Machine Learning Methods for Sleep Apnoea Detection Based on Pulse and Oximetry Data

Dongjin Yang

Supervisor: Prof. Lyudmila Mihaylova

School of Electrical and Electronic Engineering

University of Sheffield

This dissertation is submitted for the degree of

Doctor of Philosophy

October 2025

Declaration

All sentences or passages quoted in this document from other people's work have been specifically acknowledged by clear cross-referencing to author, work and page(s). Any illustrations that are not the work of the author of this report have been used with the explicit permission of the originator and are specifically acknowledged. I understand that failure to do this amounts to plagiarism and will be considered grounds for failure.

Dongjin Yang

Supervisor: Prof. Lyudmila Mihaylova

October 2025

Acknowledgements

First, I would like to express my deepest gratitude to my supervisor, Prof. Lyudmila S Mihaylova, for her exceptional guidance, unwavering support, and continuous encouragement throughout my doctoral studies. Her mentorship has been invaluable to my research and personal and academic growth, significantly boosting my confidence and development as an independent scholar. In Chinese, we have a word, GuiRen, which refers to someone who brings you support, wisdom, or good fortune at critical times. Prof. Lyudmila has been such a person to me. Her thoughtful advice and kind concern have guided my academic journey and brought warmth and strength to my daily life.

I would also like to sincerely thank my second supervisor, Prof. Heather Elphick. As a practising clinician with a demanding schedule, she generously took the time to help me refine the medical aspects of my papers. Her professional insights and careful guidance have greatly improved the accuracy and clarity of my work.

Special thanks go to Dr. Jingqiong Zhang, who carefully reviewed and improved every sentence of my journal and conference papers. Her attention to detail and generous help have supported me throughout this journey.

I am deeply grateful to my mother and family for their endless love. Without their unwavering support and encouragement, I would not have been able to start and persevere throughout my PhD journey. I would also like to thank my boyfriend, Guoxi Ye, for his invaluable patience, support, and companionship. His presence has been a constant strength and comfort during this challenging and joyful journey.

Lastly, my heartfelt thanks go to my two cats, Zimy and Gao, whose quiet companionship, mischief, and comforting presence brought light and calm to my long writing nights. In their gentle way, they reminded me that life is not just about deadlines and data, but also about warmth, softness, and moments of stillness.

Abstract

Sleep apnoea is a disease that affects children and adults and can lead to cardiovascular disease, diabetes, and cognitive impairment in severe cases. The Polysomnography is recognised as the golden diagnostic method, but it is expensive and time-consuming, making it impossible to conduct widespread screening. This thesis first introduces the background of automatic sleep apnoea detection and commonly used change detection algorithms. Chapter 3 introduces an anomaly detection method that uses an adaptive Cumulative Sum (CUSUM) change point detection algorithm to monitor outliers in the signal. The test results of the adaptive CUSUM are compared with those of the classic CUSUM. Chapter 4 proposes a novel framework for extracting features from sleep signals using wavelet transforms and uses the RUSBoost algorithm to address the data imbalance problem in sleep apnoea detection. This chapter evaluates classic machine learning methods, such as support vector machines (SVM), k-nearest neighbours, Dirichlet process Gaussian mixture model, and the ensemble method Random Undersampling Boosting (RUSBoost), which aim to address the data imbalance problem. In addition, this chapter utilises feature fusion techniques to evaluate the performance of single-signal detection and multi-signal detection. Chapter 5 presents and compares deep learning approaches, specifically Convolutional Neural Network (CNN), CNN with SVM and Recurrent Neural Network architectures. The signal-level fusion strategy enhances detection sensitivity significantly.

All proposed approaches are tested on public datasets in different environments, including the Apnoea-ECG database [1], the Childhood Adenotonsillectomy Trial database [2], and St Vincent hospital [3], demonstrating the

effectiveness of the proposed methods in identifying apnoea events in different situations.

Table of contents

List of figures

List of tables

Nomenclature

Acronyms / Abbreviations

AASM	American Academy of Sleep Medicine
ACC	Accuracy
AHI	Apnoea-Hypopnoea Index
AI	Artificial intelligence
ANN	Artificial neural networks
ASR	Automated speech recognition
AUC	Area of curve
BAFNet	Bottleneck attention fusion network
BiFA	Bitemporal Feature Alignment
BiGRU	Bidirectional gated recurrent unit
BiLSTM	Bidirectional long short-term memory
BOCD	Bayesian Online Change Detection
CHAT	Childhood Adenotonsillectomy Trial
CNN	Convolutional neural network
CPD	Change point detection

CSA	Central sleep apnoea
CT	Computed tomography
$CUSUM$	Cumulative Sum method
CWT	Continues wavelet transform
DFT	Discrete Fourier transform
DP	Dirichlet Process
DWT	Discrete wavelet transform
ECG	Electrocardiography
EEG	Electroencephalography
EMG	Electromyography
EOG	Electrooculography
$ESE\text{-}FN$	Expansion-squeeze-excitation fusion network
$EWMA$	Exponentially Weighted Moving Average
FN	False Negative
FP	False positive
$GLRT$	Generalized Likelihood Ratio Test
GMM	Gaussian Mixture Model
GP	Gaussian processing
GRU	Gated recurrent unit
HRV	Heart rate variability
kNN	k-nearest neighbours

LLR	Log-likelihood Ratio
LRT	Log-likelihood Ratio Test
$LSTM$	Long short-term memory
$MFCC$	Mel-frequency cepstral coefficient
MLP	Multi-layer perceptrons
MRI	Magnetic resonance imaging
MS	Multiple sclerosis
MSA	Mixed sleep apnoea
MSE	Merge Squeeze Excitation
NIG	Normal Inverse Gaussian
$NREM$	Non-rapid eye movement
$NSRR$	National Sleep Research Resource
$SYNC$	Normalized Difference Vegetation Index
ODI	Oxygen Desaturation Index
OSA	Obstructive sleep apnoea
PAP	Positive Airway Pressure
pdf	Probability density function
$PDFA$	Progressive Detrended Fluctuation Analysis
PSD	Power Spectral Density
PSG	Polysomnography
ReLU	Rectified Linear Unit

REM	Rapid eye movement
RF	Random Forest
RM	Recursive Merging
RNN	Recurrent neural network
ROC	Receiver Operating Characteristic
RPA	R-peak amplitude
$rPPG$	Remote photoplethysmography
RRI	R-R interva
$RSNN$	Recursive Spiking Neural Network
RUS	Random under-sampling
$SAAS$	Sleep Apnoea Analysing Strategy
$SARIMA$	Autoregressive and Integrated Moving Average
SDB	sleep disordered breathing
SENS	Sensitivity
$SMOTE$	Synthetic Minority Over-sampling Technique
SPE	specificity
$SPO2$	oxygen saturation
SST	Singular Spectrum Transformation
SVM	Support Vector Machine
$SYNC$	Synthetic
TN	True negative

TP	True positive
$TQWT$	Tunable-Q factor wavelet transform
WT	Wavelet transform

Chapter 1

Introduction

1.1 Background and Motivation

Humans spend about a third of their lives sleeping. Sleep disorders, such as insomnia and obstructive sleep apnoea, severely impact the quality of life of patients. Without restrictive criteria, the prevalence of insomnia is approximately 33% in the general population [5]. Patients with sleep apnoea experience periods of no or shallow breathing while sleeping. The former situation, in which breathing temporarily stops, is referred to as apnoea, whereas the latter, in which breathing is shallow or airflow is restricted, is referred to as hypopnoea. Both diseases can result in clinical comorbidities and thus harm human health [6].

The physiological manifestations of sleep apnoea include nocturnal snoring, episodic gasping, xerostomia upon awakening, and overall diminished sleep quality. These diseases commonly cause impaired cognitive focus, worsening of insomnia, cognitive decline, increased risk of accidents, memory impairment, and depression. Furthermore, sleep apnoea may result in several serious health issues, such as diabetes, cardiovascular complications, high blood pressure, neurological issues, and liver problems [7]. These complications significantly impair life quality due to chronic fatigue and insufficient sleep [8]. Sleep apnoea affects approximately 1% to 6% of adults and 1.2% to 5.7% of children [9]. Given its widespread occurrence worldwide and its lasting impact on sleep, it is necessary to detect and treat sleep apnoea [10].

There are three types of sleep apnoea [11]:

1) Obstructive sleep apnoea (OSA) is the more frequent pattern, characterized by the presence of thoracic effort for continuing breathing while air flow completely stops. When the hard palate muscles at the back of the throat that support the soft palate relax, the soft palate blocks air from entering the respiratory system. This causes short-term cessation of breathing.

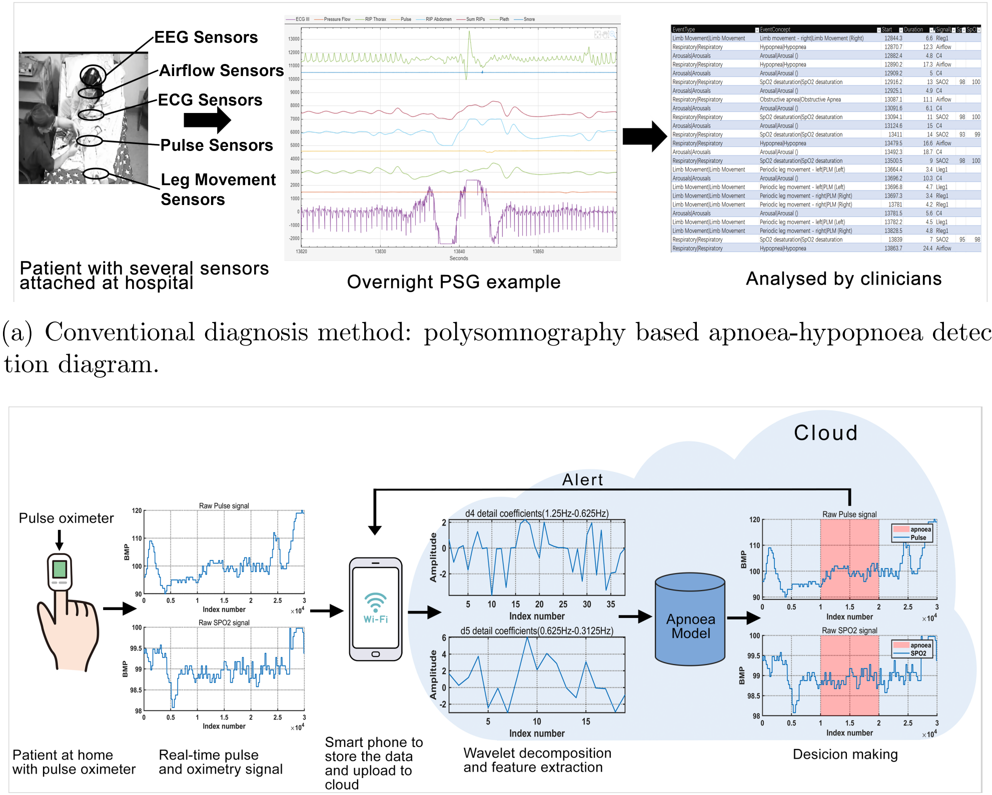
2) Central sleep apnoea (CSA) occurs when the brain fails to generate or transmit the signals that control the muscles of breathing. It is characterized by a complete cessation of respiratory movement and airflow for at least 10 seconds.

3) Mixed sleep apnoea (MSA): Complex sleep apnoea syndrome is characterized by persistent central apnoea even after obstructive events have resolved after Positive Airway Pressure (PAP) therapy. This pattern is a combination of the first two patterns, defined as central apnoea followed by obstructive ventilatory efforts in relatively short time intervals.

Polysomnography (PSG) is a gold-standard diagnostic test used to study sleep and diagnose various sleep disorders. Some people refer to PSG as a sleep study. Sleep technologists perform tests that are usually performed in hospitals, independent facilities, or specialized sleep clinics [12]. Sleep monitoring is a complex process and requires a unique set of skills, including detailed knowledge of Electroencephalography (EEG), respiration monitoring, and Electrocardiography (ECG) [13]. Sleep monitoring[13] was traditionally accomplished by recording polygraphs using ink pens that produced tracings on paper. An ordinary paper speed for sleep recording is 10 mm/s, with a 30 cm page corresponding to 30 seconds. Each period represented by a page is called an epoch. Most sleep recordings are digital these days, but the convention of scoring sleep in 30-second epochs or windows is still the standard. PSG is a recognized technique for patients' sleep analysis. However, it requires patients to go to hospitals or special wards because they need to wear relevant sensors to record signals, increasing medical expenses.

In response to this challenge, there has been a marked increase in research focused on the development of automated systems for detecting sleep disorder breathing (SDB). Research in this area has predominantly concentrated on enhancing detection capabilities through fewer signal channels or the choice of portable devices. Specifically, methods that utilize peripheral haemoglobin oxygen saturation (SPO2), ECG, and acoustic signals have been extensively explored [14]. Moreover, these detection techniques have been shown to have higher accuracy than those relying on acoustic signals. Given that SPO2 signal measurements can be readily obtained through nighttime pulse oximeters, frameworks that leverage these metrics are particularly advantageous for facilitating sleep health monitoring at home [15]. In addition, it is also considered that combining multiple signals for detection may obtain better detection results. Given the capability of pulse oximeters to concurrently measure multiple parameters, including pulse and blood oxygen levels, combining these data from the same sensor sources to detect sleep apnoea is feasible. Figure 1.1 compares the sleep apnoea diagnostic framework proposed in this study and the conventional diagnostic model. In traditional practice, clinicians manually identify pathological signals from PSG data. However, this process can be streamlined by employing machine learning techniques for direct diagnosis. In addition, more advanced technologies (such as the Apple Band) are used to monitor the heartbeat. This means that patients can collect data at home and the collection method is more convenient.

In sleep apnoea diagnosis, the Apnoea-Hypopnoea Index (AHI) quantifies the severity of the condition. The AHI represents the average number of apnoea and hypopnoea occurring per hour of sleep [16]. According to the guidelines of the American Academy of Sleep Medicine (AASM) [17], sleep apnoea is considered mild if $5 \leq \text{AHI} < 15$, moderate if $15 \leq \text{AHI} \leq 30$, and severe if $\text{AHI} \geq 30$ [18]. While AHI remains the gold standard, other indices such as the Oxygen Desaturation

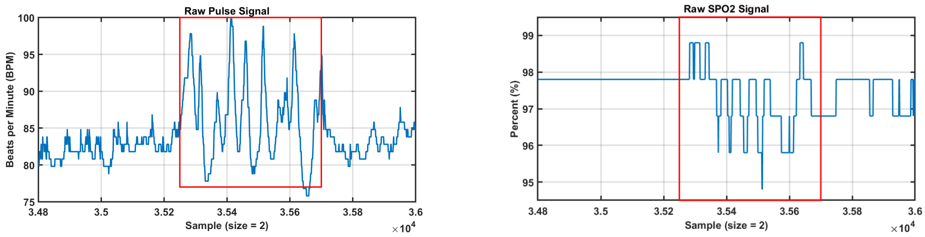

EEG Sensors
Airflow Sensors
ECG Sensors
Pulse Sensors
Leg Movement Sensors
Patient with several sensors attached at hospital
Overnight PSG example
Analysed by clinicians
Cloud
Alert
Pulse oximeter
Raw Pulse signal
Raw SPO2 signal
BMP
Index number
Wi-Fi
d4 detail coefficients(1.25Hz-0.625Hz)
d5 detail coefficients(0.625Hz-0.3125Hz)
Amplitude
Apnoea Model
apnoea
Pulse
SPO2
Patient at home with pulse oximeter
Real-time pulse and oximetry signal
Smart phone to store the data and upload to cloud
Wavelet decomposition and feature extraction
Desicion making

(a) Conventional diagnosis method: polysomnography based apnoea-hypopnoea detection diagram.

