have performed binary classification. Although our proposed model seems to have comparable scores with other studies, the number of features considered for classification in this work is lesser than those of the related works. Devot et al. [32], Kortelainen et al. [73], and Renevey et al. [108] performed binary classification using more than 3 features. Considering that, we can say that our model is slightly better than most of the works given in Table 4.2.

Comparison of multi-class predictions

Table 4.3 compares our *DeepSleep* model with prior works that have performed multi-class classification. It can be seen that our model has performed better than the *DBN* model used by Langkvist et al. [79]. It should be noted that the other works in Table 4.3 use EEG, EOG, EMG signals for sleep staging. These signals are relatively easy to perform sleep classification since they are visually distinguishable. Although our model performs better than Samy et al. and Längkvist et al., we are not able to fairly compare its performance against other deep neural net-based works as they use the standard PSG signal as input.

Hence, we attempted to reproduce some of the prior works' model architectures and applied them on our *Dozee BCG* dataset. Table 4.4 compares our *DeepSleep* model with other models when applied on our BCG dataset. It can be seen from Table 4.4 that our *DeepSleep*

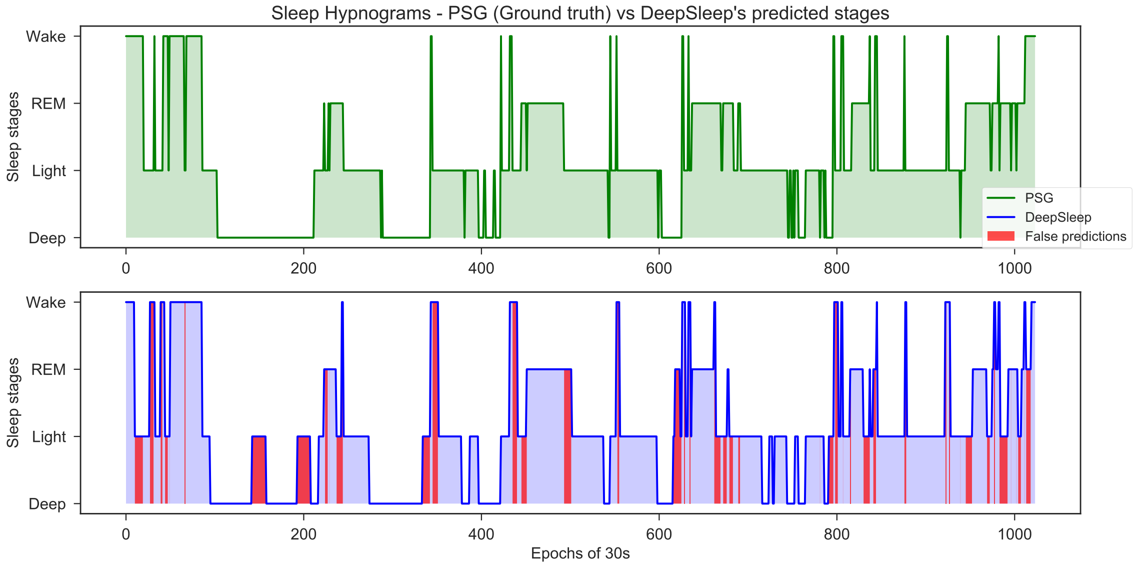


Figure 4.4: Comparing DeepSleep's true and false predictions.

Study	Sensor type	#Features	Classifier	Classes	Accuracy
DeepSleep (proposed)	**BCG**	**1**	**1D-CNN + bi-LSTM**	**W, L, REM, Deep (NREM)**	**74%**
Samy et al. [111], 2014	BCG	6	KNN, SVM, Naive-Bayes	W, L, REM, Deep (NREM)	72%
Längkvist et al. [79], 2012	EEG, EOG, EMG	1	DBN, HMM	W, REM, NREM, L	72%
Dong et al. [34], 2018	EEG, EOG	1	LSTM	W, REM, NREM, L	86%
Supratak et al. [121], 2017	EEG	1	1D-CNN + LSTM	W, REM, NREM, L	86%
Chambon et al. [21], 2018	EEG, EOG, EMG	1	1D-CNN	W, REN, NREM, L	87%

Table 4.3: Performance comparison between *DeepSleep* model and prior works that perform 4-class classification.

model performs better than the other studies. With respect to the classical machine learning-based baselines, our model performs significantly better due to its robustness to noise and its self-feature learning capabilities. The classical baseline models required hand-crafted HRV features to perform classification. However, these features rely on the accurate identification of heart peaks. Since the peak detection algorithm used in this work was mainly optimized for ECG signals, it fails to identify peaks in our *Dozee BCG* data. This could have affected the quality of the extracted features. Secondly, LDA and HMM were used as classifiers. These classifiers lack the sequential prediction capability unlike RNN-based classifiers. In terms of the deep neural net-based models, we argue that the model's architecture and the design choices have played an important role in obtaining a performance score. Dong et al. [34] and Chambon et al. [21] have designed their model to work with the EEG signals. Their architecture captures the brain activity's frequency range of 8-24 Hz and treats rest of the zones as sparse data. Hence, multilayer perceptron is used by Dong et al. to remove this sparseness and a different filter size is set by Chambon et al. to reflect these properties. BCG signal does not have such kind of discrete activity ranges. Thus when these models are applied on our BCG dataset, either most of the frequency ranges were cleared or not represented well. Supratak et al. [121] uses two branches of CNN to extract time-domain and frequency-domain features. This design seems to apply for our dataset as well but after a certain depth in the CNN layers, the CNN model begins to extract same and duplicate features. This is again attributed to the fact that heart activity does not have a discrete frequency spectrum unlike brain activity. Längkvist et al.'s [79] model is based on the Deep belief networks (DBNs). Although these models can learn from the raw signal waveforms, they seem to lose

Study	Sensor type	#Features	Classifier	Classes	Accuracy
Fonseca et al. [39], 2015	BCG (ECG, RIP)	42	LDA	W, REM, Deep, L	63% (69%)
Kortelainen et al. [73], 2010	BCG (BCG)	5	HMM, Autoregression	W, REM, Deep, L	64%
Kortelainen et al. [73], 2010	BCG (BCG)	5	HMM, Autoregression	REM, Deep (NREM)	70% (74%)
Längkvist et al. [79], 2012	BCG (EEG, EOG, EMG)	1	DBN, HMM	W, REM, Deep, L	68% (72%)
Dong et al. [34], 2018	BCG (EEG, EOG)	1	LSTM	W, REM, Deep, L	69% (86%)
Supratak et al. [121], 2017	BCG (EEG)	1	1D-CNN + LSTM	W, REM, Deep, L	70% (86%)
Chambon et al. [21], 2018	BCG (EEG, EOG, EMG)	1	1D-CNN	W, REM, Deep, L	71% (87%)
DeepSleep (proposed)	**BCG**	**1**	**1D-CNN + bi-LSTM**	**W, REM, Deep, L**	**74%**

Table 4.4: Comparison of different model architectures on the *Dozee BCG* dataset. These models have been reproduced in this work to make this comparison. The original sensor types used in these models are mentioned in the brackets. The accuracy scores stated in brackets for the baseline models denote the accuracies as reported in their original works.

their descriptive power when there is a big sequential input.

The confusion matrix in Figure 4.6 shows the misclassifications by the baseline model (Fonseca et al.'s model). Unlike *DeepSleep*'s predictions, the baseline model misclassifies *Deep* as *REM* and *Wake* and vice-versa. Even with 42 features, the model is not capable to distinguish between the two largely different stages – *Deep* and *REM*. However, Table 4.5 shows that the precision for *Light* and *Deep* stages is significantly high. It suggests that the HRV features extracted for this model have a discriminative ability towards *Deep* and *Light* sleep. Interestingly, from Fig. 4.5, we can see that the baseline model's predictions do not follow the sequential nature of sleep stages. This could be explained by the linear classifier used for the classification which does not take the temporal order into account. Because of this, the baseline model does not learn the rules of sleep transitions. This shows that although the extracted features could model the individual stages well, they could not represent the transitional and sequential property of these stages. The baseline model directly predicts