(b) Future frameworks: automatic sleep apnoea-hypopnoea event detection diagram

Fig. 1.1 Comparison of conventional and future frameworks for detecting sleep apnoea-hypopnoea

Index (ODI) [19], arousal index, and measures of sleep fragmentation have also been investigated as complementary markers of disease severity [20]. In this study, data from adults with AHI≥5 are considered as “apnoea”, otherwise, they are defined as “normal”. For children, however, the primary manifestation of apnoea is behavioural characteristics [9], so the threshold of AHI should be adjusted to a value appropriate for observing children. In Master Manual of Procedures from Childhood Adenotonsillectomy Trial (CHAT) [2], they used AHI>2 as an indicator of apnoea. The technology for automatically detecting sleep apnoea based on biological signals is developing rapidly, but there are still challenges in the task of automatic real-time detection. First, medical signal data is often highly unbalanced, which means the number of diseased segments is significantly less than the number of normal segments. This phenomenon will continue to be amplified in paediatric patients. This imbalance may cause the training results



(a) Raw pulse signal which has apnoea (b) Raw oxygen signal which has apnoea

Fig. 1.2 Examples of apnoea and normal signal

to be biased towards normal segments, thereby reducing the model's sensitivity to diseased states. Second, PSG recording is time-consuming and laborious, and patients' sleeping environment can also cause errors in their test results. Third, there are endless devices for recording human biological signals, and researchers hope to use simple equipment to achieve more accurate detection.

For sleep apnoea detection, most detection rules are based on the AASM manual [17]. This manual provides several ways to evaluate sleep signals, especially sleep stage classification and sleep apnoea analysis. Most sleep research revolves around the detection of a single signal.

This research mainly focuses on oxygen saturation data and pulse data. Some differences between normal and abnormal sleep signals could be found through data visualisation and comparison with normal sleep signals. Figure 1.2 shows the differences in oxygen and pulse data when sleep apnoea happened. From Figure 1.2, the red box circles the moment when the apnoea occurs. It can be seen that both the pulse and oxygen data fluctuate violently, which shows that the variance of the signal is constantly changing. This is also the main signal feature, mean and variance, extracted in this experiment.

1.2 Aims and Objectives

The primary aim of this thesis is to develop a practical approach for automatically detecting sleep apnoea using pulse and oximetry data. This study applies advanced machine learning algorithms and feature extraction approaches to enhance the detection accuracy. In addition, this framework ensures the feasibility of home monitoring because pulse and blood oxygen signals can be obtained using an oximeter. The specific objectives of this research include:

- Develop a home monitoring system that minimises dependence on clinical sleep analysis. Based on the blood oxygen saturation and pulse data, this system uses a machine learning framework to detect sleep apnoea, enhancing its accessibility and cost-effectiveness.
- Design, implement and evaluate an adaptive CUSUM algorithm for detecting anomalies in pulse and blood oximetry data and refine the detection threshold to enhance accuracy.
- Explore a wavelet transform-based feature extraction to detect sleep apnoea.
- Apply a RUSBoost algorithm to enhance the classification performance in imbalanced datasets, especially in paediatric sleep apnoea cases.
- Evaluate the performance of different machine learning algorithms, including Support Vector Machines (SVM), k-nearest neighbour (KNN) approach, Dirichlet process mixture model (DPMM), RUSBoost and neural networks, on balanced and unbalanced datasets to identify the most effective classification model.
- Test the proposed approaches on real-world datasets to ensure their robustness and generalizability.

1.3 Contributions and Outline of Thesis

The dissertation comprises six chapters. A brief overview of the content in each chapter is presented below.

Chapter 2 is an overview of the concepts and algorithms related to the classification of sleep apnoea. It explores current research on methods for detecting sleep apnoea, covering both traditional approaches using PSG and automated techniques which use physiological signals. This work addresses signal processing techniques, such as wavelet transforms and changing detection methods, while further exploring various machine learning and deep learning algorithms for classification purposes. This chapter presents the background of algorithmic knowledge and detection related to the proposed work in this thesis. The introduction of sleep apnoea detection algorithms can be categorised into two main types: (i) methods using traditional machine learning and feature engineering and (ii) methods using processed raw signals and deep learning algorithms. The approaches of both sets are reviewed and discussed, with detailed introductions to several widely used schemes.

Chapter 3 evaluates and compares CUSUM and adaptive CUSUM. It explained the classical CUSUM method and its limitations. Furthermore, it introduces the use of log-likelihood ratios to improve detection sensitivity for non-stationary time series. It details the implementation process and experimental setup and compares the results of adaptive CUSUM and traditional CUSUM for apnoea detection. The main contributions of this work are as follows.

- An adaptive CUSUM algorithm for detecting apnoea events is introduced.
- A reasonable subgroup process is proposed for threshold adjustment to improve detection accuracy.
- Adaptive CUSUM is confirmed to be more effective than the traditional CUSUM method on NSRR data [21]. Combining the two data results has higher accuracy.

Chapter 4 explores machine learning techniques for sleep apnoea classification, including the Dirichlet Process Gaussian Mixture Model (DPGMM) and the Rusboost algorithm based on imbalanced datasets. It explains the preprocessing steps, feature extraction, and classification methods used. A comparative evaluation of different machine learning models, including SVM and KNN, is performed on the ECG-Apnoea [1], CHAT [2] and StVincent [3] data. The results of feature fusion are also compared in this chapter. The main contributions of this work are as follows:

- A low-cost approach using blood oxygen and pulse data to detect sleep apnoea is introduced, allowing patients to monitor sleep signals at home conveniently.
- A classification framework is implemented using Dirichlet Process Gaussian Mixture Model (DPGMM).
- The RUSBoost algorithm is introduced to improve the prediction bias problem caused by imbalanced datasets in sleep apnoea detection.
- Compared different machine learning methods for apnoea detection, including SVM, KNN, etc.

Chapter 5 centres on deep learning models for sleep apnoea detection, particularly Convolutional Neural Networks (CNNs). It explores the design and implementation of CNN-based classifiers and their integration with SVM and RNN models. The main contributions of this chapter are as follows:

- A deep learning approach is proposed for detecting sleep apnoea using pulse and SPO2 signals. Three deep learning methods are adopted and compared: CNN, a CNN-SVM model, and a CNN-RNN model. The RNN block includes GRU and BiGRU structures.
- The effects of different signal fusion strategies, dropout probability settings, and window overlap lengths on detection performance are evaluated. Specifically, two fusion strategies are compared: signal stage fusion and feature stage fusion.
- The performance of the proposed approach is evaluated on a public dataset, St.Vincent University Hospital. The validation results show that combining the pulse and SPO2 signals using the proposed CNN-GRU architecture outperforms the single signal model, yielding satisfactory performance.

Chapter 6 summarises the key findings and contributions of the research. It discusses the implications of the proposed framework for home-based and clinical sleep apnoea detection. Finally, it outlines potential future research directions, including real-time implementation, further dataset validation, and the integration of wearable devices for continuous monitoring.

1.4 Associated Publications

The main results from this research are disseminated in one journal paper and two conference papers:

[J1] D. Yang, J. Zhang, H. Elphick, E. Bhargava, L. Mihaylovaa, Machine Learning Methods for Sleep Apnoea Detection Based on Imbalanced Pulse and Oximetry Data, *Journal of Machine Learning in Fundamental Sciences*, 2025(1).

[C1] D. Yang, E. Bhargava, H. Elphick and L. S. Mihaylova, "An Adaptive CUSUM Approach for Automating Sleep Apnoea Analysis Based on Pulse and Oximetry Data," In *Proceedings of the 2023 IEEE International Conference on Mechatronics and Automation (ICMA)*, Harbin, Heilongjiang, China, 2023, pp. 557-562

[C2] D. Yang, J. Zhang, H. Elphik, E. Bhargava, S. Dogramadzi, L. Mihaylova, Deep Learning Methods for Apnoea Detection Based on Pulse and Oximetry Data, In "*Proceedings of the 11th IEEE International Conference on Data Science and Systems*, Exeter, UK, 2025, Institute of Electrical and Electronics Engineers (IEEE), 2025.

Chapter 2

Literature Review

2.1 Evaluation Metrics

In binary classification, the confusion matrix categorizes outcomes into four types [22]: true positives (TP), which are instances accurately identified as positive; false positives (FP), which denote negative instances mistakenly labelled as positive; true negatives (TN), which represent negative instances correctly classified as such; and false negatives (FN), which pertain to positive instances incorrectly classified as negative.

In the comparative analysis, accuracy (ACC), sensitivity (SEN), specificity (SPE), F1 score, and Cohen's kappa were utilized. The area under the Receiver Operating Characteristic (ROC) curve, which is the area under curve (AUC), is also used. These evaluation indicators are defined as follows:

$$ACC = \frac{TP + TN}{TP + FN + TN + FP}, \tag{2.1}$$

$$SEN = \frac{TP}{TP + FN}, \tag{2.2}$$

$$SPE = \frac{TN}{TN + FP}, \tag{2.3}$$

Equation (2.1)-(2.3) calculates accuracy, sensitivity, and specificity, which are critical metrics for assessing classifier performance. However, these metrics can be misleading in cases where there are significant class imbalances.

The F1 score [23], a more robust metric, is defined as a weighted average of precision and recall, shown in equation (2.4).

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}, \tag{2.4}$$

$$precision = \frac{TP}{TP + FP} \tag{2.5}$$

Because sensitivity and recall are mathematically equivalent, the F1 score in this paper utilizes the term "recall" to discuss the classifier's ability to correctly identify actual positive cases, which is apnoea, as well as actual negative cases, which is normal. This distinction is crucial for the F1 score calculations that follow, where recall for "apnoea" equates to sensitivity, and recall for "normal" assesses the correct identification of non-apnoea cases.

Cohen's kappa [24] is a statistical coefficient designed to quantify the level of agreement between observers. It represents the observed agreement adjusted for the agreement that would be expected by chance:

$$Kappa = \frac{p_o - p_e}{1 - p_e}, \tag{2.6}$$

$$p_o = ACC, \tag{2.7}$$

$$p_e = \sum_i p_{i+} * p_{+i}, \tag{2.8}$$

where p_{i+} and p_{+i} represent the i^{th} row probabilities and i^{th} column probabilities.

AUC, or Area under the curve [22], is ideal for assessing binary classification models on imbalanced datasets. It measures the area beneath the ROC curve, which plots the true positive rate against the false positive rate at various thresholds. AUC evaluates performance without bias from class distribution, making it crucial for datasets where one class predominates.

2.2 Medical Relevance of SPO2 and Pulse Signals in Sleep Apnoea Detection

Sleep apnoea is a common breathing disorder characterised by repeated interruptions of breathing during sleep. These events can cause a significant drop in blood oxygen saturation (SPO2) and changes in heart rate, making SPO2 and pulse signals clinically important indicators for its detection. These physiological signals are typically obtained through pulse oximetry, and their non-invasive nature makes them a crucial component of clinical polysomnography and home monitoring systems. The pulse oximeter is widely recognised as an effective and clinically validated instrument for measuring peripheral SPO2 and pulse rate. In the context of sleep apnoea, recurrent apnoea and hypopnea episodes result in periodic hypoxaemia, which can be easily observed in the SPO2 signal. Both diagnostic and screening contexts have employed these desaturation episodes, which correlate with the severity of apnoea [25]. Photoplethysmography (PPG) [26], the primary method used for sensing, also detects changes in pulse shape and heart rate, which are affected by the sympathetic nervous system's response during apnoea events.

Although both SPO2 and pulse can be used independently for apnoea detection, they have disadvantages. For example, single SPO2 signal may not adequately capture the autonomic effects [1] of apnoea [27], especially in events where desaturation are limited or delayed. In addition, unrelated physiological factors, such as changes in sleep stage or motion artifacts, may impact pulse-derived parameters such as heart rate variability (HRV) [28]. Therefore, the models based on a single signal may have poor diagnostic accuracy.

Combining these two signals provides a multifaceted view of the physiological effects of sleep apnoea. PPG pulse measurements provide information about cardiovascular and autonomic disorders, while SPO2 indicates impaired respiratory function through decreased oxygen saturation. By integrating multiple signals,

[1] Autonomic effects of apnoea include sympathetic activation and cardiovascular responses, which may occur even without obvious desaturation

machine learning algorithms can detect patterns across multiple data sets that may not be clear or adequately expressed in a single signal. In wearable and home monitoring settings, this combination is very beneficial when diagnostic resources are limited, but physiological complexity is still considerable [29].

Recent studies have investigated combining ECG [30], airflow [31], and respiratory effort [32] to improve automated apnoea detection systems. While some of these approaches require multiple sensors and clinical settings, the combination of SPO2 and pulse signals offers a balance between diagnostic richness and hardware simplicity. In this study, SPO2 and pulse signals were not only analysed as separate input streams but were also concatenated into a joint feature representation. This design aims to use the extra information from both types of signals and to see if combining them helps better identify sleep apnoea.

This design is consistent with the observation that sleep apnoea impacts more than one bodily system, especially the respiratory and cardiovascular systems. It also reflects a growing trend in clinical research, where combining signals is used to make models more reliable and easier to apply to different types of data. However, effective pre-processing is important to the success of this type of signal fusion. The model will learn false patterns or underperform if the signals do not match correctly in time or there is too much noise. Using real-world data collected outside of controlled settings makes these challenges much more important. This study compared the fusion of signals under varying overlap conditions.

Combining SPO2 and pulse signals provides a reliable and useful way for detecting sleep apnoea. It advances the primary objective of this study, which is to provide an affordable, non-invasive, and expandable method for early screening.

2.3 Related Work on Sleep Apnoea Detection

In the long history of sleep research, detecting sleep apnoea has remained an ongoing challenge. Fabio Mendonca and his colleagues published a review on sleep apnoea detection [14], evaluating the performance of different algorithms and methods using various sensors, such as pulse oximeters, ECG, respiratory signals, sound, and combined approaches. This work also reviews existing algorithms whose performance has been validated experimentally, even if they have not yet been implemented in hardware, to predict trends in obstructive sleep apnoea detection. The authors provide valuable insights for researchers interested in hardware implementations of signal-processing algorithms. The paper also discusses using acoustic signals to detect apnoea events, enhancing the understanding of software implementations. However, a limitation of this study is that it predominantly focuses on single-signal detection methods. As the field evolves, there is a growing trend toward combining multiple signals for more accurate and robust apnoea detection.