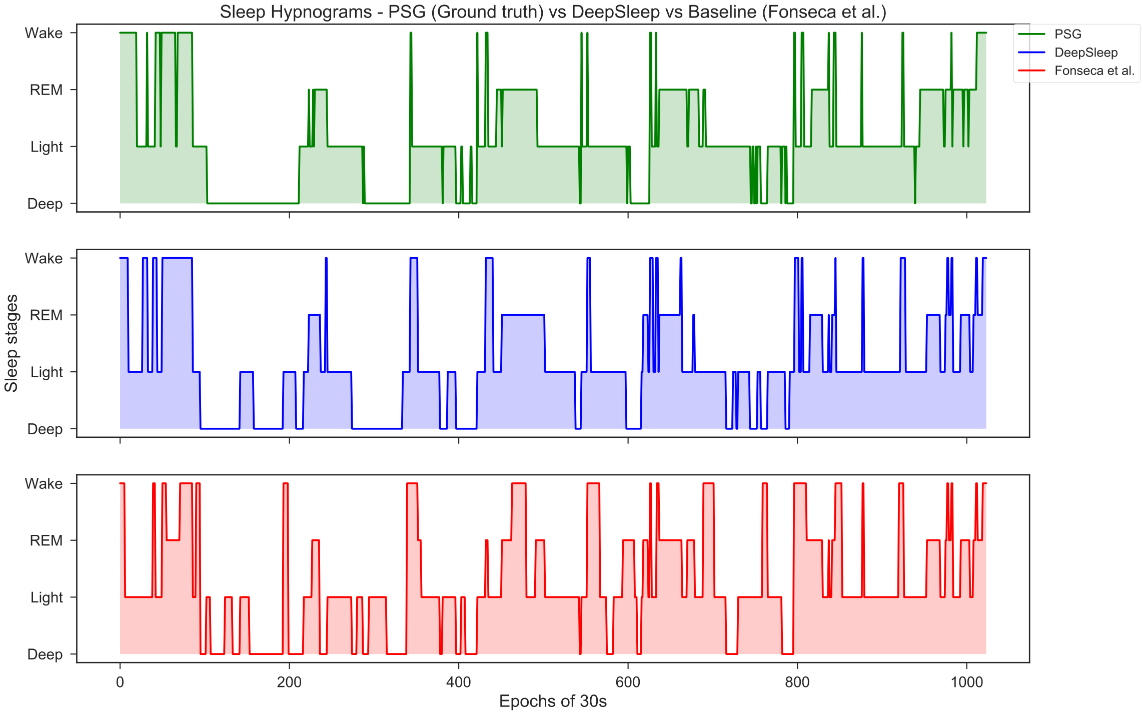


Figure 4.5: Sleep hypnogram - PSG (Ground truth) vs DeepSleep vs Baseline (Fonseca et al. [39]) predictions.

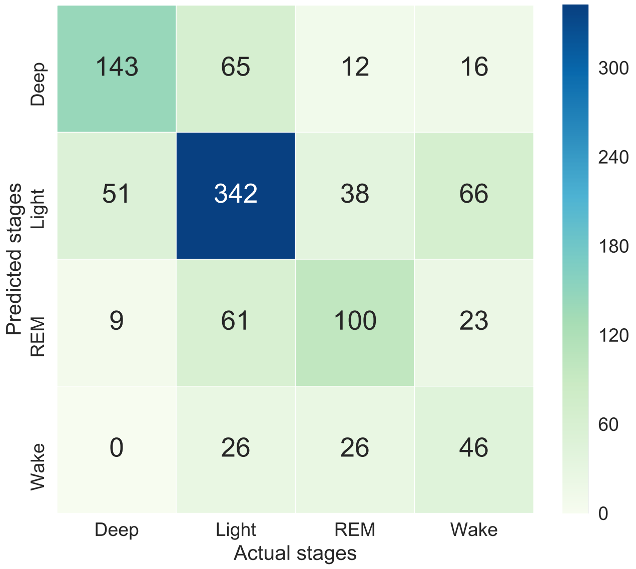


Figure 4.6: Confusion matrix depicting the classifications and misclassifications of the Baseline (Fonseca et al. [39]) model.

Wake immediately after *Deep* or *REM* (and, vice-versa) in many instances. This behaviour is not useful for sleep tracking since it does not provide meaningful information. Also, by predicting such erroneous transitions, it could mislead user into believing that they may be suffering from disorders like insomnia (in which such kind of patterns can be seen [109]).

Having said that, the baseline model also seems to agree with DeepSleep model's in producing smoother transitions (around epoch 380). The hand-engineered features seem to model the stages well, given the noisy nature of the BCG signal. The features for *Deep* and *Light* sleep could be used as an external knowledge injection to the *DeepSleep* to further improve the model's classification abilities.

Therefore, with the help of above arguments and our model's performances against prior works, we believe that our model's architecture is specifically suited for heart signals. Moreover, the representative capability of our model suggests that our model architecture could be used for other heart-based problems such as irregular heart beat detection, apnea detection, and heart-based biometrics.

4.2. Transfer learning

Table 4.6 shows the overall performance of the *DeepSleep* model on the different datasets. As described in Section 3.3.2, the *Dozee BCG* was used to pre-train the model. The other three datasets were trained and tested using the pre-trained model. *Dozee ECG*, *MIT-BIH ECG* and

	precision	**recall**	**f1-score**	**support**
Deep	0.70	0.61	0.65	236
Light	0.69	0.69	0.69	497
REM	0.57	0.52	0.54	193
Wake	0.30	0.47	0.37	98
avg / total	0.63	0.62	0.62	1024

Table 4.5: Precision, Recall and F1-score of the baseline (Fonseca et al. [39]) model.

Dataset	Sensor type	#Features	#Recordings	Classifier	Classes	Accuracy
Dozee BCG	BCG	1	51	1D-CNN + bi-LSTM	W, L, REM, Deep (NREM)	74%
Dozee ECG	ECG	1	51	1D-CNN + bi-LSTM	W, L, REM, Deep (NREM)	77%
MIT-BIH ECG	ECG	1	80	1D-CNN + bi-LSTM	W, L, REM, Deep (NREM)	82%
Fitbit-PPG	PPG	1	12	1D-CNN + bi-LSTM	W, L, REM, Deep (NREM)	63%

Table 4.6: Performance of DeepSleep model on different datasets and sensor types

Dataset	Sensor type	#Features	#Recordings	Classifier	Classes	Accuracy
MIT-BIH	ECG	1	80	1D-CNN + bi-LSTM	W, L, REM, Deep (NREM)	86%
Dozee BCG	BCG	1	51	1D-CNN + bi-LSTM	W, L, REM, Deep (NREM)	80%
Dozee ECG	ECG	1	51	1D-CNN + bi-LSTM	W, L, REM, Deep (NREM)	84%
Fitbit-PPG	PPG	1	12	1D-CNN + bi-LSTM	W, L, REM, Deep (NREM)	67%

Table 4.7: *DeepSleep* model's performance on different datasets when pre-trained using *MIT-BIH ECG*.

Fitbit-PPG were used to test the transfer learning setting of this model. It can be seen that the *DeepSleep* model performs the best on the *MIT-BIH* dataset. This can be attributed to the fact that around 80 full-night recordings containing high resolution of the ECG data is used for inference testing. The low signal-to-noise ratio of the ECG signal has helped the model to perform better. This is evident from the slightly better scores using the *Dozee ECG* data. Also, it should be noted that both the *Dozee ECG* and the *MIT-BIH* contain almost equal distribution of *REM*, *Deep* and *Wake* classes unlike the *Dozee BCG* data. Hence, there is a lower effect of class imbalance when using these datasets. However, the performance of the *DeepSleep* model is significantly low when tested on the *Fitbit-PPG* data. This could be attributed to the fact that the resolution of the PPG signal was too low for the proposed model. The sampling rate of the PPG data varied from 100-120 Hz whereas, the pre-trained model had weights learned using an input signal that was sampled at 250 Hz. One way to address this issue would be to oversample the PPG dataset, which we did in this work. Although we could handle the resolution by oversampling, it did not provide any new signal information. Due to this, the model was able to train and predict using this dataset but its representative ability was lesser than the other signals. Possible ways to address this issue could be to use generative networks to generate new samples from the source signal instead of simply duplicating them. Alternatively, a different classification model could be designed which has lesser depth and is based on the aggregated features rather than the raw signal waveform.

According to Torrey and Shavlik, transfer learning could also be used to offset the problems posed by limited or unlabelled data [127]. To test this hypothesis, we have also pre-trained our model using the *MIT-BIH*'s ECG data as its size is significantly larger than the other datasets. Later, we tested if our model, pre-trained using the *MIT-BIH ECG*, is able to classify stages using the *Dozee BCG* data. Table 4.7 shows the performance of the *DeepSleep* model on the *Dozee BCG*, *Dozee ECG* and the *Fitbit-PPG* dataset, when pre-trained using the *MIT-BIT ECG* dataset.

4.3. Sleep quality correlation measurement

Table 4.8 shows the perceived sleep quality scores reported by the users, compared against the objective scores calculated by the PSG study and by our model. The formula used by the sleep experts to calculate the sleep quality is:

$$SQ = \frac{REM(min) + NREM(min) - Awakening(min)}{\text{Total sleep duration (min)}} \tag{4.1}$$

We used the Pearson product-moment correlation test to compute the correlation coeffi-

Subject ID	Duration	1st Record Date	SATED score 1	2nd Record Date	SATED score 2	Mean SATED score	NIMHANS score	DeepSleep score
Subject_06_SA	7h 30min	6/8/2018	82	7/8/2018	84	83	86	80
Subject_07_SA	8h	7/8/2018	76	8/8/2018	77	76.5	79	72
Subject_10_SA	6h 14min	10/8/2018	74	11/8/2018	78	76	80	84
Subject_12_SA	6h 30min	12/8/2018	87	13/8/2018	84	85.5	83	80
Subject_14_SA	7h 10min	14/8/2018	86	14/8/2018	83	84.5	84	76
Subject_15_SA	6h 5min	15/8/2018	84	16/8/2018	85	84.5	80	79
Subject_16_SA	6h 50min	16/8/2018	82	17/8/2018	84	83	80	78
Subject_17_SA	6h 10min	17/8/2018	77	18/8/2018	82	79.5	82	84
Subject_20_SA	5h 55min	20/8/2018	64	21/8/2018	71	67.5	70	80
Subject_21_SA	6h 42min	21/8/2018	70	22/8/2018	73	71.5	70	75
Subject_22_SA	7h 24min	22/8/2018	92	23/8/2018	90	91	88	90
Subject_24_SA	7h 5min	24/8/2018	81	25/8/2018	81	81	83	88
Subject_25_SA	7h 10min	25/8/2018	79	26/8/2018	79	79	75	84
Subject_26_SA	6h 45min	26/8/2018	90	27/8/2018	92	91	92	84
Subject_30_SA	7h 5min	30/8/2018	88	31/8/2018	90	89	92	86

Table 4.8: SATED scores: Perceived sleep quality scores reported by users compared against the scores calculated by the PSG and the *DeepSleep* model

cient between the mean SATED scores and the scores calculated by the *DeepSleep* model. With a coefficient of $r = 0.43$, it suggests that there is a positive correlation between the perceived scores and the objective scores calculated by our model. However, the objective scores of our model has a strong positive correlation with the objective scores of PSG ($r = 0.48$), suggesting that the objective scores underestimate or do not account for some of the perceived factors.

Initially, for the user study, we only asked the users to fill in the questionnaire few minutes after waking up. We believed that the user would be able to score his sleep quality, w.r.t to the SATED dimensions, if they have fresh memory of their sleep. Although it was true, by following this method, we could not get a good indication of *alertness* and *satisfaction*. Almost every user reported having a maximal score of 10 for alertness immediately after waking up. Hence, we decided to collect another set of scores from the users after a day of their initial recording. This way we had two sets of subjective scores - one immediately after waking up and the other after a day of recording. This way we could see some changes in their perceived scores for alertness and overall satisfaction. We noticed that some of the perceived scores for alertness and satisfaction varied a lot from the original score. Feedback from some of the users supported this fact as they felt that by the end of the day they had a better understanding of their previous night's overall sleep quality.

Although this user study helped us understand the factors affecting perceived sleep quality, it had some limitations. Only two of the five dimensions in the SATED framework actually questioned the subjective quality of sleep. Most of the users reported high and similar scores for duration, efficiency and timing. This, coupled with the limited size of the user study, did not provide us with new or varied information. Secondly, the user study was conducted for only a single night. It is difficult to encapsulate the overall subjective sleep quality with a single night of PSG study. However, the need for ground truth data to base our predictions upon limits our study to just one night. Moreover, since the subjects of this user study were located in India, it was difficult to get their timely feedback and subjective answers related to the study. Lastly, we used simple objective sleep metrics like time spent in *REM*, *Deep* and *Wake* to calculate our objective scores. Factors like time to sleep, circadian rhythm, time taken to sleep, habitual waking time and mean of weekly sleeping times have been used in objective sleep quality measurement by some of the studies and sleep clinics. These factors have been shown to capture some of the dimensions in perceived sleep measurement [76, 129]. However, we use the simple formulation of objective sleep quality measurement as it is widely used by sleep clinics and also due to the limitation of our user study. Given the small size of our user study, we could not experiment with different weights for each of the sleep stages as recommended by Ohayon et al. [96].

5 Discussion and Future work

Driven by the motivation to provide a non-intrusive way of monitoring and quantifying sleep, this thesis aimed to enable long-term sleep monitoring by improving the sleep classification system using non-obtrusive wearable sensors [51, 62, 77]. In this thesis, we aimed to address the following research question:

> *How can a sleep classification system be modelled, using BCG sensor data, in order to achieve a performance comparable to medical standards?*

In this chapter, we revisit the research questions and discuss how our model's architecture and our training strategies achieve the research goal. In addition, we provide future research directions by identifying remaining open problems and improvements.

5.1. Discussion

In this section, we discuss our approach towards designing the classification model and how it helped in addressing the research questions of this thesis. We further discuss how our model is robust towards noise and the nature of the heart signals. We discuss about the generalizable capability of our model in learning features from one type of source signal and apply it on other signal types. We discuss how the model's sequential learning ability helped us in obtaining a positive correlation with the subjective sleep quality scores. Finally, we point out the limitations of our work and the challenges that this classification system would face while deploying it into a production environment.

In this thesis we designed a model architecture which is different from most of the related works. The proposed model comprised of a *representation* and a *sequential* layer. Additionally, the model was first *pretrained* and then *finetuned* to obtain better results than the random initialization of weight matrix. The main focus of this work was to identify sleep stages using the BCG-based heart signal. Our proposed model was able to identify sleep patterns from the BCG signal. By using 2-phase training technique, the model's classification performance was improved even with the limited amount of data. The first research sub-question, introduced in section 1.3, laid focus on building a model robust to noise. By using multiple stacks of CNN blocks in representation layer, our model was able to detect sleep patterns from the noisy source of heart signal. BCG signals are characterised by the presence of multiple J-peaks (explained in section 2.1.3) due to noise artefacts introduced by breathing and movement. Even after being heavily influenced by external artefacts, the *DeepSleep* model achieved an F1-score of 74%. Having said that, the model performs better on the ECG data which could be owed to the fact that the ECG signal has a significantly lower signal-to-noise ratio than a BCG signal. This shows that although our model is designed to be robust to noise, it is not able to completely differentiate between the normal samples and the noise artefacts. This could be addressed by adding more number of samples to the problem space and incorporating a noise-detection or anomaly detection model. By doing this,

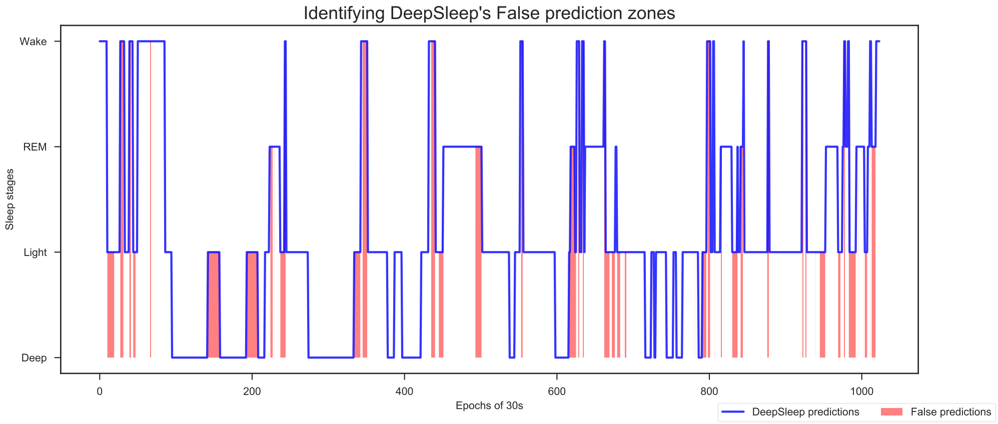


Figure 5.1: DeepSleep's true and false prediction zones. The zones highlighted in red indicate the falsely predicted sleep stage sequence.

we can increase the variance between the normal and the noisy signals. Eventually, usage of large enough number of data points would also mean that the *pre-training* phase could then be essentially bypassed.

The design choices of the model focuses on making it learn and predict from sequences of heart signal to take the temporal nature into consideration. The *sequential* layer in our proposed model handles this aspect of the problem. The *bi-directional LSTM* first learns the sequences in the forward run and then approaches them in a backward direction. This way the *bi-LSTM* learns the sleep patterns more efficiently. In addition to that, we implemented a novel way to reset the states of the *bi-LSTMs* whenever it gets an input from a different recording. By resetting states, we ensure that the *bi-LSTMs* treat the input signal as a signal consisting of 41 different recordings and not as one long signal. This way the *sequential* layer learns sleep patterns unique to each recording, just like the way a sleep expert would do. Additionally, Figure 5.1 shows that the disagreement between the true and false predictions is usually confined to the sequence of the transitional stages, like *Light* to *Deep* or *Light* to *REM*. This disagreement gives a sudden or erratic transition of sleep stages in an otherwise smooth sleep pattern. Interestingly, the *DeepSleep* model has supported some of these confusions in the sleep pattern and in some cases, it has opted to stick with a smoother transitions instead of erratic ones. This observation is quite interesting since it gives an indication that the model is, sometimes, better at making decisions about the stage classification than the human scorers, suggesting a reduced scoring bias introduced by the human scorers. This gives an overall view about how well the model was able to generalize the sleep sequence, even if the complete sequence was not predicted accurately. This shows that our model is *generalizable* and this property could be helpful in sleep-monitoring applications, if not for clinical applications.