The following section presents some studies on sleep apnoea detection based on the work of Mendonca et al. [14] and is divided into two parts according to the type of signal used.

2.3.1 Detection Based on ECG and EEG signal

Since brain activity is highly correlated with sleep apnoea, EEG and ECG are always considered signals for investigation, even though their acquisition is complex. In 2017, Gokhan Memis and Mustafa Sert presented a multimodal approach for the OSA classification task [33]. They extracted features from ECG and SPO_2 signals, combined them with appropriate fusion methods, and fed them to learners. The researchers use Support Vector Machine (SVM) and k-nearest neighbours (kNN) classifiers to demonstrate the effectiveness of their proposed method by considering different test scenarios. Moreover, in all scenarios, the average accuracy of the SVM method is 96.64%, which is the best classification method among the proposed multimodal methods with feature-level fusion. Although the author provides good training ideas, he does not describe the details of the selected features. Therefore, despite the impressive performance, it remains possible that the results are attributable to the selected features being particularly well-suited to the specific dataset. In addition, they did not discuss the sensitivity of their model, which means that the robustness and generalizability of the proposed approach cannot be fully assessed. Therefore, we adopted their idea of feature fusion, but added more evaluation criteria to test the robustness of the model. Zhao et al. [34] introduced a method for classifying OSA and CSA by analyzing the properties of sample entropy and variance within two sub-bands of electroencephalogram (EEG) signals. The classification was performed using several machine learning techniques, including SVM, random forests (RF), and kNN, and reached high accuracy, which is impressive.

Different from the articles introduced above, the classification of EEG signals, published in 2019 [35], uses different frequencies to classify. Using frequency to study the signal can extract features more intuitively, which is very helpful for classifying sleep stages. This work employs energy, entropy, and variance computed for each frequency band obtained from the decomposed EEG signal. Like this author, Wu Huang [36] proposed using frequency as a classification method. However, instead of extracting statistical measures from decomposed bands, Huang treated frequency analysis as a pre-processing step by applying multi-channel signal superposition in the frequency domain and derived 12 relevant features. These features were subsequently used to train an SVM classifier, achieving a classification accuracy of 98.28%. In 2011, Varun Chandola published an article introducing a Gaussian Process Based Online Change Detection Algorithm [37]. He gave a detailed algorithm of Gaussian processing (GP) and added further calculations of Gaussian parameters. The new algorithm, called Toeplitz-SolveInc, is more efficient, computationally faster, and can have more cloud memory than the old one. To prove that his algorithm is more efficient, he also gave a table [37] (Table 2.1) to show other time series algorithms performance, such as Seasonal Autoregressive and Integrated Moving Average (SARIMA), Recursive Merging (RM), Cumulative Sum method (CUSUM), Log-likelihood Ratio Test (LRT) and Bayesian Online Change Detection (BOCD). In this table, Synthetic (SYNC1, SYNC2, SYNC3) datasets is an artificially constructed time series used to test the performance of the algorithm at known change points. Normalized Difference Vegetation Index (NDVI1, NDVI12,NDVI13) time series is satellite remote sensing data used to monitor vegetation changes.

Table 2.1 Relative Performance of Different Change Detection Algorithms.

	GPC	SARM	RM	CSUM	LRT	BOCD
SYNC1	√	√	√	×	√	×
SYNC2	√	×	√	×	√	×
SYNC3	√	√	√	×	×	×
ECG1	√	N A	√	√	×	√
ECG2	√	N A	√	√	×	√
NVDI1	√	×	×	×	×	×
NVDI2	√	√	√	×	√	×
NVDI3	√	√	√	×	×	×
NVDI4	√	√	√	√	√	×

Table 2.1 gives references on which method is suitable for ECG signal. Compared with the automatic calculation of parameters above, the article [38] published in 2016 gives a new application idea of Gaussian parameters. This paper presents the recently proposed modelling of normal inverse Gaussian (NIG) probability density function (pdf) modelling in the adjustable tunable-Q factor wavelet transform (TQWT) domain for computer-aided sleep apnoea diagnosis from single-lead ECG signals. The researchers calculated the corresponding NIG parameters based on the subbands of each ECG signal segment decomposed by TQWT, which are used as features in the proposed apnoea detection algorithm. The advantage of this algorithm is that the characteristic parameters are intuitively given. However, as Varun said, the calculation and parameter selection increase the workload. Despite this, the performance of this algorithm is also superior after all the parameters are explicit.

2.3.2 Detection Based on Oximetry and Pulse Signal

Although the ECG and EEG signals are suitable for detection, collecting EEG and ECG data is quite complicated. In comparison, the blood oxygen and pulse signals selected in this experiment are relatively more straightforward to obtain, which is the same as what Chutinan et al. [39] used. He employed sleep sounds, SPO_2, and pulse rate to detect sleep apnoea. A notable innovation introduced in their methodology was the fusion of SPO_2 and pulse rate data into a singular signal for the training phase, which resulted in an accuracy of 79%. While this accuracy is lower in comparison to previous studies on apnoea detection, it is important to note that the dataset utilized was characterized by an imbalance, with only 10% positive cases. This imbalance significantly influenced the training process, skewing the results towards negative predictions. The studies mentioned earlier all focused on analyzing the characteristics of signals based on time information. However, information about frequency, as recommended by the AASM [17], is very useful for detection. Using this frequency information well can help improve the model. Verónica [29] uses the wavelet transform to analyze an overnight airflow signal and get 90.99% accuracy on OSA detection. He suggested using the order of the appropriate frequency band for feature selection, which is a good way to reduce information loss. In addition, he compared the accuracy of different classification models and found that AdaBoost.M2 performed best among other classifiers, which inspired this article.

Li et al. [40] proposes a new framework based on a clustering method, the DPGMM, which is introduced in this paper. She used the wavelet transform to generate features from SPO_2 in the frequency domain. In her research, two datasets are used and compared. One dataset achieves an accuracy of 97.01%, making it a very effective classifier. However, the performance with the other dataset is slightly inferior under this model. The difference between the two datasets stems from the balance of the data, which is a common issue in medical testing. This is also the issue that is discussed in this experiment. Regarding training with imbalanced data, Ahnaf [41] gives a RUSBoost application on sleep apnoea identification that is similar to this paper. He uses ECG data and its spectral data to take features. The accuracy of his experiments is 85.37%, which is the highest in his experiment. In his further research [30], he added the TQWT technology, which is very useful in decomposing ECG signals and increased the accuracy to 88.75%.

In a recent publication, Manish Sharma [32] and colleagues explored multi-signal detection of sleep apnoea, comparing the effects of balanced and unbalanced data using the RUSboost algorithm. They achieved an accuracy of 89.39% in the optimal data combination and reported an area of curve (AUC) of 0.905. However, the study did not compare the performance across different algorithms with the same dataset, limiting a deeper understanding of how data imbalance affects detection efficacy. Additionally, while the researchers balanced the data in their experiments, they did not specify the method used. If the data were merely down-sampled, this could result in the loss of critical information from the balanced data, potentially skewing the accuracy of the results. Daniel [42] lists several simple feature fusion methods for the multi-signal feature fusion technique. And how to use it is also listed in detail. This is very helpful for the feature fusion part later in this article.

2.4 Change Detection within Non-Stationary Time Series

Sleep apnoea can also be regarded as an abnormal signal or change point detection to determine whether a disease exists. Therefore, in addition to using different signal features, an endless stream of research focuses on the time series signal itself. Change point detection (CPD) is helpful in time series modelling and forecasting [43] and in applications such as medical condition monitoring, climatic change detection, voice and image analysis, and human activity analysis. Many of the proposed approaches for identifying change points in time series are enumerated, classified, and compared by Aminikhanghahi and Cook [43]. According to this survey, the window size influences practically all change detection techniques. Combining various window widths can be an effective way to use the best window length for each subsequence.

The sleep signal is instantaneously observed, and real-time change detection can be a future development trend. Thomas [44] proposed a more suitable method for changing detection: CUSUM and kernel-based methods for online detection. Although this algorithm is widely used, it primarily involves comparing detection change scores with a threshold to determine if a change has occurred, making selecting an optimal threshold challenging. Furthermore, these methods are typically designed for stationary sequences, whereas sleep signals are predominantly non-stationary. One of the ongoing challenges in CPD is effectively handling non-stationary time series. This section primarily introduces various applications for CPD and discusses the CUSUM algorithm, including its role in sleep apnoea detection.

2.4.1 Related work

The CPD issues consist of spotting abrupt changes in data when a feature of the time series changes [45]. Sometimes used interchangeably are similar ideas: segmentation, edge detection, event detection, anomaly detection, and changing point detection. CPD is closely related to the currently accepted problem of change point estimate [46]. Change point estimations, however, mostly define the kind and extent of the found change. Data mining, statistics, and computer science have all seen CPD investigated during the past few decades. Based on their 2017 survey, Aminikhanghahi and Cook [43] classified several main uses for CPD shown below. The methods involved in these purposes are also frequently used in apnoea.

The first is medical condition monitoring. The identification of patterns in physiological indicators like heart rate, electroencephalogram (EEG), and electrocardiogram (ECG) makes automated, real-time monitoring of patient health possible.

For lesion changes in multiple sclerosis (MS) patients, Bosc et al. [47] developed an automated multimodal serial MRI method. They combine brain extraction, picture registration, intensity normalisation, and hypothesis testing using the generalised likelihood ratio test (GLRT). The approach discovers little variations in a large population with relapsing-remitting multiple sclerosis better than by hand identification. Automated medical imaging technologies such as this accelerate diagnosis by processing large volumes of data far more rapidly than human assessment, while maintaining consistent accuracy and reducing inter-observer variability [48][49]. These properties illustrate the speed and precision with which automation can support clinical decision-making in neuroimaging.

Apart from medical visuals, time series is another approach that is often utilised to convey medical information. Especially for ECG recordings, Bayesian change point detection has helped improve time-series data processing, especially for ECG recordings. Conventional methods, which rely on statistical tests or predefined thresholds, are computationally expensive and find it difficult to detect minute changes in brain activity. Malladi et al. [50] show that Bayesian models and linear complexity techniques are fit for real-time segmenting of epileptic activity. Furthermore, their method can generate approximations, which reduces the quadratic complexity of these methods and raises their efficiency.

Furthermore, CPD is needed in sleep analysis. Progressive Detrended Fluctuation Analysis (PDFA) was introduced by Staudacher and associates for online HRV altering point identification during sleep [51]. The method follows real-time HRV transitions, sympathetic activity, and brief moments of wake during sleep phase changes. Since PDFA can be separated between rapid eye movement (REM) and non-rapid eye movement (NREM) sleep periods, it is a possible technique for examining sleep patterns. It can manage PSG data for the whole night, making it suitable for clinical and research sleep monitoring. Developing a real-time system to identify significant heart rate fluctuations in anaesthetised youngsters, Ansermino et al. [52] Using an adaptive Kalman filter, the method models heart rate signals and aggregates Exponential Weighted Moving Average (EWMA) predictions with CUSUM testing to identify clinically significant changes. This technique enables doctors of anaesthesia to track and react to physiological changes during paediatric surgeries. Furthermore, the implementation of CUSUM reveals that it may be used for change point detection, which offers concepts for the later usage of CUSUM for binary classification.

The second area for CPD is speech recognition. Speech recognition is the technique of converting spoken words or phrases into written form. However, in apnoea detection, speech recognition is obviously related to audio detection. Change point detection methods help to segment audio and separate noise from words, phrases, sentences, and silence. Compared with apnoea detection, the CPD method helps to distinguish snoring from normal breathing. In addition, in sleep classification, it is also particularly important to separate the noise pattern in the sleep signal.

Le et al. [53] used an audio dataset to detect sleep apnoea based on breathing sounds. They split the data into two (no apnoea, apnoea/hypopnoea) or three categories (no apnoea, hypopnoea, apnoea) using 1 minute per epoch. They applied weights to the deep learning network model to handle the data imbalance to achieve the best classification. The binary classification result achieved an accuracy of 88.8%. The best weights for no apnoea, apnoea, and hypopnoea were 1.0, 1.3, and 2.1, respectively. Although the results far surpassed the relevant studies at that time, the weighting was too dependent on the data. Changes in the source of the data set will affect the choice of weights. Although the results far surpassed the relevant research at that time, the weighting was too dependent on the data. Changes in the source of the data set will affect the choice of weights. However, this study still provides a strong basis for detecting apnoea in audio data.

In audio recognition, the audio spectrum is commonly used. Serrano et al. [54] proposed an audio-based technique for detecting and classifying OSA syndrome, depending on Mel-spectrograms [2] generated from ambiently recorded sleep noises. Apnoea detection and AHI estimation are performed using their method, combining a pre-trained VGGish convolutional neural network with a Bidirectional LSTM (biLSTM) network. The model demonstrated great ac-

[2] A mel-spectrogram is a sound representation that shows how energy is distributed over time and frequency, with the frequency axis adjusted to match human hearing sensitivity.

curacy, recall, and F1-scores based on five-fold cross-validation on a dataset of 192 patients. Particularly, low-cost, non-invasive equipment like smartphone microphones suggests the potential for at-home applications. The capacity to generalise to various populations and real-world noise conditions, however, like with other deep learning models, stays to be verified. Especially in environments where traditional polysomnography is difficult or unavailable, the study contributes to the expanding field of work regarding audio-based replacements to classic sleep study methods.

Two other similar areas of change detection are based on medical image analysis and video analysis. Images find use in many fields, from remote sensing to surveillance to medical diagnosis and treatment to civil infrastructure to underwater sensing [55]. Finding diverse points of view in several photos of the same scene taken at different times has drawn a lot of interest. Acknowledging unanticipated events like abnormal sleep imaging could be seen as a change point problem.

Jagadeesh P. et al. [56] presents a detection method for sleep apnoea syndrome, termed the Sleep Apnoea Analysing Strategy (SAAS), which integrates image preprocessing, feature extraction, and neural network classification to enhance diagnostic accuracy and efficiency. In comparison to the conventional image analysis model, SAAS has demonstrated a superior prediction accuracy of 96.84% during numerous training cycles, significantly reducing both the false positive rate and the false negative rate. This method relies on recognising essential anatomical features in medical imaging techniques like X-rays, CT scans, or MRIs, which provide an early identification and classification of sleep apnoea without an expensive and complex PSG. This study represents a more pragmatic and proactive way to enhance intelligent medical systems.

Chiu et al. [57] presented a video-based, contactless approach for detecting sleep apnoea that uses a single RGB camera to capture respiratory movements and physiological signs throughout nocturnal sleep. Their methodology includes optical flow from the nose and throat areas to evaluate respiratory effort and incorporates remote photoplethysmography (rPPG) to extract HRV features. Two deep learning models are employed: one for segment-level classification of apnoea events and another for estimating AHI severity. The system achieved an accuracy of 78.45% in per-segment detection and 76.47% in per-recording AHI classification, based on a dataset of 61 participants annotated via standard PSG protocols. In comparison to contact-based ECG devices, the results are approximately equivalent. However, the approach is significantly more user-friendly as it is non-invasive and inconspicuous. The limited dataset and its reliance on infrared lighting conditions could affect its applicability to other situations. This study demonstrates the increasing use of camera-based health monitoring and the application of deep learning to extract diagnostically valuable information through non-contact approaches.

Video analysis is normally related to human activity analysis. By means of aspects of observed sensor data from mobile devices or smart homes, changing point identification here detects activity breakpoints or transitions [43]. These change points will help activity-aware services, segmenting activities, interact with individuals with minimum disturbance, and spot behavioural changes exposing information on health issues.

Using data from sensors in wearable devices and cellphones, human activity recognition [58] groups daily activities, therefore enabling applications including elder care, rehabilitation, and wellness management. Conventional Machine Learning approaches depend on human-generated traits, so they often find challenges in complex activity detection. By automating feature extraction, improving accuracy, and controlling temporal dynamics in data, Deep Learning methods [59] solve these restrictions.

2.4.2 Applications of CUSUM Algorithm

The CUSUM control chart is a widely used statistical process control tool for detecting small, sustained changes in process parameters, initially developed for quality control in industrial contexts. Recent advances have seen the application of CUSUM methods in various domains, including public health surveillance, signal change detection, and ecological monitoring. This subsection summarizes the key developments in CUSUM, particularly adaptive versions.

CUSUM method on real-time detection

Mesnil and Petitgas [60] illustrated using the CUSUM method to monitor aquatic ecosystems, specifically to detect persistent changes in fish stock indicators. This study demonstrated the flexibility of the CUSUM chart for ecological monitoring, allowing for the detection of shifts in environmental indicators based on real-time survey data. This also provides the basis for real-time detection of sleep apnoea using the CUSUM method. In addition, the authors emphasized the importance of tuning CUSUM parameters, such as the reference value (k) and decision interval (h), to balance the prompt detection of significant changes and minimise false alarms. In our case, the reference value k is set to the mean of the sample data, representing the expected baseline under normal conditions. The decision interval h is defined as the window size, which determines the temporal resolution of detection. A larger window improves robustness to noise, while a smaller one increases sensitivity but may raise false alarms. This setting ensures that the CUSUM test is aligned with the time scale of the analysed respiratory segments.

Furthermore, Mesnil and Petitgas' work highlighted the challenges of setting appropriate thresholds for CUSUM charts in natural resource monitoring, where data variability can often confound change detection efforts. The threshold selection of CPD has always been an issue worthy of study, but the error it brings can be reduced by replacing CUSUM with Adaptive CUSUM.

Adaptive CUSUM for detecting changes in various signals

Alippi and Roveri [61] presented an adaptive CUSUM-based test for detecting changes in the behaviour of various signals, focusing on applications such as fault detection and climatic phenomena. Unlike traditional CUSUM, which requires fixed parameters defined at design time, the adaptive approach introduced by

Alippi and Roveri allows the test parameters to be automatically adjusted based on the evolving data characteristics. This adaptability makes the method suitable for scenarios where the parameters of the underlying signal are not known a priori, or when they evolve, such as a sleep signal. The authors showed that this adaptive mechanism enhances both the timeliness and effectiveness of change detection compared to traditional methods, especially in cases where abrupt changes are mixed with smoother, long-term drifts.

Adaptive CUSUM for time series data

Vanli and Giroux [62] proposed an adaptive CUSUM method to monitor integer-valued time-series data, which is especially useful for applications like public health surveillance where count data often exhibit seasonality and autocorrelation. Traditional CUSUM methods are limited in requiring the specification of a particular mean shift size, which can lead to suboptimal detection when the actual shift differs from this design value. The adaptive CUSUM introduced by Vanli and Giroux addresses this issue by using an EWMA to estimate both level and trend changes, allowing the detection mechanism to adapt to varying shift magnitudes. Their work also incorporated an Integer-valued Generalized Autoregressive Conditional Heteroskedasticity (INGARCH) model to account for autocorrelation and seasonality, making it particularly suitable for count data like infectious disease occurrences, which often show temporal correlations

2.5 Wavelet Transform on Feature Extraction

2.5.1 Wavelet Transform

Wavelet transform (WT) has become a widely used mathematical tool for the multi-resolution decomposition of time series signals and has potential applications in the field of computer vision [63]. The pulse and oxygen signals are non-stationary and originate from a nonlinear system, and WT can identify subtle morphological changes in non-stationary signals [64]. The WT is divided into discrete wavelet transforms and continuous wavelet transforms. Compared to the continuous wavelet transform (CWT), the discrete wavelet transform (DWT) can reduce computational complexity and capture frequency and time location information more effectively than the discrete Fourier transform (DFT).

Figure 2.1 illustrates the computation of the DWT. $x[n]$ here is the pulse and SPO$_2$ signal segment. It gives a filter bank tree. Each level has a high-pass filter g[n], which is the mother wavelet, and a low-pass filter h[n], which is the mirror version of the mother wavelet. The relationship between h[n] and g[n] be described as equation (2.9). It is taken from Rioul and Vetterli [65]

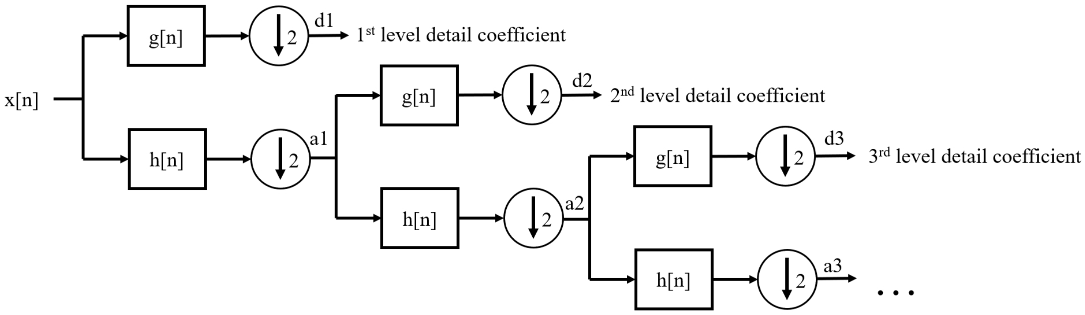

x[n]
g[n]
h[n]
2
d1
1st level detail coefficient
a1
d2
2nd level detail coefficient
a2
d3
3rd level detail coefficient
a3
...

Fig. 2.1 Decomposition process of a signal using DWT

$$g[L-1-n] = (-1)^n \cdot h[n]. \tag{2.9}$$

The coefficients from the high-pass filter are called detail coefficients, which are used to compute features. The coefficients from the low-pass filter are called approximate coefficients. These approximate coefficients will be downsampled by a factor of two and further decomposed into another set of approximate and detailed coefficients. This process will continue for $N = log_2(M)$ times, where M is the length of segment $x[n]$ [66]. The approximation coefficients a_i and the detail coefficients d_i can be calculated as follows

$$\begin{aligned} a_i[k] &= \sigma_n a_{n-1}[n] \cdot h[2k-n], \\ d_i[k] &= \sigma_n a_{n-1}[n] \cdot g[2k-n] \end{aligned} \tag{2.10}$$

where a_0 is the raw signal segment value $x[n]$, and i is the level. Since the invention of the wavelet transform, many experts have created many different types of wavelets. This experiment mainly compares Daubechies series wavelets (db) 1 to 4, among which db1 is the well-known Haar wavelet. Figure 2.2 shows the wavelet used in this research. Different wavelets apply to different signals due to their different shapes, which will be confirmed later in the performance of each chapter.

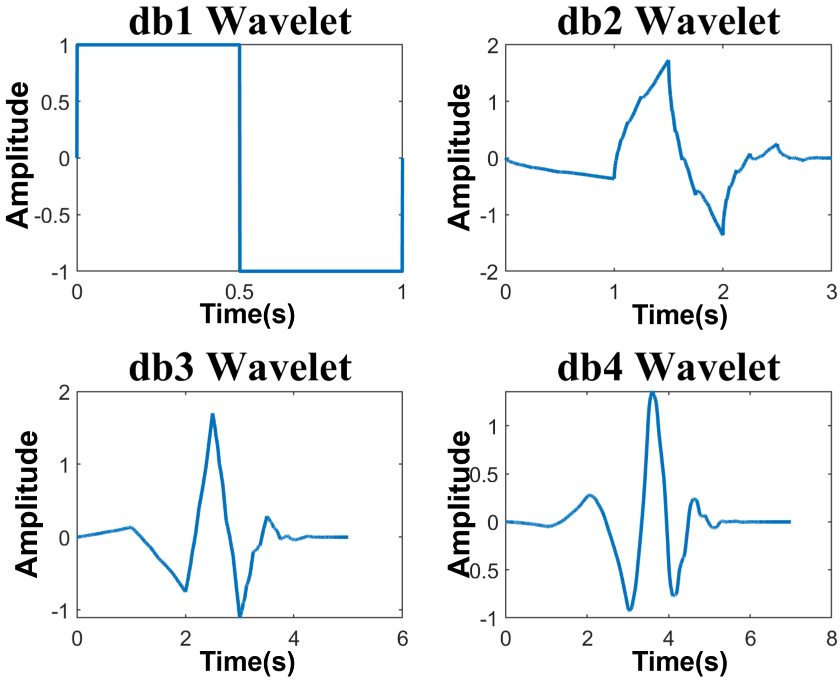


Fig. 2.2 Different wavelet plot

2.5.2 Wavelet Based Features

In this study, the signal was divided into multiple 1-minute windows. Since the sampling frequency of each signal is different, the number of values used for calculation is also different. Additionally, typical hyperpnoea (also referred to as apnoea in this study) following apnoea episodes occurs between 0.784–0.890 Hz events, which increase the respiratory rate. Children who tested positive for apnoea exhibited greater variability in the Power Spectral Density (PSD) range of 0.35–0.43 Hz [31]. Therefore, experiments will select levels that contain frequencies within this range. For instance, Figure 2.3 shows an example of SPO_2 signal whose sample rate is 10 Hz and whose level 4 and 5 coefficients are used. After obtaining the required detail coefficients, the following features are extracted to measure the information they contain:

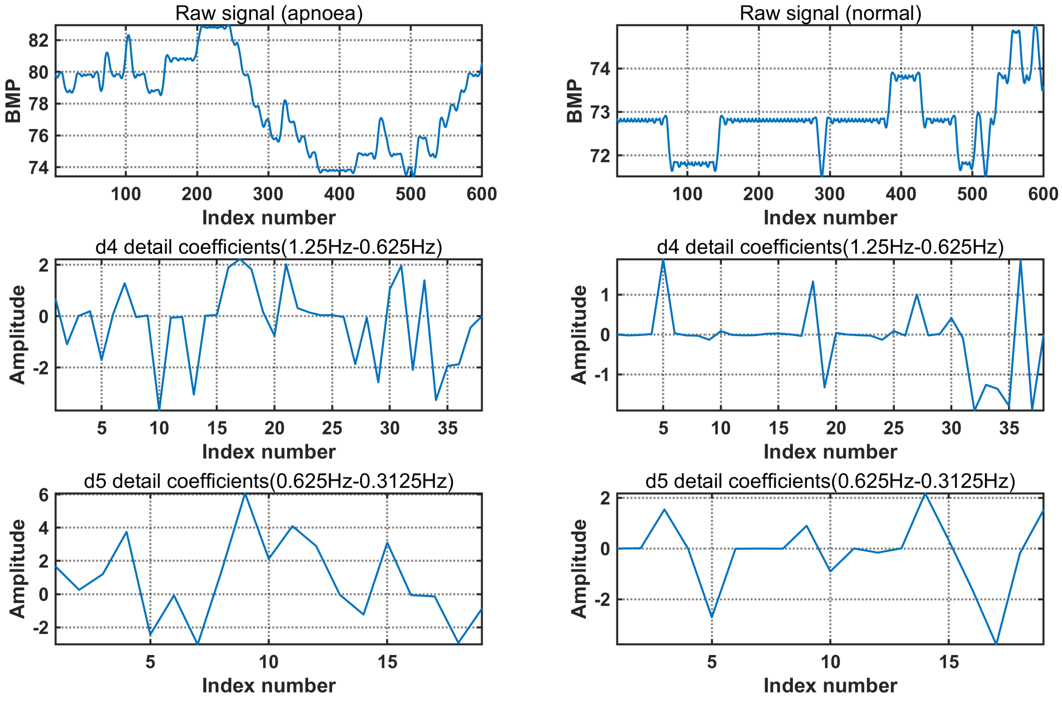


Fig. 2.3 Decomposition process of a signal using DWT

- Mean Energy: The average energy found in the wavelet detail coefficients varies between “apnoea” and ‘normal’ segments, making it a useful feature for differentiation. The average energy E of a fragment is calculated as follows:

$$E = \frac{1}{N}\Sigma_i^N d(i)^2$$

, where $d(i)$ represents the i-th element of the wavelet coefficients for a segment and N denotes the total number of coefficients.

- Statistical features are computed, including range, variance, and maximum value.

- Number of the large coefficients [40]: The count of points exceeding the threshold can serve as a feature to differentiate the "apnoea" segment from the "normal" segment in signal measurements. However, this feature is uncertain. The original work utilized this feature to achieve high accuracy, but the threshold selection is subjective. Additionally, if this feature is used, the signal needs to be resampled. To avoid this, the CUSUM [61] method is used to judge different thresholds.

2.5.3 Statistical Features Introduction

Feature extraction mainly targets signal and wavelet transform detail coefficients, which are introduced in Section 2.5.1. The specific features for signal values are as follows:

- Variance: Since the variance of apnoea subsequences is usually higher than the variance presented in normal subsequences [67], variance can be used as a classification feature for the classifier.

- Range: The range is determined by the difference between the highest and lowest signal values within a segment. Typically, SPO2 values decrease after apnoea events and remain relatively stable during normal subsequences. In addition, pulse values usually oscillate more during apnoea and less during normal subsequences. Therefore, the range can be selected as a feature to detect apnoea.

- Average: SPO2 values usually drop after an apnoea event. The average value of the SPO2 segment may change after an apnoea. By observing the changes in the pulse signal, it can be found that the pulse signal has evident oscillations during the apnoea event. If it occurs for a short period, it appears as a falling signal with a very high slope.

- Minimum: SPO2 values typically decrease significantly following apnoea events, as previously mentioned, which is not a prevalent occurrence in normal sleep. Consequently, evaluating the minimal SPO2 value within a segment is advantageous. Similarly, the minimum pulse value typically decreases substantially within the apnoea event segment due to the large oscillations.

- Kurtosis [68]: The peakedness of a signal can be ascertained using kurtosis. A dataset with a kurtosis more significant than zero is characterised by a higher peakedness and more extreme values. A flatter distribution is indicated by a kurtosis that is less than 0. The stability of both SPO2 and pulse signals is altered following an apnoea event. Compared to the 'normal' segment, the kurtosis value of the abnormal segment increases.
- Shannon Entropy [69]: Shannon entropy is a metric that quantifies the amount of information in a system using the probabilities of various scenarios. It quantifies the changes in energy distribution during the decomposition process, illuminating the fundamental dynamic behaviour and signal irregularities. The Shannon entropy escalates with a rise in uncertainty of an event or entity and diminishes with a drop in uncertainty.