Our *DeepSleep* model's comparison with other deep neural net based models suggest that although the task of sleep classification is common for all of these models, the architecture and the design choices play an important role for different signal source. Deep neural network-based studies described in Section 2.3.3 predominantly use the EEG signal type. This signal is sometimes supported by EMG and EOG. Hence the model architecture reflect the type of signals used for training. For instance, Supratak et al. [121] designed a 2-branched CNN model for feature extraction followed by an LSTM for sequential learning. The 2-branched CNN is highly reflective of the nature of the EEG signal since the EEG signal is predominantly active in a specified frequency range of 8-20 Hz. This frequency range is captured by the standalone frequency (or the spectral branch) modelled in Supratak et al.'s work. This nature of the EEG is similarly captured in Dong et al.'s work as well [34]. The multilayer perceptrons preceding the LSTM layer ensure that the sparse signal values lying outside the active EEG frequency range is removed. Hence it is not surprising when these models perform quite well on the standard PSG signals but yield lower accuracy scores when applied on the ECG and the BCG signal. In this matter, our *DeepSleep* model's architecture could be used as the starting point for training or designing other heart-related tasks.

Furthermore, the performance comparison of the reproduced traditional machine learning-based baseline models with our *DeepSleep* model suggests some significant advantages of using deep neural networks. While reproducing Kortelainen et al.'s and Fonseca et al.'s models, we applied the standard beat detection algorithm, the Pan-Tompkins algorithm [98]. The Pan-Tompkins algorithms works better with the ECG signal due to its low signal-to-noise ratio. The peaks are clearly distinguishable. However, when applied on the BCG data, many of the peaks were left undetected. To obtain a better classification accuracy, we manually checked every labelled beat and corrected them visually. This difficulty was also noted by Kortelainen et al. [73] in their work owing to the contamination of the BCG signal with the movement and breathing artefacts. After the manual inspection of the peaks, over 142 and 5 features were extracted for Fonseca et al.'s and Kortelainen et al.'s model. As these features are extracted using the beat intervals, a wrongly detected beat introduced noisy features. This could be attributed to the fact that the reproduced works fared poorly (Fonseca et al.: 63% vs 69% originally; Kortelainen et al.: 70% vs 74% originally for 2-classes) on our *Dozee-BCG* dataset as compared to their original scores on the ECG dataset. Owing to the noisy nature of the BCG signal, Kortelainen et al. performed only 2-class classification in their work. Although the *DeepSleep* model requires way more training data and training time to achieve similar scores on the BCG data, it allows for automatic feature extraction and noise suppression without any requirement for manual inspection. In this aspect, the deep neural net-based models could be used as alternatives to the more traditional models to design an end-to-end classification system. Having said that, these deep neural net models lack the explainability factor which is highly important in the healthcare domain.

The second sub-question focusses on building a *transferable* sleep classification model. By using the *pretraining* and *finetuning* strategy to train the model, this work has enabled the model to be reused for different sources of heart signal. The *pretrain* layer of the model makes it possible to load the pretrained weights and apply the whole pipeline described in this thesis to a heart signal obtained from ECG or PPG. The model's performance on *Dozee ECG* (77%), *MIT-BIH* ECG (82%) and the *Fitbit-PPG* (63%) shows that *transfer learning* is possible with sleep scoring problem. The performance was better when applied on the ECG signal than on the PPG signal. This could be attributed to the low signal-to-noise ratio in the ECG when compared to the other non-invasive sensor signals. Moreover, the higher number of recordings for the ECG signals has a positive effect on the model's performance. It should also be noted that the transfer learning setting in our scenario was too restrictive in terms of signal parameters like sampling rate and number of samples. The model was primarily pre-trained on the BCG dataset which was sampled at 250 Hz. Since, the *Dozee ECG* and the *MIT-BIH* ECG were sampled at 250 Hz, the pre-trained model's weights were able to adapt and learn the features from the ECG signals with less preprocessing. However, the PPG signal was sampled at an average sampling rate of 120 Hz. Hence, it would not entirely correct to assume that the same pre-trained model would be able to learn features from the PPG signal. For such lower sampled signal, a smaller model could have generalized the features better than our default 16-layered CNN model.

Interestingly, the *DeepSleep* model performed better for the transfer learning setting when trained using the *MIT-BIH ECG* data. This result is in line with one of the application of transfer learning wherein a largely available dataset could be used to pre-train a model and be used for classification for an intrinsically different data type. This aspect has been widely employed in the fields of image recognition using ImageNet [58], NLP's language models [115, 138] and speech recognition models [3]. Given the encouraging results in this work, the highly and openly available ECG datasets could be used to train a robust neural net which can be used as a base model for other heart-based sensor types.

Our third sub-question focussed on testing if the objective sleep quality or sleep efficiency score correlates with the perceived sleep quality of the subjects. Our correlation test suggests that there exists a positive correlation between the perceived scores and the objective scores calculated by our model ($r = 0.43$). Additionally, the objective scores of our model has a strong positive correlation with the objective scores of PSG (r = 0.48). This suggests that the objectives scores of the PSG and DeepSleep match. However, the calculation of the objective scores takes only the measurable quantities like total time spent in REM and Deep sleep. Hence, the correlation factor is affected by the model's precision. The alertness, drowsiness and the satisfiability factors are not accounted as their assessment can be done only after the normal duration of sleep. Hence, these objective scores do not give much insight into the perceived scores even though they have a positive correlation. It can be further improved by accounting for short awakenings, ambience factors (like temperature, light and humidity) and sleeping habits like average time to sleep and the usual time of sleeping. Additionally, our user study was highly limiting in nature to make any conclusion regarding the correlation between objective and perceived sleep measurement. With only 16 subjects and 5 dimensions of subjective quality assessment, it is difficult to model user's perception of sleep quality. Other factors like time slept and time taken to sleep determine the circadian rhythm of a user. This factor is important as it suggests the usual sleeping habits of a user. For instance, a person who has been sleeping at 11 PM almost regularly and takes about 10 minutes to fall asleep could be said to have a better night of sleep than an individual who sleeps at the same time but takes around 20-30 minutes. As our SATED-based user study only considered data from one night of sleep, it was difficult to consider such sleep rhythm patterns into our objective scoring. Additionally, the user study could be improved by adding more dimensions to the assessment and by considering user's feedback alongwith their scores. Long-term quality assessment frameworks like PSQI can be used in conjunction with SATED questions to form a better understanding of the perceived sleep quality.

In addition to the above points, training a deep neural network from the ground-up is a highly computationally intensive task. The *pre-train* phase takes about 15 hours of training time when run on a single GPU. By using distributed GPU computing, we reduced the training time to around 9 hours. In addition to that, hyper-parameter tuning and searching takes around 8 hours of computational time. This inhibits conducting the experiments frequently with different architecture after every run. Hence, it is important to optimize the size, complexity and the number of parameters of the deep neural network. Although this has not been a main concern during this work, it is important to consider this factor for the model to be easily deployable on portable devices or to be production-ready. Having said that, to the best of our knowledge, no literature till date has provided a heart signal-based pre-trained model or even a methodology to build one. With this work, we provide a way to pre-train an end-to-end, sample-level model using BCG and ECG signals. We believe that this could pre-trained model could be used to further improve the state of the art in sleep monitoring tasks and also other heart-related problems like irregular heart-beat detection [104], classification of heartbeats [30] and heartrate-based biometrics [87, 97, 139]

5.2. Future Work

The presented classification model is, to the best of our knowledge, the first model to use deep learning algorithms to predict the sleep stages from a non-invasive BCG data. Most of the prior works have worked on either the EEG or the ECG signals. Although their performances were comparable to the clinical accuracy, their mode of data collection is invasive and not suitable for daily home monitoring of sleep. This thesis attempts to address this shortcoming by designing a model that can learn features and predict sleep stages from the non-invasive BCG signal and is robust to noise. Having said that, the performance obtained by our proposed model is in the right direction but not yet comparable to the clinical accuracy. In this section, we identify some of the gaps in this work and present some improvements that could definitely increase the performance of our model.

Oversampling technique

In this work, we adopted random oversampling as the sampling technique to perform pre-training. By oversampling the data, we present equal distributions of all the classes to the CNNs in the *representation* layer. We have shown how oversampling followed by pre-training improved the validation loss over normal pre-training (i.e pre-training on non-oversampled dataset). However, it should be noted that the random or SMOTE-based oversampling technique randomly selects samples of the minority class and creates synthetic samples from the original data. This way the temporal order of the heart signal gets disturbed. The disturbance of the temporal order could bias CNNs into learning incorrect features. One of the improvement in this direction could be to use a *Seq2Seq autoencoder* [122] to encode a signal length of 30 seconds and create a new synthetic sequence of the signal length. The generative property of the *Seq2Seq* network could retain a high amount of correlation and temporal order of the original sequence and still generate a new sequence. This method is used quite widely in the field of speech recognition where synthetic voice samples are generated to increase the training data. This way we could retain the temporal property of the signal and also generate higher amount of training data. Gong et al. [47] provide a novel way to oversample data for sequence classification by using a generative Recurrent Neural Network (RNN). This generative RNN forms a kernel to capture the similarity between the sequences and then generates new synthetic sequences.