2.5.4 Related work

Feature extraction and classification techniques are critical in signal processing for cognitive task recognition. This subsection will introduce recognition and classification applications based on wavelet transform and feature selection.

The most common application is combined with human biological signals. For example, when analyzing EEG signals, artefacts are disturbances that may occur during signal acquisition, which affect the signal analysis. Kousarrizi [70] and his colleague uses the wavelet transform alongside various learning methods to identify trials that contain artefacts. Wavelet transform techniques efficiently extract pertinent information from EEG data, tackling noise and signal artefacts. This study shows the viability of employing wavelet transform with classification methods utilising machine learning or deep learning techniques.

As comprehension of wavelet transform advances, individuals analyse the correlation between wavelet coefficients and classification tasks. Amin et al. [71] applied the DWT to sleep apnoea detection, demonstrating its effectiveness in computing relative wavelet energy from approximation and detail coefficients across multiple decomposition levels. The features were categorised using machine learning approaches, such as SVM, multi-layer perceptrons (MLP), and kNN. The method attained above 98% classification accuracy. This highlights the capability of DWT for accurate and effective EEG signal classification.

Due to the wide variety of wavelets, scientists have also begun to try to use targeted wavelets to transform signals to obtain better features. A TQWT and a statistical approach are proposed to analyse various EEG records and classification of epileptic seizures [72]. The research utilises TQWT to decompose EEG signals into subbands, facilitating the extraction of statistical features for classification through machine learning techniques. The approach effectively distinguishes brain signals, showing its promise for various EEG classification applications.

In addition to biological signals, wavelet transform has achieved significant advances in engineering. Syed and Muralidharan [73] use the DWT to diagnose issues in planetary gearboxes. DWT collects statistical information by decomposing the vibration signal into frequency bands. The information can then be gathered using machine learning methods. This study proves that DWT outperforms alternative fault classification methods, demonstrating its dependability for equipment condition monitoring and fault detection.

2.6 Machine Learning and Deep Learning

The introduction of artificial intelligence (AI) into the medical field has always been a typical application, and sleep apnoea applications are no exception. Artificial intelligence algorithms can generally be classified into machine learning and deep learning. This section mainly introduces the relevant machine learning algorithms and deep learning algorithms used in this study. Applications in the past decade are also introduced at the end.

2.6.1 Imbalanced Data

Imbalanced datasets frequently occur in practical applications and may significantly affect the classification accuracy of machine learning systems. This issue occurs when the number of instances in one class significantly differs from that of other classes, leading traditional classifiers to preferentially select the majority class while frequently ignoring the minority class [74]. This problem is common in practical applications such as fraud detection, medical diagnosis, and other applications where minority groups are less frequent but often more important.

Many approaches have been tried to solve the classification problem of imbalanced datasets. Ganganwar [75] presents an extensive review of the challenges and solutions of handling imbalanced datasets for classification tasks. A common approach to dealing with imbalanced data problems is to rebalance the data through oversampling or undersampling artificially. The representative algorithm for oversampling is Synthetic Minority Over-sampling Technique (SMOTE), and the representative algorithm for undersampling is Random under-sampling (RUS). When the data is extremely unbalanced (usually 100:1 or higher), oversampling and undersampling can be combined to achieve a better balance. In addition, the article emphasizes that effectively solving the problem of class imbalance usually requires combining data and algorithm-level techniques to optimize classification performance, such as combining RUS with Boosting. In addition, he emphasizes the importance of hybrid methods and specific environmental solutions.

2.6.2 Machine Learning Method

Machine learning is a broad field that spans information technology, statistics, probability, artificial intelligence, psychology, neurobiology, and various other disciplines [76]. Several machine learning algorithms have been developed, improved, and refined. Machine learning algorithms are categorised into the following groups according to the intended result of the algorithm:

1) Supervised learning [77] is a type of machine learning where a model is trained on labelled data, using input-output pairs to learn a mapping function that predicts outcomes for unobserved inputs.

2) Unsupervised learning [78] is a type of machine learning where a model identifies patterns or structures in unlabelled data, often used for clustering or dimensionality reduction.

3) Semi-supervised learning [79] integrates labelled and unlabelled instances to produce an appropriate function or classifier.

4) Reinforcement learning [80] is the algorithm that develops a policy for action based on its observations of the environment. Every action influences the environment, offering input to inform the learning process.

This study uses supervised machine learning. This section briefly describes the content of supervised machine learning and introduces several related basic algorithms used.

Supervised learning involves generating models that predict outcomes based on input features. Typically, the process is divided into two parts: training and testing. During the training, algorithms discover patterns from labelled datasets, which are subsequently tested on previously unobserved test data to ensure the prediction accuracy. Supervised learning tasks are often classified as either classification (discrete outcomes) or regression (continuous outcomes) [81]. Here, we mainly discuss binary classification algorithms. Figure 2.4 gives a general supervised classification architecture.

Decision trees

The decision tree algorithm is a supervised learning technique for classification tasks. It recursively divides the dataset into subsets according to feature values,

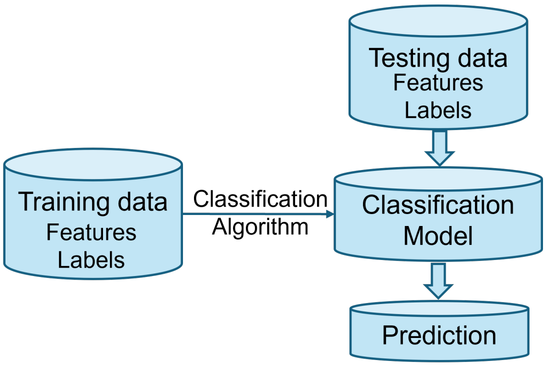
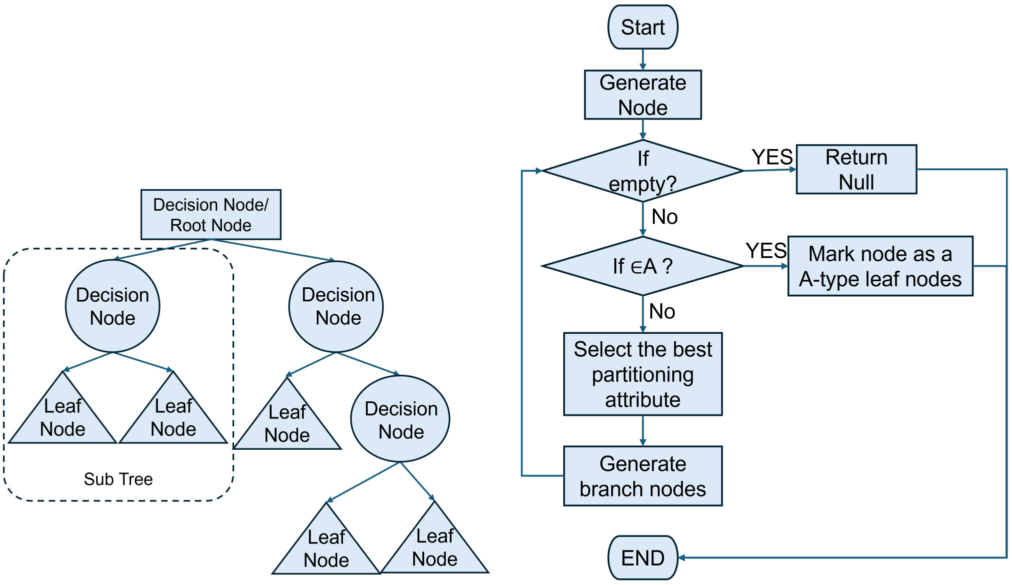


Fig. 2.4 Supervised classification architecture

forming a tree-like structure comprising decision and leaf nodes (see Figure 2.5(a)). The flow chart of the decision tree is shown in Figure 2.5(b). It can be seen from the flowchart that Selecting the best Splitting measure is the focus of the decision tree process. Below is an overview of the Splitting measures [82].



(a) Decision tree architecture

(b) Decision tree flow chart

Fig. 2.5 Decision tree architecture and flow chart

First, Assume the decision tree training sample set is D. Each sample in D belongs to one of m classes. Then, the Information entropy of D before and after splitting are:

$$H(D) = -\Sigma_{i=1}^{m} p_i log p_i, \tag{2.11}$$

where m is number of classes (labels) and $m = 2$ here because this is a binary classification. p_i is the proportion of samples in D belonging to class i. Suppose we split D based on an feature A. Let A have v possible values, which partition D into subsets $\{D_1, D_2, ...D_v\}$. Then, the conditional entropy after splitting is:

$$H(D|A) = \Sigma_{j=1}^{v} \frac{|D_j|}{|D|} H(D_j) \tag{2.12}$$

where $|D_j|$ is the number of samples in D_j. Then, the information gain is

$$Gain(D, A) = H(D) - H(D|A) \tag{2.13}$$

If A is the feature that maximizes the goodness measure, the gain ratio can be computed:

$$GainRatio(D, A) = \frac{Gain(D, A)}{IV(A)}, \tag{2.14}$$

$$IV(A) = -\Sigma_{i=1}^{v} \frac{|D_j|}{|D|} log_2 \frac{|D_j|}{|D|}, \tag{2.15}$$

Another Splitting measure rule is Gini value:

$$\begin{aligned} Gini(p) &= \Sigma_{i=1}^{n} p_n(1 - p_n), \\ &= 1 - \Sigma_{i=1}^{n} p_n^2, \qquad = 2p(1 - p), \end{aligned} \tag{2.16}$$

where p_n is the probability that the sample point belongs to n.

Linear SVM

Linear SVM aims to identify the optimal hyperplane that distinguishes data points of varying classes in a feature space while maximising the margin between the classes. Given a training dataset of n points of the form $[(x_1, y_1), ..., (x_n, y_n)]$, where x_i is the data and y_i is the label of x_i. Normally, the y_i are either 1 or -1, which explains why the SVM can do a binary classification. For these data, the purpose of SVM is to find the 'maximum margin hyperplane' [83] that divides

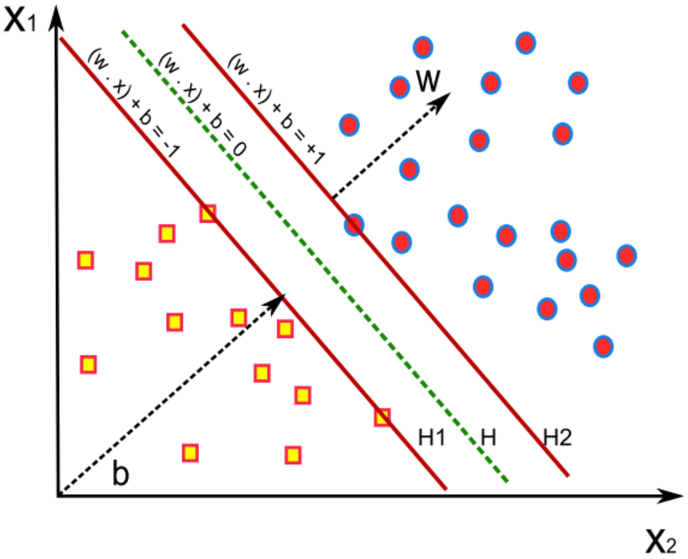


Fig. 2.6 SVM classifier: hard margin hyperplane [4]

point groups x_i, one of which is divided by $y = 1$, and the other is $y = -1$. This hyperplane maximises the distance between the hyperplane and the closest point x_i to any set. The hyperplane can be written as

$$w^T x - b = 0, \tag{2.17}$$

where w denotes the (potentially unnormalized) normal vector of the hyperplane. The hyperplane's offset from the origin along the normal vector w is defined by the parameter $\frac{b}{\|w\|}$. This equation allows for two types of margins: hard and soft. If the training data is linearly separable and devoid of errors, including outliers and noise, a hard margin is employed [83]. With a normalized or standardized dataset, these hyperplanes can be described by these equations [84]

$$\begin{aligned} & w^T x_i - b \geq 1,\ if\ y_i = 1, \\ & w^T x_i - b \leq -1,\ if\ y_i = -1, \\ & y_i(w^T x_i - b) \geq 1,\ for\ all\ 1 \leq i \leq n. \end{aligned} \tag{2.18}$$

Figure 2.6 shows the SVM classifier by hard margin hyperplane. Hard margins are based on the assumption that the data is idealised. However, obtaining an independent line to split the data in space in reality is impossible. Even a curved decision boundary or a hyperplane that can precisely separate the data won't finish the classification if there is noise in the data. Ignoring a few data points is a better method of smoothing the boundaries than circling or looping around outliers. This is what a soft margin's existence means. Here, slack variables are added to explain soft margin better. Assume a distance S_k on the incorrect side of the hyperplane that is within the permitted range and does not defy the constraints. This can be expressed as equation (2.19).

$$y_i(w^T x + b) \geq 1 - S_k. \tag{2.19}$$

In this case, Lagrangian variables can be introduced to penalize large slacks

$$min\ L = \frac{1}{2} w^T w - \sum \lambda_k (y_k (w^T x_k + b) + S_k - 1) + \alpha \sum S_k, \tag{2.20}$$

where decreasing α allows more data to be on the wrong side of the hyperplane and will be treated as outliers, providing a smoother decision boundary[84].

Naive Bayes

Bayesian classification [81] is a supervised learning method and a statistical classification method. Assumes a foundational probabilistic model and facilitates the principled representation of uncertainty regarding the model by ascertaining the probability of the outcomes. The primary objective of Bayesian classification is to address prediction challenges. This categorisation offers effective learning methods and can integrate observed data. Bayesian categorisation offers valuable insights for comprehending and assessing learning algorithms. It computes explicit probabilities for hypotheses and enhances resilience against noise in input data.

Assume two random events, A and B, with the probabilities $P(A)$ and $P(B)$ respectively. The Bayes Rule is:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2.21}$$

Converted into a classification task, A and B can be considered features and classification labels. The expression is

$$P(label|features) = \frac{P(features|label)P(label)}{P(features)} \tag{2.22}$$

For binary classification tasks, this algorithm takes out samples with label 0 and label 1, respectively, and builds two models, $model0$ and $model1$. When predicting the category of a new sample, the sample features are brought into $model0$ and $model1$, respectively. The sample belongs to the model that fits the model better. The Bayesian formula becomes:

$$P(y|(X)) = \frac{P(X|y=1)P(y=1)}{P(X)}, \tag{2.23}$$

$$P(X) = P(X|y=1)P(y=1) + P(X|y=0)P(y=0), \tag{2.24}$$

$$\tag{2.25}$$

where $X = \{x_1, x_2, ..., x_n\}$ is the features with different labels $y = \{0, 1\}$. If the x_i is independent, then

$$P(X|y=1) = \prod_{i=1}^{n} P(x_i|y=1) \tag{2.26}$$

K-nearest neighbour approaches

The kNN algorithm is a simple, non-parametric supervised learning technique. It categorises a new data point according to the dominant labels or mean values of its nearest neighbours [85].

KNN functions by calculating the distance between a specified query point and every point in the training dataset, typically employing the Euclidean distance formula:

$$d(x, y) = \sqrt{\Sigma_{n}^{i=1}(x_i - y_i)^2}, \tag{2.27}$$

where x and y are two points in an n-dimensional space. The algorithm calculates the k closest points (nearest neighbours) to the query point based on the distances. The parameter k has a significant impact on the performance. A lower k value decreases the approximation error of the model, making the prediction outcomes sensitive to adjacent instance locations. If the adjacent instance points are noisy, the forecast will be erroneous. A smaller k value indicates that the model is complicated and susceptible to over-fitting. An increased k value results in a simpler overall model, which is susceptible to under-fitting.

KNN labels a new point in classification tasks using the most popular label among its neighbours. As the dataset size increases, it has substantial computing costs since it involves distance calculations between the query instance and each instance in the training set. This trait frequently needs efficient data structures such as KD-trees or ball trees to speed up neighbour searches, particularly in massive datasets.

2.6.3 Deep Learning Methods

Before introducing deep learning methods, this dissertation introduces the concept of artificial neural networks (ANNs) [86].

This computing processing system primarily draws inspiration from the functioning of biological nervous systems, such as the human brain. ANNs comprise many interconnected computational nodes, which are grouped and learned from the input to improve the final output. As shown in Figure 2.7, the hidden layer will then make decisions from the previous layer and weigh how the random changes within it will damage or improve the final output. This process is called the learning process. Stacking multiple hidden layers together is usually called deep learning [87]. Like machine learning, deep learning can be

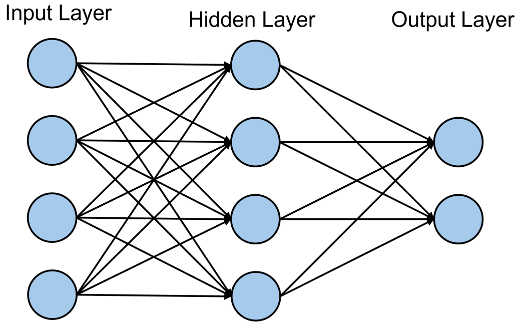


Fig. 2.7 A basis structure of a common ANN architectures

divided into supervised learning and unsupervised learning based on how the prediction results are made. More introductions of these are in Section 2.6.2. CNNs differ from conventional ANNs in their application in pattern identification inside images [87]. This enables us to incorporate image-specific characteristics into the design, optimize the network for image-centric applications and lower the necessary parameters to establish the model. However, this study uses time series data. CNN also has its limitations for time series. It can extract spatial features well but cannot preserve temporal features well. This will be discussed in more detail in Chapter 5.

Convolutional Neural Network

A CNN model is a specialised deep learning model mainly designed for image and video processing. Inspired by the human visual system, it is applied to identify characteristics and patterns from unprocessed images automatically. While building a CNN model, the choice of layer is important in the network. The following is a list of every layer of an original CNN used in this work, and the detailed setting of each layer will be explained in Chapter 5.

- **Input Layer:** The input layer is the entry point for data into the CNN. For feature extraction, it can be the original processed signal data. For classification tasks, features can be extracted in low dimensions. This layer defines the shape and type of the input data, ensuring that the network knows the dimensions and characteristics of the incoming data stream.

- **Convolutional Layer:**The convolutional layer [88] is the core building block of CNN. It applies a set of learnable filters (kernels) to the input. Each kernel performs a sliding convolution on the data in all directions. This process can extract feature maps from the data. The basic equation of convolution operation is expressed in equation 2.28.

$$\begin{aligned} Feature\ map &= input \otimes kernel \\ &= \sum_{y=0}^{columns} \left(\sum_{x=0}^{rows} input(x-a, y-b) kernel(x,y) \right), \end{aligned} \tag{2.28}$$

  where $\otimes$ denotes the convolution operation. a and b represent the offset from the centre point of the kernel (or filter) to the location of the input data being processed. x and y are the row and column indices used to iterate over the kernel matrix.

- **Batch Normalization Layer:** A Batch Normalisation Layer [89] is used to normalise feature maps. This layer ensures that the data fed into later layers has a zero mean and unit variance, allowing faster learning rates and more efficient training epochs.

- **Activation Function:** The activation Function Layer makes the model nonlinear, which is essential for learning complicated patterns. A Rectified Linear Unit (ReLU) activation function in this layer performs a threshold process that chooses which feature maps are turned on. ReLU function [90] denoted as equation (2.29).

$$ReLU(x) = \begin{cases} 0, & for\ x < 0 \\ x, & for\ x \geq 0 \end{cases} \tag{2.29}$$

- **Pooling Layer:** The Pooling Layer [88] decreases the data dimension and reduces the network's computational burden and number of parameters. This layer functions independently on each feature map to condense the features found in the sub-regions of the map. In time series analysis, max pooling is a common way to get the most important features. This makes it easier for the network to find patterns across different time scales by choosing the highest value from each sub-region.

- **Dropout Layer:** The dropout layer [91] is a type of regularisation that prevents overfitting in the network. At each update during training, this layer randomly sets a certain number of input units to zero, helping to eliminate random correlations in the data. Dropout is helpful in big networks and a good way to improve generalisation in the final model.

- **Fully Connective Layer:** The fully connected layer [88] combines the high-level features of the convolutional and pooling layers. Its output is a vector, with each entry linked to all outputs from the layer before it. This vector is used to make the final classification choice.

- **Softmax Layer:** The softmax layer [88] is generally employed as the final output layer in classification work. The softmax activation function inspired its name. This function can compress any real integer into a K-dimensional real vector with each member in the range of [0,1] and a sum of 1. As a result, the output of the softmax function may be considered a probability distribution that can be used to express the predicted probability for various categories. In our case, the segments with and without apnoea. Mathematically, the softmax function [90] can be expressed as equation (2.30)

$$Softmax(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}}, \tag{2.30}$$

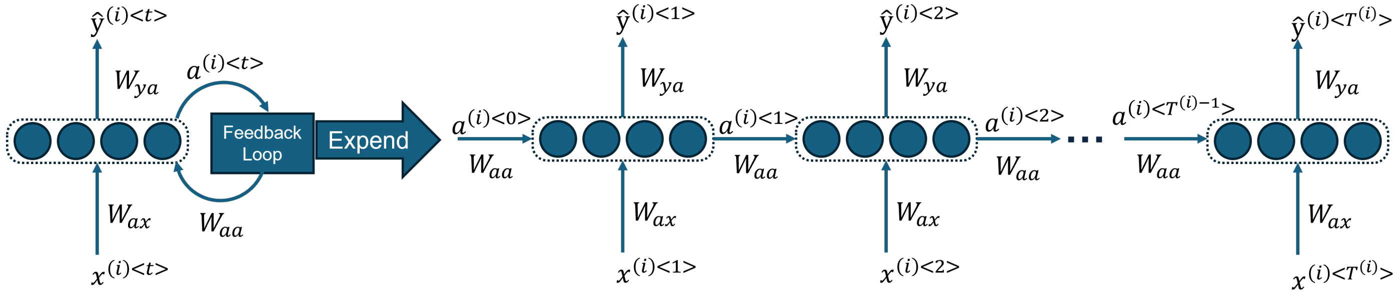
where z is a K-dimensional input vector, i is the i^{th} element in the vector, and K is the total number of categories.

Recurrent Neural Network

A recurrent neural network (RNN) is designed to process sequential data. Unlike traditional feed-forward networks, RNNs feature loops in their architecture that enable them to maintain a form of internal memory [92]. This memory assists the network in recalling previous inputs. Consequently, RNNs are excellent for tasks where the order of data is crucial, such as speech recognition, language processing, and time series forecasting. Figure 2.8 demonstrates the fundamental architecture of a single hidden layer RNN. It contains a single set of input, hidden, and output units, with the hidden units connected through feedback loops. An RNN can be designed as many temporal iterations of the same network, with each iteration transmitting a message to its subsequent part. The trainable parameters of an RNN consist of the weights and biases that are uniformly applied across all time steps [93]. Equations (2.31) to (2.35) express the computations at each time step during the forward propagation of an RNN.

$$\begin{aligned}
a^{<t>} &= \Psi_1(W_{ax}x^{<t>} + W_{aa}a^{<t-1>} + b_a) && (2.31)\\
&= \Psi_1\left(\begin{bmatrix} W_{ax} & W_{aa} \end{bmatrix}\begin{bmatrix} x^{<t>} \\ a^{<t-1>} \end{bmatrix} + b_a\right) && (2.32)\\
&\equiv \Psi_1(W_a[x_{<t>}; a^{<t-1>}] + b_a) && (2.33)\\
\widehat{y}^{<t>} &= \Psi_2(W_{ya}a^{<t>} + b_y) && (2.34)\\
&\equiv \Psi_2(\Psi_2(W_y a^{<t>} + b_y)), && (2.35)
\end{aligned}$$

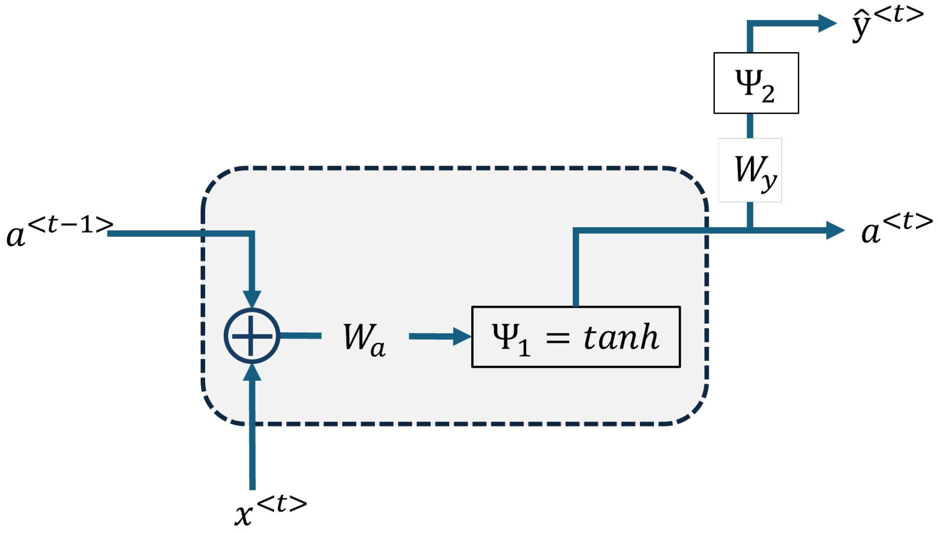
where W_a is a single matrix concatenated by W_{ax} and W_{aa}. The variables b_a and b_y represent the biases linked to the calculation of the hidden state $a^{<t>}$ and the predicted output $\widehat{y}^{<t>}$, respectively. Ψ_1 and Ψ_2 represent the activation function. Normally, $tanh()$ is chosen for Ψ_1. The choice of Ψ_2 depends on the property of the output. The most common ones are $sigmoid, softmax, relu$. These computational processes can be expressed by Figure 2.9. Figure 2.8 mainly describe an RNN architecture designed for the situation when the lengths of the input and output sequences are equal. In many sequence cases, things may be different. Taking the topic discussed in this thesis as an example, sleep apnoea detection is a binary classification problem. The input here is time series data, and the output is a label of "apnoea" or "normal". The RNN model only uses the



Train/Test samples: $(x^{(i)}, y^{(i)})$=($[x^{(i)<1>}, x^{(i)<2>}, \ldots, x^{(i)<T^{(i)}>}]$, $[y^{(i)<1>}, y^{(i)<2>}, \ldots, y^{(i)<T^{(i)}>}]$)

- $x^{(i)<t>}$ - Input data at time step t for the i^{th} training/test samples.
- $y^{(i)<t>}$ -Target at time step t for the i^{th} training/test samples.
- $\hat{y}^{(i)<t>}$ - Predicted output at time step t for the i^{th} training/test samples.
- $a^{(i)<t>}$ - Hidden state (real value) at time step t for the i^{th} training/test samples.
- W_{ax}, W_{ya}, W_{aa} - Weight matrix associated with the input, output and Hidden states, respectively.

Fig. 2.8 Compact and expanded forms of the basic recurrent neural network (RNN) with a single hidden layer.



- $x^{<t>}$ - Input data at time step t.
- $\hat{y}^{<t>}$ - Predicted output at time step t.
- $a^{<t-1>}$, $a^{<t>}$ - Hidden states at time step $t-1$ and t.
- W_a, W_y - Weight matrix associated with the input and output.
- $\oplus$ - Sum

Fig. 2.9 Basic RNN unit cell

final hidden state to produce the final output $\hat{y}$ [94]. Figure 2.10 demonstrates this Many-to-one RNN structure to make it clear. The dotted box in the figure represents an RNN cell.