Unsupervised and Semi-supervised learning

One of the biggest challenge in tackling a machine learning problem is obtaining high quality of annotated data. However, labelling data, especially sensor data like the BCG signal that are generated at a high sampling rate, is very expensive and complex. For sleep classification, annotated data can only be obtained by conducting regular sleep studies in collaboration with sleep doctors and hospitals. Having said that, the unlabelled BCG signal from the sensor sheet is easily available as the users can record the data from the comfort of their home. If these high volumes of unlabelled data can be used for training our model then it would certainly improve its performance. Works like those of Dai et al.[29] and Ramachandran et al. [105] state how autoencoders can be used to learn representations from the unlabelled data. More importantly, Ramachandran et al. [105] in their study provide a novel way to pretrain the *sequential* networks by using unsupervised *seq2seq* learning. Secondly, autoencoder models could also be used as an annotation tool that would annotate the unlabelled data and re-training (semi-supervised learning) them could improve the final classification.

Multi-Modality

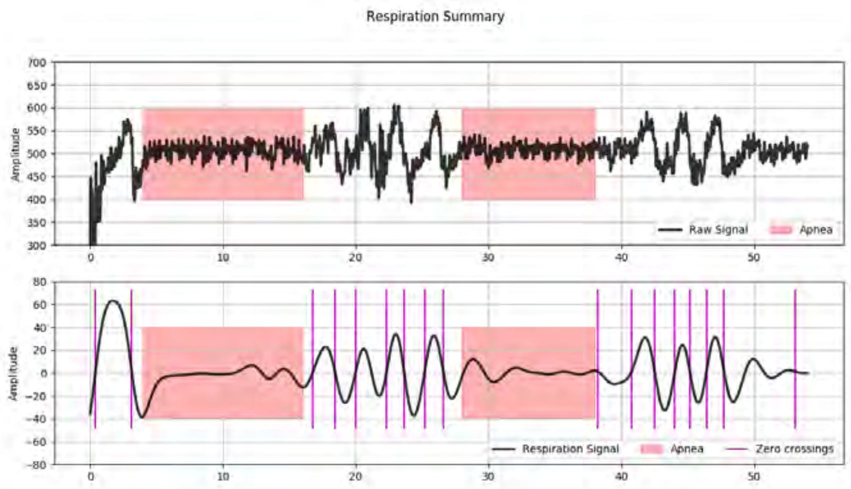


Figure 5.2: Apnea instances identified from the respiratory signal of the *Dozee BCG* data.

In this thesis, we specifically used only the heart signal from the BCG data. However, it has been explained in Section 2.1 and 2.1.2 that stages like *Wake*, *Deep*, and *REM* could be further differentiated using movement signal and breathing signal. One improvement in this direction could be to use multiple modalities to train the sleep classification model. This way, we can further capture the sleep patterns it its most truest form. Additionally, by incorporating different types of signals into model designing, we can build a multi-task model that can identify periods of snoring, irregular breathing, irregular heart beat and wakefulness. Figure 5.2 shows the identification of an apnea instance from the respiratory signal by using simple signal processing techniques. These additional predictions could further improve the model in making a decision about the stage scoring. Lastly, hand-engineered HRV features could be used to inject contextual knowledge into the model. A combination of expert's knowledge and a self-supervised end-to-end feature extractor and classification system could further improve the prediction ability of the model. Other external factors such as room temperature, luminosity and humidity could also be injected into the model's knowledge base.

Effect analysis

This work could also be used as a starting point to study the effect of factors such as alcohol consumption, smoking, physical activity and dietary-habits on the person's heart activity. Demographical and physiological studies like these could give the person a better insight into his health. Since wearable signals are easily recordable and available, explanation and recommendation could be incorporated to further guide people towards healthier lifestyle [102]. Effect analysis could further be used to study the effect of ambience factors like light, noise level, temperature and humidity on the overall quality of the sleep. This analysis could also help in improving the correlation between the objective sleep quality scores and the perceived scores and also help in treating serious sleep issues [8, 126].

Explainability

Visualization and explanation of classification steps of the model could give insights into improving the development and accuracy of the model. This way we can reduce the *black-box* nature of the deep learning models. One such implementation of explainability could be the usage of attention-based LSTM models for sequential learning. Attention-based models help us to selectively supervise the working of the RNN-based models by paying close attention to the internal contextual state sequences. We can check which parts of the signal have got higher attention from the sequential model. Attention mechanism is widely used in time-series classification [35] tasks such as sentiment analysis [136], video captioning [42] and speech recognition [69]. Using such attention mechanism we can understand why some of the sleep stages were predicted with a higher probability than others.

The explainability factor is useful in the healthcare domain. The predictions have to be accurate and should be able to explain why the model made a certain prediction. Apart from tracking, wearable users would want to know why their model assigned a lower or higher score to their sleep quality, for example. Explanation provides transparency into the model's working and makes it more reliable for health monitoring.

Privacy

Wearable signals are easily available owing to the sensors-in-the-wild paradigm. Although this provides a great benefit in the application of healthcare monitoring, it should be noted that this data is easily accessible too, raising privacy concerns. In our thesis, we faced difficulty in accurately predicting sleep stages from the PPG signal as these signals were obfuscated to preserve the privacy of the users. A different way of model designing and training should be studied which can form a balance between preserving data privacy and maintaining model's training ability. Works like that of Shokri and Shmatikov [117] and Li et al. [84] provide a way to perform machine learning while preserving privacy on the cloud by using double key encryption on datasets and allowing training only on a subset of the dataset.

Other ways to preserve privacy of the user data would be to run the machine learning models inside the sensor devices. Currently, most of the machine learning models run on large cloud servers. Because of this, it is required that the user data be transmitted and stored on the remote servers. Chances of stealing and illegal usage of data is higher if the server itself becomes vulnerable to attacks. Hence, edge computing should be employed wherein the models are deployed and run within the device (sensor device or mobile phone) itself [9, 130, 138]. This way the data can be limited locally to the device, thus, providing the users with more control over their data.

5.3. Conclusion

The goal of this study has been to model a sleep classification system using BCG sensor data such that it achieves a performance comparable to the medical standards. Although extensive studies have been performed in this field, several of the related works approached this problem using EEG, ECG, EMG and EOG signals. The mode of collection of these signals is highly obstructive and is not well suited for long-term sleep monitoring. Our proposed DeepSleep model potentially provides a way for long-term sleep monitoring from the comfort of one's home.

Our model achieved the research objective by using different preprocessing, model design and training strategies. Our hybrid model architecture enabled us to perform sleep classification with least pre-processing and feature engineering. By using stacks of CNN layers to learn features from the noisy BCG dataset, we were able to tackle the first research subquestion of modelling a robust subject-independent system. The shortcut connections along with the sequential layer ensured that the model pools features common to all the subjects and also differentiate them based on the bi-LSTM's context state information. Finally, by employing a 2-phase training strategy, we were able to address our second research subquestion of testing our model's transferability to other sensor types. By pre-training and fine-tuning our model using the BCG data, we were able to build a pre-trained model that has proved to perform well on the ECG sensor signal. Our DeepSleep model achieved an accuracy of 82%, 77% and 63% when tested on the MIT-BIH ECG, Dozee ECG, and the Fitbit-PPG dataset. With a correlation score of $r = 0.43$, our SATED-based user study on perceived sleep quality further showed that the subjective sleep quality is correlated to the objective sleep quality measurement reported by our model. Since our model identifies the sequences of stages significantly well, we show that with better objective scoring parameters and larger subjective study, we can further understand user's perception of his sleep quality.

The performance achieved by our DeepSleep model provides a first step towards sleep monitoring using non-invasive heart signal. The results show that this model, with some improvements, could be applied on ECG and PPG-based signals as well. With a mean f1-score of 74%, and our model's capability to learn the biological rules of sleep, we showed that it is possible to monitor, if not diagnose, our sleep patterns. Through our work, we provided a way to pre-train an end-to-end, sample-level model using BCG and ECG signals. We believe that this pre-trained model could also be used to further improve the state of the art in sleep monitoring tasks and other heart-related problems. This possibility could be a positive step in the healthcare domain as we, could track, monitor and quantify our sleep-health without having to forego our privacy and comfort.

Bibliography

[1] Best sleep trackers to look out for, 2018. URL `https://sleeptrackers.io/sleep-trackers/`.

[2] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

[3] Samuel Albanie, Arsha Nagrani, Andrea Vedaldi, and Andrew Zisserman. Emotion recognition in speech using cross-modal transfer in the wild. *arXiv preprint arXiv:1808.05561*, 2018.

[4] J Alihanka, K Vaahtoranta, and I Saarikivi. A new method for long-term monitoring of the ballistocardiogram, heart rate, and respiration. *American Journal of Physiology-Regulatory, Integrative and Comparative Physiology*, 240(5):R384–R392, 1981.

[5] John Allen. Photoplethysmography and its application in clinical physiological measurement. *Physiological measurement*, 28(3):R1–39, mar 2007. ISSN 0967-3334 (Print). doi: 10.1088/0967-3334/28/3/R01.

[6] John Allen. Photoplethysmography and its application in clinical physiological measurement. *Physiological measurement*, 28(3):R1, 2007.

[7] Tim Althoff, Eric Horvitz, Ryen W White, and Jamie Zeitzer. Harnessing the web for population-scale physiological sensing: A case study of sleep and performance. In *Proceedings of the 26th International Conference on World Wide Web*, pages 113–122. International World Wide Web Conferences Steering Committee, 2017.