Long Short-Term Memory (LSTM)

The Long Short-Term Memory (LSTM) model [95] is an advanced recurrent neural network designed to deal with the challenges of exploding and vanishing gradients frequently occurring when learning long-term relationships, even with extensive time lags [96]. The LSTM architecture contains a series of recurrently linked sub-networks called memory blocks. The concept of the memory block is to preserve its state across time and control the information flow through non-linear gating techniques. Figure 2.11 shows the structure of a standard LSTM block, including the gates, the input signal x(t), the output y(t), the activation functions, and the peephole connections [97]. An LSTM unit cell

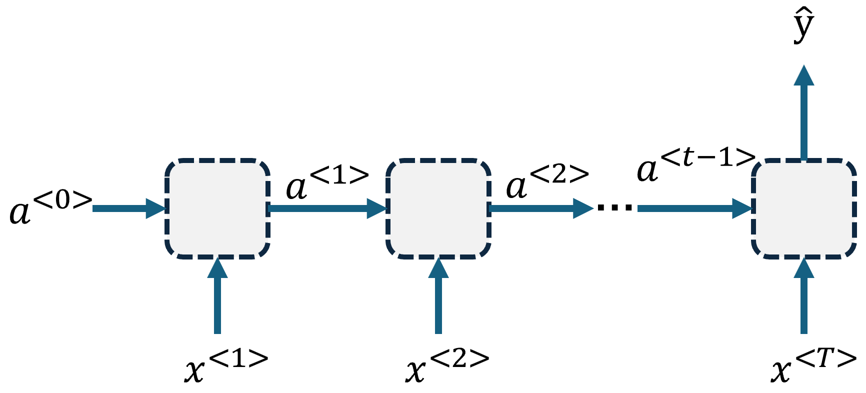
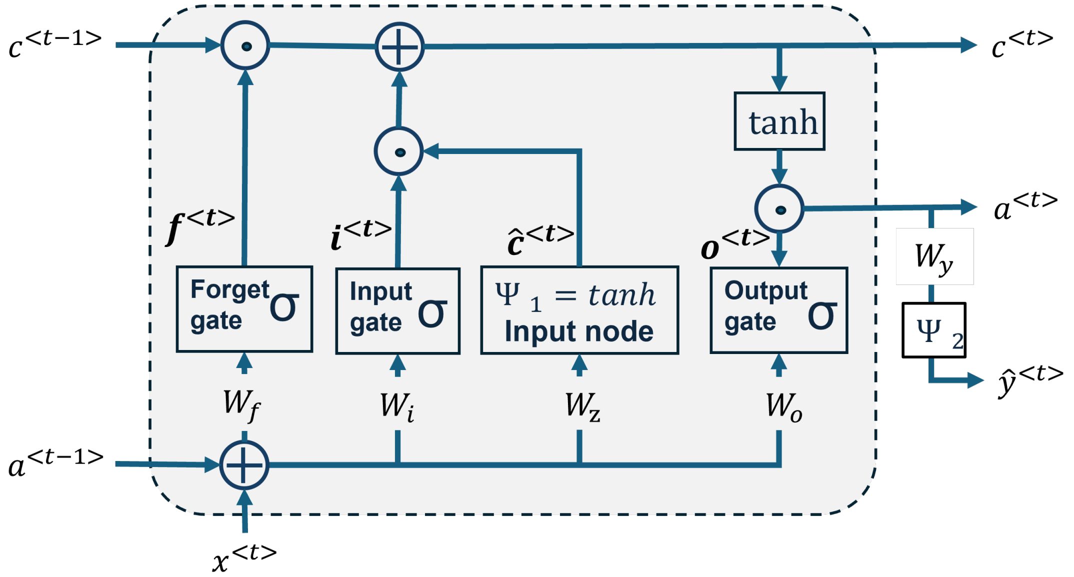


Fig. 2.10 RNN Many-to-one architecture



- $x^{<t>}$, $\hat{y}^{<t>}$ - Input data and predicted output
- $a^{<t-1>}$, $a^{<t>}$ - Hidden state
- $c^{<t-1>}$, $c^{<t>}$ - Internal state
- $\hat{c}^{<t>}$ - candidate internal state
- $f^{<t>}$, $i^{<t>}$, $\hat{c}^{<t>}$, $o^{<t>}$ – Output for forget, input and output gate
- W – Weight matrix
- $\odot$, $\oplus$ - multiplication and sum

Fig. 2.11 LSTM unit cell

fundamentally comprises the internal state and the gates. The internal state c functions as the memory of the LSTM, regulated by the input and the forget gate, and produces a candidate internal state $\widehat{c}^{<t>}$ [93]:

$$\widehat{c}^{<t>} = \Psi_1(W_c x^{<t>} + W_c a^{<t-1>} + b_c), \tag{2.36}$$

where W_z and b_z are the weights and biases related to the input node. The gates of an LSTM operate to manage the flow of information, hence regulating the internal state and output of the memory cell [93]. The input gate controls the new data from the current input to the cell state. Part of the prior cell state data is discarded using the forget gate. The output gate regulates the amount of cell state that is output. These gates are typically implemented using a sigmoid activation function, which produces values between 0 and 1. Therefore, these gates can be considered filters controlling the amount of information that passes through. The calculations for each gate at time step t are provided by the equations (2.37) to (2.39):

$$i^{<t>} = \sigma(W_i x^{<t>} + W_i a^{<t-1>} + b_i) \tag{2.37}$$

$$f^{<t>} = \sigma(W_f x^{<t>} + W_f a^{<t-1>} + b_f) \tag{2.38}$$

$$o^{<t>} = \sigma(W_o x^{<t>} + W_o a^{<t-1>} + b_o), \tag{2.39}$$

where W_i,W_f and W_o are the weights and b_i,b_f and b_o are biases. The updated internal state $c^{<t>}$ is then computed according to the following equation:

$$c^{<t>} = \widehat{c}^{<t>} \odot i^{<t>} + c^{<t-1>} \odot f^{<t>} \tag{2.40}$$

The symbol $\odot$ denotes multiplication. Finally, the predicted output $\widehat{y}^{<t>}$ and hidden state output $a^{<t>}$ as follows:

$$y^{<t>} = \Psi_2(W_y a^{<t>} + b_y) \tag{2.41}$$

$$a^{<t>} = o^{<t>} \odot tanh(\widehat{c}^{<t>}) \tag{2.42}$$

Gated Recurrent Unit

Cho et al. [98] and Chung et al. [99] provide an RNN architecture named Gated Recurrent Unit (GRU), which is simpler and more computationally efficient than the LSTM network. GRU simplifies the input gate and forget gate into one gate called the update gate, and there is no separate memory unit in this structure. Figure 2.12 shows the GRU cell and the variables are explained. The update gate decides how much past information to bring into the current hidden state. The function of the reset gate is similar to that of the forget gate, which determines how much past information should be reset. These two computations are shown by equation (2.43) and equation (2.44).

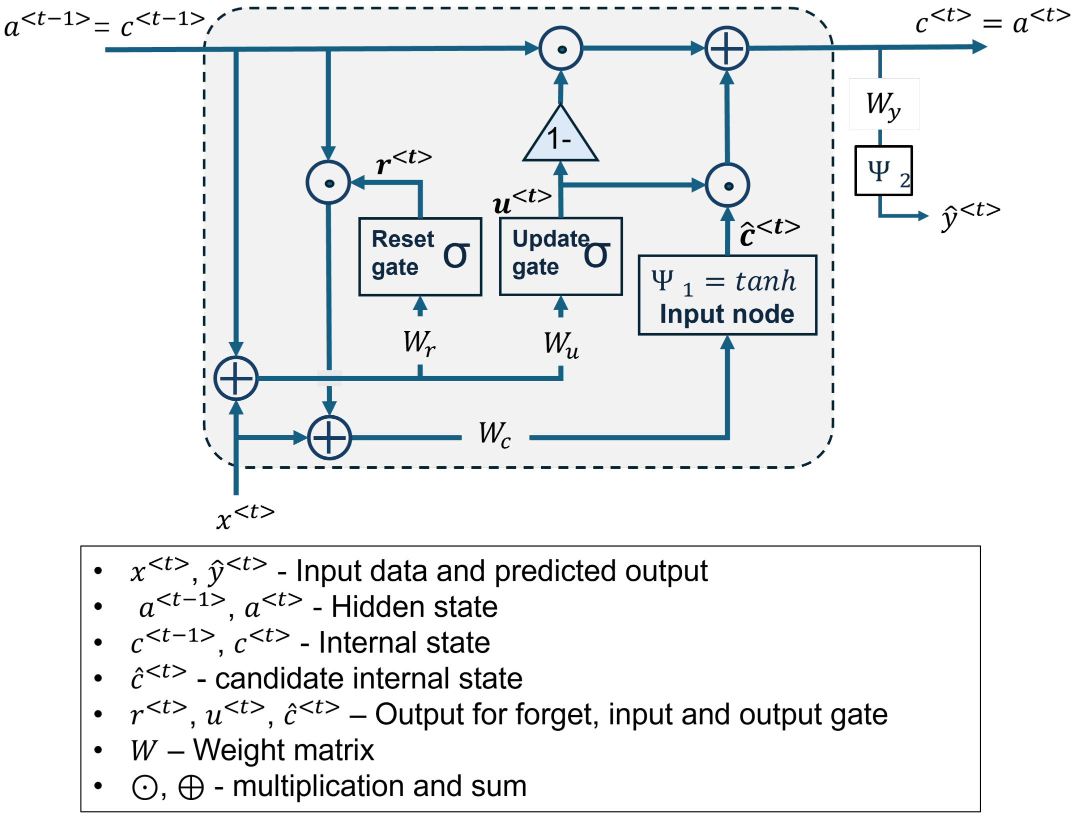

$a^{<t-1>} = c^{<t-1>}$
$c^{<t>} = a^{<t>}$
W_y
Ψ_2
$\hat{y}^{<t>}$
1-
$r^{<t>}$
$u^{<t>}$
$\hat{c}^{<t>}$
Reset gate σ
Update gate σ
$\Psi_1 = tanh$
Input node
W_r
W_u
W_c
$x^{<t>}$
• $x^{<t>}$, $\hat{y}^{<t>}$ - Input data and predicted output
• $a^{<t-1>}$, $a^{<t>}$ - Hidden state
• $c^{<t-1>}$, $c^{<t>}$ - Internal state
• $\hat{c}^{<t>}$ - candidate internal state
• $r^{<t>}$, $u^{<t>}$, $\hat{c}^{<t>}$ – Output for forget, input and output gate
• W – Weight matrix
• ⊙, ⊕ - multiplication and sum

Fig. 2.12 GRU cell

$$r^{<t>} = \sigma(W_r c^{<t-1>} + W_r x^{<t>} + b_r) \tag{2.43}$$

$$u^{<t>} = \sigma(W_u c^{<t-1>} + W_u x^{<t>} + b_u), \tag{2.44}$$

where b_r and b_u is the biases. Then, based on these two gate outputs, the hidden state output is calculated:

$$\widehat{c}^{<t>} = \Psi_1(W_c x^{<t>} + W_c c^{<t-1>} + b_c) \tag{2.45}$$

Finally, the internal state $c^{<t>}$ and the output $\widehat{y}^{<t>}$ are computed:

$$c^{<t>} = c^{<t-1>} \odot (1 - u^{<t>}) + u^{<t>} \odot \widehat{c}^{<t>} \tag{2.46}$$

$$\widehat{y}^{<t>} = \Psi_2(W_y c^{<t>} + b_y) \tag{2.47}$$

2.6.4 Related Work

Machine learning and deep learning have received much attention over the past few years as methods to improve the accuracy, efficiency, and accessibility of sleep testing. SVM, RF, and KNN are some of the most commonly used machine-learning techniques for classifying sleep stages and detecting sleep disorders. Machine learning is to train processed features extracted from physiological signals, such as EEG, ECG, etc.. In contrast, deep learning models, especially CNN and RNN, can learn complex patterns independently from raw data. This means that feature engineering does not need to be done manually and the model is more efficient overall.

Deep learning methods tend to be more accurate and generalise better when working with large datasets; however, they require more data and are more computationally complex. This makes machine learning-based methods more useful when resources are limited. This subsection reviews past research using machine learning and deep learning for sleep analysis and weighs the advantages and disadvantages of each approach.

Machine Learning

Recent machine learning and AI developments have enabled more efficient, cost-effective, and non-invasive diagnostic approaches [100]. The study examined 132 research articles and proved how AI-based techniques could efficiently detect and classify apnoea occurrences using electrocardiograms, pulse oximeters, and sound signals. The study points out that while present research provides promising outcomes, additional research in data integration, model correctness, and real-world validation is required before widespread clinical application.

Deep Learning

With the development of AI and automation, deep learning has gradually been proven to improve the accuracy of sleep analysis. By analysing research published between 2008 and 2018, Mostafa et al. [101] provide insights into the effectiveness, advantages, and potential future directions of deep learning for sleep apnoea detection. Same as [100], although CNN, RNN, and hybrid models have shown high performance, they believe that ongoing research needs to focus more on overcoming data challenges and ensuring clinical implementation.

Most sleep analysis applications nowadays concentrate on feature extraction and signal processing. Sillaparaya et al. proposed a deep-learning approach to classify OSA using snoring sounds. They obtained an accuracy of 85.25% by using Mel-frequency cepstral coefficients (MFCC) and a three-layer fully connected network. However, there is still more work to be done, as demonstrated by the difficulties presented by feature overlap, class imbalance, and small datasets. Barnes and his colleagues [102], unlike Sillaparaya and his group, used a single-channel EEG and created a CNN architecture with three convolutional layers to detect sleep apnoea. Their framework is more understandable than others because they train a CNN classifier using raw EEG data. This provides inspiration and evidence for our direct use of signals.

In addition to determining whether or not sleep apnoea occurs, there are also classification approaches for the severity of detected apnoea. The AHI quantifies the severity of the condition [16], which is explained in Chapter 1. A new method to estimate AHI was created by Werthen-Brabants et al. [103]. They named it RSN-Count and based it on a Recursive Spiking Neural Network (RSNN). They use CNN to pull out features and RSN-Count to treat apnoea events as separate units in time. This is different from the normal CNN classifier. CNN is a spatial feature-focused algorithm. It is ideal for this study because the goal is to count apnoea rather than time series positions.

Sleep stage classification is also a popular topic in sleep analysis. Jia et al. [104] proposed SleepPrintNet, a deep learning model that integrates EEG, Electrooculography (EOG), and Electromyography (EMG) signals for automatic sleep staging. The 1D-CNN part of this deep learning model processes raw signals to find time-domain patterns and temporal dependencies. In the 2D-CNN part, spectral-spatial maps of EEG signals are used to find frequency and spatial relationships in the signals. Finally, all the features are concatenated and filled in the classification layer to get the result. This network emphasises the focus of CNN in different dimensions to extract features, which is also the effect expected to be achieved in this study.

Since this study uses two data types, a two-dimensional (2D) CNN is the model basis. Jiménez-García et al. [105] published a 2D CNN model to detect sleep apnoea in children using airflow and oximetry. They used a CNN model with two convolutional blocks to classify the severity of apnoea in children. Due to the extreme data imbalance, the classification results were not satisfactory. Later, they improved the model by adding Bidirectional Gated Recurrent Units (BiGRU) as a prediction model [106]. The authors of [106] pointed out that BiGRU is used to analyse the temporal patterns of data in two directions. However, it was not compared with the Gated Recurrent Units (GRU) model [107]. Our study changes this model to a GRU model and found that the overall ability to predict apnoea was better than BiGRU.

In addition to CNN-RNN, the CNN with a support vector machine (CNN-SVM) hybrid model is a popular classification model nowadays. Baresary and his colleague apply CNN-SVM to classify sleep apnoea by using PSG [108]. The approach demonstrated strong performance. However, they did not give the dataset conditions, and the details and parameters of the model are unknown.

They simulated noise and added it to the raw data to get closer to reality. The current trend in apnoea detection is to use less and less expensive signal data to detect the disease. Although their research was highly accurate, it was time-consuming and costly.

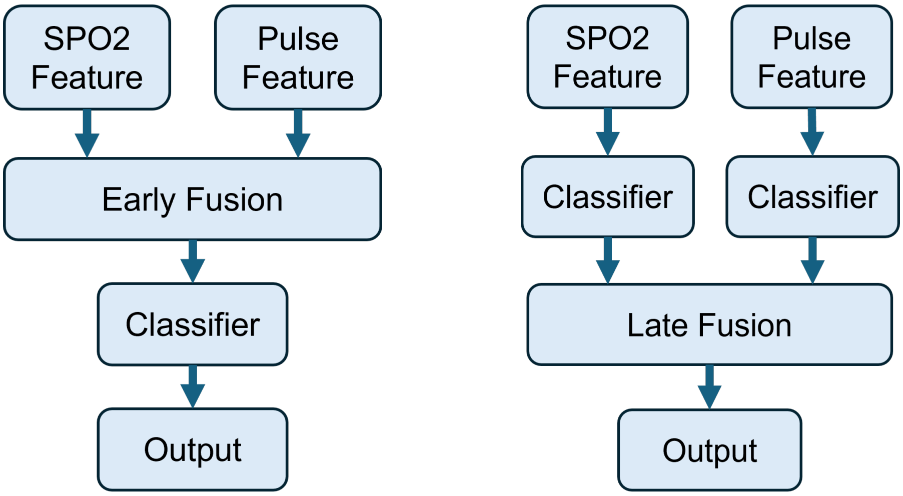
2.7 Feature Fusion

Given that this study encompasses two types of signals, it is essential to consider integrating the signal features before inputting them into the classifier. Regarding the sequence of fusion and prediction, feature fusion is categorized into early fusion and late fusion [42]. Early fusion involves combining the features before input into the classifier, while late fusion entails inputting the features into the classifier independently and then fusing the prediction scores.