[8] Federico Alvarez, Mirela Popa, Vassilios Solachidis, Gustavo Hernández-Peñaloza, Alberto Belmonte-Hernández, Stylianos Asteriadis, Nicholas Vretos, Marcos Quintana, Thomas Theodoridis, Dario Dotti, et al. Behavior analysis through multimodal sensing for care of parkinson's and alzheimer's patients. *IEEE MultiMedia*, 25(1):14–25, 2018.

[9] Pete Beckman, Rajesh Sankaran, Charlie Catlett, Nicola Ferrier, Robert Jacob, and Michael Papka. Waggle: An open sensor platform for edge computing. In *SENSORS, 2016 IEEE*, pages 1–3. IEEE, 2016.

[10] Yoshua Bengio. Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML Workshop on Unsupervised and Transfer Learning*, pages 17–36, 2012.

[11] Siddharth Biswal, Joshua Kulas, Haoqi Sun, Balaji Goparaju, M Brandon Westover, Matt T Bianchi, and Jimeng Sun. Sleepnet: Automated sleep staging system via deep learning. *arXiv preprint arXiv:1707.08262*, 2017.

[12] Manuel Blanco-Velasco, Binwei Weng, and Kenneth E Barner. Ecg signal denoising and baseline wander correction based on the empirical mode decomposition. *Computers in biology and medicine*, 38(1):1–13, 2008.

[13] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[14] Y-Lan Boureau, Jean Ponce, and Yann LeCun. A theoretical analysis of feature pooling in visual recognition. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 111–118, 2010.

[15] Daniel J Buysse. Sleep health: can we define it? does it matter? *Sleep*, 37(1):9–17, 2014.

[16] Daniel J Buysse, Charles F Reynolds III, Timothy H Monk, Susan R Berman, and David J Kupfer. The pittsburgh sleep quality index: a new instrument for psychiatric practice and research. *Psychiatry research*, 28(2):193–213, 1989.

[17] A John Camm, Marek Malik, J Thomas Bigger, Günter Breithardt, Sergio Cerutti, Richard J Cohen, Philippe Coumel, Ernest L Fallen, Harold L Kennedy, RE Kleiger, et al. Heart rate variability. standards of measurement, physiological interpretation, and clinical use. *European heart journal*, 17(3):354–381, 1996.

[18] Janet S Carpenter and Michael A Andrykowski. Psychometric evaluation of the pittsburgh sleep quality index. *Journal of psychosomatic research*, 45(1):5–13, 1998.

[19] Meredith A Case, Holland A Burwick, Kevin G Volpp, and Mitesh S Patel. Accuracy of smartphone applications and wearable devices for tracking physical activity data. *Jama*, 313(6):625–626, 2015.

[20] Nicola Cellini, Katherine A Duggan, and Michela Sarlo. Perceived sleep quality: The interplay of neuroticism, affect, and hyperarousal. *Sleep health*, 3(3):184–189, 2017.

[21] Stanislas Chambon, Mathieu N Galtier, Pierrick J Arnal, Gilles Wainrib, and Alexandre Gramfort. A deep learning architecture for temporal sleep stage classification using multivariate and multimodal time series. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2018.

[22] Sun-Taag Choe and We-Duke Cho. Simplified real-time heartbeat detection in ballistocardiography using a dispersion-maximum method. *Biomedical Research*, 28(9), 2017.

[23] Sudhansu Chokroverty, Robert J Thomas, and Meeta Bhatt. *Atlas of Sleep Medicine E-Book*. Elsevier Health Sciences, 2013.

[24] François Chollet et al. Keras. `https://keras.io`, 2015.

[25] Jason C Cole, Sarosh J Motivala, Daniel J Buysse, Michael N Oxman, Myron J Levin, and Michael R Irwin. Validation of a 3-factor scoring model for the pittsburgh sleep quality index in older adults. *Sleep*, 29(1):112–116, 2006.

[26] Alexandru Corlateanu, Serghei Covantev, Victor Botnaru, Victoria Sircu, and Raffaella Nenna. To sleep, or not to sleep – that is the question, for polysomnography. *Breathe*, 13(2):137 LP – 140, jun 2017. URL `http://breathe.ersjournals.com/content/13/2/137.abstract`.

[27] Gemma Curtis. Your life in numbers - the sleep matters club, 2018. URL `https://www.dreams.co.uk/sleep-matters-club/your-life-in-numbers-infographic/`.

[28] George E Dahl, Tara N Sainath, and Geoffrey E Hinton. Improving deep neural networks for lvcsr using rectified linear units and dropout. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 8609–8613. IEEE, 2013.

[29] Andrew M Dai and Quoc V Le. Semi-supervised sequence learning. In *Advances in Neural Information Processing Systems*, pages 3079–3087, 2015.

[30] Rodolfo de Figueiredo Dalvi, Gabriel Tozatto Zago, and Rodrigo Varejão Andreão. Heartbeat classification system based on neural networks and dimensionality reduction. *Research on Biomedical Engineering*, 32(4):318–326, 2016.

[31] Jordi De Batlle, Ivan D Benitez, Mireia Dalmases, Anna Mas, Oriol Garcia-Codina, Antonia Medina-Bustos, Joan Escarrabill, Esteve Salto, Manuel Sanchez-de-la Torre, and Ferran Barbe. Usefulness of a 5-item scale to assess sleep health status: Results of the catalan health survey 2015. In *C80-D. SLEEP AND HEALTH POLICY*, pages A6535–A6535. American Thoracic Society, 2017.

[32] Sandrine Devot, Anna M Bianchi, Elke Naujoka, Martin O Mendez, Andreas Braurs, and Sergio Cerutti. Sleep monitoring through a textile recording system. In *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE*, pages 2560–2563. IEEE, 2007.

[33] Jishnu Dey, Tanmoy Bhowmik, Saswata Sahoo, and Vijay Narayan Tiwari. Wearable ppg sensor based alertness scoring system. In *Engineering in Medicine and Biology Society (EMBC), 2017 39th Annual International Conference of the IEEE*, pages 2422–2425. IEEE, 2017.

[34] Hao Dong, Akara Supratak, Wei Pan, Chao Wu, Paul M Matthews, and Yike Guo. Mixed neural network approach for temporal sleep stage classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(2):324–333, 2018.

[35] Qianjin Du, Weixi Gu, Lin Zhang, and Shao-Lun Huang. Attention-based lstm-cnns for time-series classification. In *Proceedings of the 16th ACM Conference on Embedded Networked Sensor Systems*, pages 410–411. ACM, 2018.

[36] Jeffrey S Durmer and David F Dinges. Neurocognitive consequences of sleep deprivation. In *Seminars in neurology*, volume 25, pages 117–129. Copyright© 2005 by Thieme Medical Publishers, Inc., 333 Seventh Avenue, New York, NY 10001, USA., 2005.

[37] Kelly R Evenson, Michelle M Goto, and Robert D Furberg. Systematic review of the validity and reliability of consumer-wearable activity trackers. *International Journal of Behavioral Nutrition and Physical Activity*, 12(1):159, 2015.

[38] Fitness Facts. Topic: Fitness & activity tracker, 2018. URL `https://www.statista.com/topics/4393/fitness-and-activity-tracker/`.

[39] Pedro Fonseca, Xi Long, Mustafa Radha, Reinder Haakma, Ronald M Aarts, and Jérôme Rolink. Sleep stage classification with ecg and respiratory effort. *Physiological measurement*, 36(10):2027, 2015.

[40] Pedro Fonseca, Niek den Teuling, Xi Long, and Ronald M Aarts. Cardiorespiratory sleep stage detection using conditional random fields. *IEEE journal of biomedical and health informatics*, 21(4):956–966, 2017.

[41] Centers for Disease Control, Prevention (CDC, et al. Unhealthy sleep-related behaviors–12 states, 2009. *MMWR. Morbidity and mortality weekly report*, 60(8):233, 2011.

[42] Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu, and Heng Tao Shen. Video captioning with attention-based lstm and semantic consistency. *IEEE Transactions on Multimedia*, 19(9):2045–2055, 2017.

[43] Laurent Giovangrandi, Omer T Inan, Richard M Wiard, Mozziyar Etemadi, and Gregory TA Kovacs. Ballistocardiography—a method worth revisiting. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pages 4279–4282. IEEE, 2011.

[44] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256, 2010.

[45] Ary L Goldberger, Luis A Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23):E215–20, 2000.

[46] Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. Current perspective. *Circulation*, 101:e215–e220, 2000.

[47] Zhichen Gong and Huanhuan Chen. Model-based oversampling for imbalanced sequence classification. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1009–1018. ACM, 2016.

[48] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

[49] Alex Graves, Navdeep Jaitly, and Abdel-rahman Mohamed. Hybrid speech recognition with deep bidirectional lstm. In *Automatic Speech Recognition and Understanding (ASRU), 2013 IEEE Workshop on*, pages 273–278. IEEE, 2013.

[50] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*, pages 6645–6649. IEEE, 2013.

[51] Kristina Grifantini. How's my sleep?: Personal sleep trackers are gaining in popularity, but their accuracy is still open to debate. *IEEE pulse*, 5(5):14–18, 2014.

[52] Pat Hamilton. Open source ecg analysis. In *Computers in Cardiology, 2002*, pages 101–104. IEEE, 2002.

[53] John Harrington, Preetam J Schramm, Charles R Davies, and Teofilo L Lee-Chiong. An electrocardiogram-based analysis evaluating sleep quality in patients with obstructive sleep apnea. *Sleep and Breathing*, 17(3):1071–1078, 2013.

[54] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[55] Sonja Hermann, Luca Lombardo, Giuseppe Campobello, Martin Burke, and Nicola Donato. A ballistocardiogram acquisition system for respiration and heart rate monitoring. In *2018 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pages 1–5. IEEE, 2018.

[56] Geoffrey E Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

[57] Arianna Huffington. *The Sleep Revolution: Transforming Your Life, One Night at a Time*. Harmony, 2016.

[58] Minyoung Huh, Pulkit Agrawal, and Alexei A Efros. What makes imagenet good for transfer learning? *arXiv preprint arXiv:1608.08614*, 2016.

[59] Omer T Inan, Pierre-Francois Migeotte, Kwang-Suk Park, Mozziyar Etemadi, Kouhyar Tavakolian, Ramon Casanella, John Zanetti, Jens Tank, Irina Funtova, G Kim Prisk, et al. Ballistocardiography and seismocardiography: A review of recent advances. *IEEE journal of biomedical and health informatics*, 19(4):1414–1427, 2015.

[60] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

[61] Anil Jain and Douglas Zongker. Feature selection: Evaluation, application, and small sample performance. *IEEE transactions on pattern analysis and machine intelligence*, 19(2):153–158, 1997.

[62] L Jeon and Joseph Finkelstein. Consumer sleep tracking devices: a critical review. *Digital Healthcare Empowering Europeans: Proceedings of MIE2015*, 210:458, 2015.

[63] Murray W Johns. A new method for measuring daytime sleepiness: the epworth sleepiness scale. *sleep*, 14(6):540–545, 1991.

[64] Murray W Johns. Daytime sleepiness, snoring, and obstructive sleep apnea: the epworth sleepiness scale. *Chest*, 103(1):30–36, 1993.

[65] Murrayb W Johns. Reliability and factor analysis of the epworth sleepiness scale. *Sleep*, 15(4):376–381, 1992.

[66] Mohammad Kachuee, Mohammad Mahdi Kiani, Hoda Mohammadzade, and Mahdi Shabany. Cuffless blood pressure estimation algorithms for continuous health-care monitoring. *IEEE Transactions on Biomedical Engineering*, 64(4):859–869, 2017.

[67] M Kania, M Fereniec, and R Maniewski. Wavelet denoising for multi-lead high resolution ecg signals. *Measurement science review*, 7(4):30–33, 2007.

[68] Chang-Sei Kim, Stephanie L Ober, M Sean McMurtry, Barry A Finegan, Omer T Inan, Ramakrishna Mukkamala, and Jin-Oh Hahn. Ballistocardiogram: Mechanism and potential for unobtrusive cardiovascular health monitoring. *Scientific reports*, 6:31297, 2016.

[69] Suyoun Kim, Takaaki Hori, and Shinji Watanabe. Joint ctc-attention based end-to-end speech recognition using multi-task learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pages 4835–4839. IEEE, 2017.

[70] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[71] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[72] Kristen L Knutson, Karine Spiegel, Plamen Penev, and Eve Van Cauter. The metabolic consequences of sleep deprivation. *Sleep medicine reviews*, 11(3):163–178, 2007.

[73] Juha M Kortelainen, Martin O Mendez, Anna Maria Bianchi, Matteo Matteucci, and Sergio Cerutti. Sleep staging based on signals acquired through bed sensor. *IEEE Transactions on Information Technology in Biomedicine*, 14(3):776–785, 2010.

[74] Anastasi Kosmadopoulos, Charli Sargent, David Darwent, Xuan Zhou, and Gregory D Roach. Alternatives to polysomnography (psg): a validation of wrist actigraphy and a partial-psg system. *Behavior research methods*, 46(4):1032–1041, 2014.

[75] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[76] Andrew D Krystal and Jack D Edinger. Measuring sleep quality. *Sleep medicine*, 9: S10–S17, 2008.

[77] Clete A Kushida, Arthur Chang, Chirag Gadkary, Christian Guilleminault, Oscar Carrillo, and William C Dement. Comparison of actigraphic, polysomnographic, and subjective assessment of sleep parameters in sleep-disordered patients. *Sleep medicine*, 2 (5):389–396, 2001.

[78] Glenn J Landry, John R Best, and Teresa Liu-Ambrose. Measuring sleep quality in older adults: a comparison using subjective and objective methods. *Frontiers in aging neuroscience*, 7:166, 2015.

[79] Martin Längkvist, Lars Karlsson, and Amy Loutfi. Sleep stage classification using unsupervised feature learning. *Advances in Artificial Neural Systems*, 2012, 2012.

[80] Jongpil Lee, Jiyoung Park, Keunhyoung Luke Kim, and Juhan Nam. Sample-level deep convolutional neural networks for music auto-tagging using raw waveforms. *arXiv preprint arXiv:1703.01789*, 2017.

[81] Jung-Min Lee, Young-Won Kim, and Gregory J Welk. Track it: Validity and utility of consumer-based physical activity monitors. *ACSM's Health & Fitness Journal*, 18(4): 16–21, 2014.

[82] Henry Leung and Simon Haykin. The complex backpropagation algorithm. *IEEE Transactions on signal processing*, 39(9):2101–2104, 1991.

[83] Changzhi Li, Victor M Lubecke, Olga Boric-Lubecke, and Jenshan Lin. A review on recent advances in doppler radar sensors for noncontact healthcare monitoring. *IEEE Transactions on microwave theory and techniques*, 61(5):2046–2060, 2013.

[84] Ping Li, Jin Li, Zhengan Huang, Tong Li, Chong-Zhi Gao, Siu-Ming Yiu, and Kai Chen. Multi-key privacy-preserving deep learning in cloud computing. *Future Generation Computer Systems*, 74:76–85, 2017.

[85] Feng Lin, Yan Zhuang, Chen Song, Aosen Wang, Yiran Li, Changzhan Gu, Changzhi Li, and Wenyao Xu. Sleepsense: A noncontact and cost-effective sleep monitoring system. *IEEE transactions on biomedical circuits and systems*, 11(1):189–202, 2017.

[86] Michael Littner, Max Hirshkowitz, Milton Kramer, Sheldon Kapen, W McDowell Anderson, Dennis Bailey, Richard B Berry, David Davila, Stephen Johnson, Clete Kushida, et al. Practice parameters for using polysomnography to evaluate insomnia: an update. *Sleep*, 26(6):754–760, 2003.

[87] André Lourenço, Hugo Silva, and Ana Fred. Unveiling the biometric potential of finger-based ecg signals. *Computational intelligence and neuroscience*, 2011:5, 2011.

[88] David C Mack, James T Patrie, Paul M Suratt, Robin A Felder, and Majd Alwan. Development and preliminary validation of heart rate and breathing rate detection using a passive, ballistocardiography-based sleep monitoring system. *IEEE Transactions on Information Technology in Biomedicine*, 13(1):111–120, 2009.

[89] Amol D Mali. Recent advances in minimally-obtrusive monitoring of people's health. *International Journal of Monitoring and Surveillance Technologies Research (IJMSTR)*, 5 (2):44–56, 2017.

[90] Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber. Stacked convolutional auto-encoders for hierarchical feature extraction. In *International Conference on Artificial Neural Networks*, pages 52–59. Springer, 2011.

[91] Hawley E Montgomery-Downs, Salvatore P Insana, and Jonathan A Bond. Movement toward a novel activity monitoring device. *Sleep and Breathing*, 16(3):913–917, 2012.

[92] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010.

[93] Hongbo Ni, Mingjie He, Guoxing Xu, Yalong Song, and Xingshe Zhou. Extracting heartbeat intervals using self-adaptive method based on ballistocardiography (bcg). In *International Conference on Smart Homes and Health Telematics*, pages 37–47. Springer, 2017.

[94] Yuval Nir, Thomas Andrillon, Amit Marmelshtein, Nanthia Suthana, Chiara Cirelli, Giulio Tononi, and Itzhak Fried. Selective neuronal lapses precede human cognitive lapses following sleep deprivation. *Nature medicine*, 23(12):1474, 2017.

[95] National Institutes of Health et al. Brain basics: understanding sleep. *NIH Publication*, (06-3440), 2014.

[96] Maurice Ohayon, Emerson M Wickwire, Max Hirshkowitz, Steven M Albert, Alon Avidan, Frank J Daly, Yves Dauvilliers, Raffaele Ferri, Constance Fung, David Gozal, et al. National sleep foundation's sleep quality recommendations: first report. *Sleep Health*, 3(1):6–19, 2017.

[97] Ramaswamy Palaniappan and Shankar M Krishnan. Identifying individuals using ecg beats. In *Signal Processing and Communications, 2004. SPCOM'04. 2004 International Conference on*, pages 569–572. IEEE, 2004.

[98] Jiapu Pan and Willis J Tompkins. A real-time qrs detection algorithm. *IEEE Trans. Biomed. Eng*, 32(3):230–236, 1985.

[99] O Parra, N Garcia-Esclasans, JM Montserrat, L García Eroles, J Ruiz, JA López, JM Guerra, and JJ Sopena. Should patients with sleep apnoea/hypopnoea syndrome be diagnosed and managed on the basis of home sleep studies? *European Respiratory Journal*, 10(8):1720–1724, 1997.

[100] Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. Understanding the exploding gradient problem. *CoRR, abs/1211.5063*, 2012.

[101] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830, 2011.

[102] Xuefeng Peng, Jiebo Luo, Catherine Glenn, Jingyao Zhan, and Yuhan Liu. Large-scale sleep condition analysis using selfies from social media. In *International Conference on Social Computing, Behavioral-Cultural Modeling and Prediction and Behavior Representation in Modeling and Simulation*, pages 151–161. Springer, 2017.

[103] Esteban J Pino, Javier AP Chávez, and Pablo Aqueveque. Bcg algorithm for unobtrusive heart rate monitoring. In *Healthcare Innovations and Point of Care Technologies (HI-POCT), 2017 IEEE*, pages 180–183. IEEE, 2017.

[104] Pranav Rajpurkar, Awni Y. Hannun, Masoumeh Haghpanahi, Codie Bourn, and Andrew Y. Ng. Cardiologist-level arrhythmia detection with convolutional neural networks. *CoRR*, abs/1707.01836, 2017. URL http://arxiv.org/abs/1707.01836.

[105] Prajit Ramachandran, Peter J Liu, and Quoc V Le. Unsupervised pretraining for sequence to sequence learning. *arXiv preprint arXiv:1611.02683*, 2016.

[106] Allan Rechtschaffen. A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects. *Public health service*, 1968.

[107] Stephen J Redmond, Philip de Chazal, Ciara O'Brien, Silke Ryan, Walter T McNicholas, and Conor Heneghan. Sleep staging using cardiorespiratory signals. *Somnologie-Schlafforschung und Schlafmedizin*, 11(4):245–256, 2007.

[108] Philippe Renevey, Ricard Delgado-Gonzalo, Alia Lemkaddem, Martin Proença, Mathieu Lemay, Josep Solà, Adrian Tarniceriu, and Mattia Bertschi. Optical wrist-worn device for sleep monitoring. In *EMBEC & NBC 2017*, pages 615–618. Springer, 2017.

[109] Brady A Riedner, Michael R Goldstein, David T Plante, Meredith E Rumble, Fabio Ferrarelli, Giulio Tononi, and Ruth M Benca. Regional patterns of elevated alpha and high-frequency electroencephalographic activity during nonrapid eye movement sleep in chronic insomnia: a pilot study. *Sleep*, 39(4):801–812, 2016.

[110] Haşim Sak, Andrew Senior, and Françoise Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Fifteenth annual conference of the international speech communication association*, 2014.

[111] Lauren Samy, Ming-Chun Huang, Jason J Liu, Wenyao Xu, and Majid Sarrafzadeh. Unobtrusive sleep stage identification using a pressure-sensitive bed sheet. *IEEE Sensors Journal*, 14(7):2092–2101, 2014.

[112] William R Scarborough, Edgar F Folk, Patricia M Smith, and Joseph H Condon. The nature of records from ultra-low frequency ballistocardiographic systems and their relation to circulatory events. *American Journal of Cardiology*, 2(5):613–641, 1958.

[113] P Scherer, JP Ohler, H Hirche, and HW Höpp. Definition of a new beat-to-beat-parameter of heart rate variability (abstr). *Pacing Clin Electrophys*, 16:939, 1993.

[114] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[115] Tushar Semwal, Promod Yenigalla, Gaurav Mathur, and Shivashankar B Nair. A practitioners' guide to transfer learning for text classification using convolutional neural networks. In *Proceedings of the 2018 SIAM International Conference on Data Mining*, pages 513–521. SIAM, 2018.

[116] Valeriy Sharapov. *Piezoceramic sensors*. Springer Science & Business Media, 2011.

[117] Reza Shokri and Vitaly Shmatikov. Privacy-preserving deep learning. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pages 1310–1321. ACM, 2015.

[118] Michael H Silber, Sonia Ancoli-Israel, Michael H Bonnet, Sudhansu Chokroverty, Madeleine M Grigg-Damberger, Max Hirshkowitz, Sheldon Kapen, Sharon A Keenan, Meir H Kryger, Thomas Penzel, et al. The visual scoring of sleep in adults. *Journal of Clinical Sleep Medicine*, 3(02):22–22, 2007.

[119] Yannick Stephan, Angelina R Sutin, Sophie Bayard, Zlatan Križan, and Antonio Terracciano. Personality and sleep quality: Evidence from four prospective studies. *Health Psychology*, 37(3):271, 2018.

[120] Yu Sun and Nitish Thakor. Photoplethysmography revisited: from contact to noncontact, from point to imaging. *IEEE Transactions on Biomedical Engineering*, 63(3): 463–477, 2016.

[121] Akara Supratak, Hao Dong, Chao Wu, and Yike Guo. Deepsleepnet: A model for automatic sleep stage scoring based on raw single-channel eeg. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(11):1998–2008, 2017.

[122] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112, 2014.

[123] Masashi Tabuchi and Mark N Wu. Sleep: Setting the 'circadian'alarm clock. *Current Biology*, 28(1):R26–R28, 2018.

[124] Maria Thomas, Helen Sing, Gregory Belenky, Henry Holcomb, Helen Mayberg, Robert Dannals, Henry Wagner JR, David Thorne, Kathryn Popp, Laura Rowland, et al. Neural basis of alertness and cognitive performance impairments during sleepiness. i. effects of 24 h of sleep deprivation on waking human regional brain activity. *Journal of sleep research*, 9(4):335–352, 2000.

[125] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

[126] Carlos Torres, Jeffrey C Fried, Kenneth Rose, and Bangalore S Manjunath. A multiview multimodal system for monitoring patient sleep. *IEEE Transactions on Multimedia*, 2018.

[127] Lisa Torrey and Jude Shavlik. Transfer learning. In *Handbook of Research on Machine Learning Applications and Trends: Algorithms, Methods, and Techniques*, pages 242–264. IGI Global, 2010.

[128] Orestis Tsinalis, Paul M Matthews, Yike Guo, and Stefanos Zafeiriou. Automatic sleep stage scoring with single-channel eeg using convolutional neural networks. *arXiv preprint arXiv:1610.01683*, 2016.

[129] Alexander TM Van De Water, Alison Holmes, and Deirdre A Hurley. Objective measurements of sleep for non-laboratory settings as alternatives to polysomnography–a systematic review. *Journal of sleep research*, 20(1pt2):183–200, 2011.

[130] Blesson Varghese, Nan Wang, Sakil Barbhuiya, Peter Kilpatrick, and Dimitrios S Nikolopoulos. Challenges and opportunities in edge computing. *arXiv preprint arXiv:1609.01967*, 2016.

[131] Lorcan Walsh, Seán McLoone, Joseph Ronda, Jeanne F Duffy, and Charles A Czeisler. Noncontact pressure-based sleep/wake discrimination. *IEEE Transactions on Biomedical Engineering*, 64(8):1750–1760, 2017.

[132] Yuhua Wang, Sammia Ali, Jayawan Wijekoon, R Hugh Gong, and Anura Fernando. A wearable piezo-resistive sensor for capturing cardiorespiratory signals. *Sensors and Actuators A: Physical*, 2018.

[133] Mark Weiser, Rich Gold, and John Seely Brown. The origins of ubiquitous computing research at parc in the late 1980s. *IBM systems journal*, 38(4):693–696, 1999.

[134] Meng Xiao, Hong Yan, Jinzhong Song, Yuzhou Yang, and Xianglin Yang. Sleep stages classification based on heart rate variability and random forest. *Biomedical Signal Processing and Control*, 8(6):624–633, 2013.

[135] Heetae Yang, Jieun Yu, Hangjung Zo, and Munkee Choi. User acceptance of wearable devices: An extended perspective of perceived value. *Telematics and Informatics*, 33(2): 256–269, 2016.

[136] Min Yang, Wenting Tu, Jingxuan Wang, Fei Xu, and Xiaojun Chen. Attention based lstm for target dependent sentiment classification. In *AAAI*, pages 5013–5014, 2017.

[137] Dong Yu and Michael L Seltzer. Improved bottleneck features using pretrained deep neural networks. In *Twelfth annual conference of the international speech communication association*, 2011.

[138] Jianfei Yu, Minghui Qiu, Jing Jiang, Jun Huang, Shuangyong Song, Wei Chu, and Haiqing Chen. Modelling domain relationships for transfer learning on retrieval-based question answering systems in e-commerce. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 682–690. ACM, 2018.

[139] Qingxue Zhang, Dian Zhou, and Xuan Zeng. Heartid: a multiresolution convolutional neural network for ecg-based biometric human identification in smart health applications. *IEEE Access*, 5:11805–11816, 2017.