2.7.1 Fusion Scheme

Based on the processing level at which the fusion occurs, the conventional feature fusion techniques usually fall into two categories: early and late fusion. Snoek et al. [109] compares two approaches to combining information from different modalities. Early fusion is fusing multiple feature layers first and then training the classifier on the fused features (unified detection is performed only after complete fusion). This type of method is also called skip connection. As shown in Figure 2.13(a), Early fusion integrates features from different modalities into a unified representation before applying machine learning. Only one training session is enough to achieve classification results. However, the challenge involves effectively integrating different features into a common representation.

Late fusion, on the contrary, improves detection performance by combining classification results from different layers (see Figure 2.13(b)). Each classifier is analysed separately, and different features are learned individually. Then, the prediction scores from each classifier are fused. This approach shows the advantages of different classifiers but needs multiple training processes, which



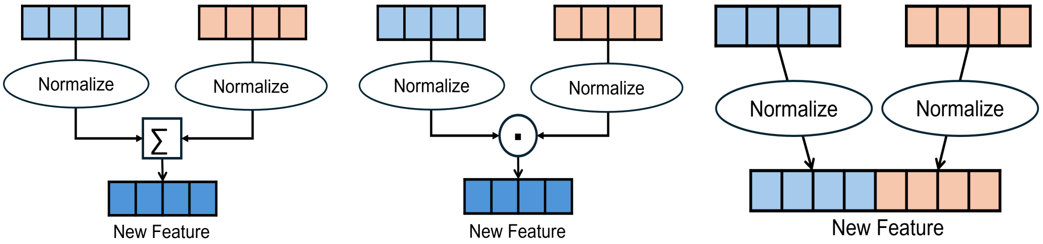
(a) Decision tree architecture (b) Decision tree flow chart

Fig. 2.13 Feature fusion concept diagram

increases the computing cost. In addition, correlations between features may be reduced.

2.7.2 Fusion Method

Michelsanti and his colleague [42] list various fusion methods. Figure 2.14 shows the most popular fusion methods. Initially, all features are normalized to ensure



(a) Addition-Based fusion (b) Product-Based fusion (c) Concatenation-Based fusion

Fig. 2.14 Simplified graphical example of the fundamental fusion methods.

uniformity. In Addition-based fusion, different features are summed algebraically, including mean, pooling, and weighted sum. Similarly, different features are multiplied in Product-based fusion to derive a new feature. In Concatenation-based fusion, different features will be connected in series to formulate a novel integrated feature.

Since feature fusion is the fusion of features of multiple dimensions, this technology is primarily used in deep learning networks. This study is mainly for data analysis, so the selected features are of a single dimension. When fusion, one feature a can be used as the real part and the other feature b as the imaginary part to generate new features in the form of $y = a + bi$. However, it increases the computational complexity. The primary purpose of this study is to explore feature fusion's impact on sleep apnoea detection, so Concatenate-based fusion is used.

In addition to these fundamental fusion methods, other advanced methods, such as Attention-based fusion and Squeeze-Excitation fusion, have been used to improve feature integration. Attention-based fusion [110] is a novel scheme for combining features from different layers or branches in neural networks using attention mechanisms. It assigns weights based on the importance of the features to solve the problem of semantic and scale inconsistencies. This function improves the quality of fused features in various scenarios, such as skip connections and multi-scale layers. Squeeze-and-excitation fusion is a method to enhance the most valuable features and suppress redundant features during feature fusion [111]. The "squeeze" operation aggregates global information by recalibrating channel-wise feature responses, while the "excitation" operation emphasises interdependencies among features.

2.7.3 Related work

This section evaluates all these methods described in Subsection 2.7.2 and their contributions to different applications.

Chen et al. [112] introduced a novel bottleneck attention fusion network (BAFNet) to detect sleep apnoea. Bottleneck attention fusion enhances feature fusion flexibility and reduces computational complexity. It also controls the information flow between R-R intervals (RRI) and R-peak amplitudes (RPA) networks, ensuring the model focuses on key features for sleep apnoea detection. Addition-based fusion is usually used in network features, especially in data of different dimensions. Afouras et al. [113] apply 1D-ResNet separately on enrolment Audio, occluded Video, and noisy audio to build an audio-visual model for separating a single speaker from a mixture of sounds, such as other speakers and background noise. They use Addition-based fusion to summarise the 1D-ResNet features and get enhanced audio.

Since this study is about the temporal signal of sleep, directly adding or multiplying the features obtained from sleep data will lose the required temporal information. This is explained in the paper OSA detection using Feature Level Fusion [33]. The author uses concatenation fusion at the feature level and gets an accuracy of 96.52%. Although his features are different from ours, they still inspire the feature fusion of this study.

In addition to basic fusion technology, some extended fusion methods have been appropriately applied. Shu and his colleagues [114] proposed a novel expansion-squeeze-excitation fusion network (ESE-FN) to effectively solve the problem of activity recognition for the elderly. In human activity recognition, the data from each sensor may provide unbalanced discriminative information. Regarding this issue, Laitrakun [115] proposes the Merge-Squeeze-Excitation (MSE) feature fusion, which emphasizes informative feature maps and suppresses ambiguous feature maps during the fusion process.

Chapter 3

Sleep Apnoea Detection with an Adaptive CUSUM Approach

As mentioned in section 2.4, sleep apnoea events can be regarded as abnormal events in sleep breathing, which means that the detection of apnoea can be regarded as change point detection. This Chapter introduced an anomaly detection approach using the adaptive CUSUM change point detection algorithm to monitor outliers in the signal. The study focused on two signals, the pulse signal and the oximetry signal, intending to detect apnoea using a single signal. In addition, the test results of the adaptive CUSUM will be compared with the test results of the classical CUSUM. Additionally, the threshold selection has been modified from a constant to a value related to the standard deviation of the selected signal, based on a rational subgroup process. The results of the comparison confirm that the adaptive CUSUM is better than the classical CUSUM in the accuracy of automatic detection.

3.1 Classical CUSUM Approach

The CUSUM method is a sequential analysis algorithm developed by E. S. Page of the University of Cambridge [116]. It is often used to monitor change detection [117]. CUSUM has different methods to determine if the process is out of control. Rather than finding non-constant control limits for the CUSUM, it is easier to transform the CUSUM to a score for which the control limits are constant. This separate score is called the Mean adjusted CUSUM [118], which is mainly introduced here.

Instead of examining the mean of each subgroup independently, the CUSUM plot displays information accumulating current and previous samples. Therefore, a CUSUM chart is generally better at detecting small changes in the process mean than an X-bar chart. CUSUM plots rely on the specification of the target value and the standard known or reliable estimated deviation. Therefore, after establishing process control, it is best to use the CUSUM chart [119]. CUSUM plots typically represent runaway process summations by accumulating upward or downward drift until it crosses a boundary. Following the CUSUM procedure presented by Koshti[120], the steps for creating a CUSUM control chart can be summarized as follows.

Let us collect m samples, each of size n, and compute each sample's mean μ_i. A Gaussian (normal) distribution $\mathcal{N}(.)$ is considered in the analysis:

$$x_i \sim \mathcal{N}(\mu_i, {\sigma_i}^2), \tag{3.1}$$

where μ_i is the mean and σ_i is the standard deviation of the samples. Then, the CUSUM control chart is formed by plotting one of the following quantities [121]:

$$C_m = \sum_{i=1}^{m}(x_i - \hat{\mu}_i), \tag{3.2}$$

against the sample number m, where $\hat{\mu}_i$ is the estimate of the in-control mean.

The Tabular CUSUM for Monitoring the Process Mean

In the last part of the introduction, when the process is under control, x_i has a normal distribution with mean μ_i and standard deviation σ_i (known or estimable). The tabular CUSUM works by accumulating deviations from μ_i, which are displayed as C^+ and C^-. C^+ means one sided upper and C^- means one sided lower. With the starting value $C_i^+ = C_i^- = 0$, they are calculated as equation (3.3) and equation (3.4) [116] .

$$C_i^+ = max[0, x_i - (\mu_i + k) + C_{i-1}^+], \tag{3.3}$$

$$C_i^- = max[0, (\mu_i - k) - x_i + C_{i-1}^+], \tag{3.4}$$

where μ_i is the process mean at time step i. k is the "slack" allowed in the process, and it is chosen halfway between the target μ_0 and the mean shift of interest μ_1 to detect

$$k = \frac{1}{2} \cdot |\mu_1 - \mu_0| \tag{3.5}$$

Here, the μ_0 denotes the in-control mean, representing the baseline expected value of the process. mu_1 denotes the out-of-control mean, which corresponds to the shifted level that the test is designed to detect. In practice, the mean at step i, μ_i, is equal to μ_0 when the process remains stable. Once a shift occurs, it moves toward μ_1.

In the CUSUM method, the cumulative sums C^+ and C^- track positive and negative deviations from a target value μ_0, accumulating only deviations that exceed a threshold k. In this case, with a chosen mean shift of 1, the value of k is set at 0.5. The process is monitored against a decision interval H. If either C^+ or C^- surpasses H, it indicates that the process is out of control [122]. The H interval is determined as follows

$$H = h \times \sigma, \tag{3.6}$$

where h is a commonly used constant whose values are between 2 and 4 and $\times$ denotes the multiplication operation. In the adaptive CUSUM algorithm presented in the next section, the value of σ varies according to a decision-making rule.

3.2 Adaptive CUSUM Based on Log-likelihood Ratio.

The adaptive CUSUM is presented to overcome the problem of unknown parameters changing over time. The combination of detecting changing processes and estimating parameters is thought to improve performance [123]. The concept is to guess the parameters in a continuous form, with the CUSUM test beginning immediately regardless of the precision of the prediction. Because more sample estimation may result in more accurate estimation, the estimation procedure proceeds while detection is performed [124]. This section mainly introduced what the adaptive CUSUM is and how it improves the classical CUSUM algorithm.

Let $X = \{x_1, x_2, ..., x_n\}$ be a random collection of data received consecutively. For the sleep apnoea analysis, these are typically from PSG, such as SPO2 and pulse data. We suppose that each value x_i belongs to a known pdf $p(x_i, \theta)$, in this case, to a Gaussian distribution. These samples have a known mean μ_i and variance σ^2. These samples may have a change happened at time $\widetilde{t_c}$ called apnoea that is modeled by and instant modification to the value of θ, therefore, there are two possible hypotheses: $\mathcal{H}_0$ for pre-change (with a parameter $\theta = \theta_0$) and $\mathcal{H}_1$ for post-change (with other parameters $\theta = \theta_1$). These hypotheses can be expressed as:

$$H_0 : X_i \sim \mathcal{N}(\mu_{\mathcal{H}_0}, \sigma^2) \tag{3.7}$$

$$H_1 : X_i \sim \mathcal{N}(\mu_{\mathcal{H}_1}, \sigma^2) \tag{3.8}$$

The parameter θ is in the form: $\{\mu, \sigma\}$.

The likelihood function for a normal distribution $\mathcal{N}(\mu, \sigma^2)$ is given by:

$$p(x_i|\mu) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left(-\frac{(x_i - \mu)^2}{2\sigma^2}\right) \tag{3.9}$$

The instantaneous log-likelihood ratio test is used to decide between the two hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$. The log-likelihood ratio is written based on a normal distribution as:

$$S_i = ln\left(\frac{p(x_i, \theta_{\mathcal{H}_1})}{p(x_i, \theta_{\mathcal{H}_0})}\right) \tag{3.10}$$

$$= ln\left(exp\left(-\frac{(x_i-\mu_{\mathcal{H}_1})^2}{2\sigma^2}+\frac{(x_i-\mu_{\mathcal{H}_0})^2}{2\sigma^2}\right)\right) \tag{3.11}$$

$$= \frac{(x_i-\mu_{\mathcal{H}_0})^2-(x_i-\mu_{\mathcal{H}_1})^2}{2\sigma^2} \tag{3.12}$$

$$= \frac{2x_i(\mu_{\mathcal{H}_1}-\mu_{\mathcal{H}_0})+\mu^2_{\mathcal{H}_0}-\mu^2_{\mathcal{H}_1}}{2\sigma^2} \tag{3.13}$$

$$= \frac{\mu_{\mathcal{H}_1}-\mu_{\mathcal{H}_0}}{\sigma^2}(x_i-\frac{\mu_{\mathcal{H}_1}+\mu_{\mathcal{H}_0}}{2}) \tag{3.14}$$

Then, the cumulative sum of S_i from 0 to n is

$$S_n = \sum_{i=0}^{n} S_i \tag{3.15}$$

The decision function G_n and the changing time estimate $\widetilde{t_c}$ are:

$$G_n = S_n - \min_{1 \leq t_c \leq n} S_{t_c-1}, \tag{3.16}$$

$$\widetilde{t_c} = \min_{1 \leq t_c \leq n} S_{t_c-1} \tag{3.17}$$

When the difference G_n between the value of the cumulative sum S_n and its present minimum value at time n is greater than a specified threshold value h, CUSUM detects a change in sample x_i [61]. The equation (3.15) to (3.17) provides the direct formulations of the adaptive CUSUM algorithm. For purposes of real-time change detection, the equations are reformulated in a recursive format as follows:

$$S_n = S_{n-1} + S_n, \tag{3.18}$$

and the decision function G_n written in the same format as the CUSUM algorithm as equation (3.3) is:

$$G_n^+ = max[0, G_{n-1} + S_n] \tag{3.19}$$

This decision function will be compared with the threshold H, and the excess is considered a change, which might be apnoea.

Threshold Selection Based on Rational Subgroup

The change detection process uses the expression of the decision interval H from equation (3.6). However, different sample sizes affect the variance, which means that the thresholds for each window are not defined under the same conditions [125]. To solve this, the standard deviation σ should be replaced by sampled deviation in rational subgroup [126] (of sizes $n > 1$):

$$\sigma_{sample} = \frac{\sigma}{\sqrt{n}} \tag{3.20}$$

In the classical CUSUM algorithm, σ for threshold calculation is the deviation of signal data value. Usually, the threshold of the adaptive CUSUM is a self-defined constant within a specific range. In this test, according to the same threshold calculation idea (using variance), the adaptive CUSUM uses the decision function Gm to calculate σ and adds overlap to achieve an adaptive threshold.

3.3 Implementation and Analysis

3.3.1 Data Analysis and Pre-processing and Initialization

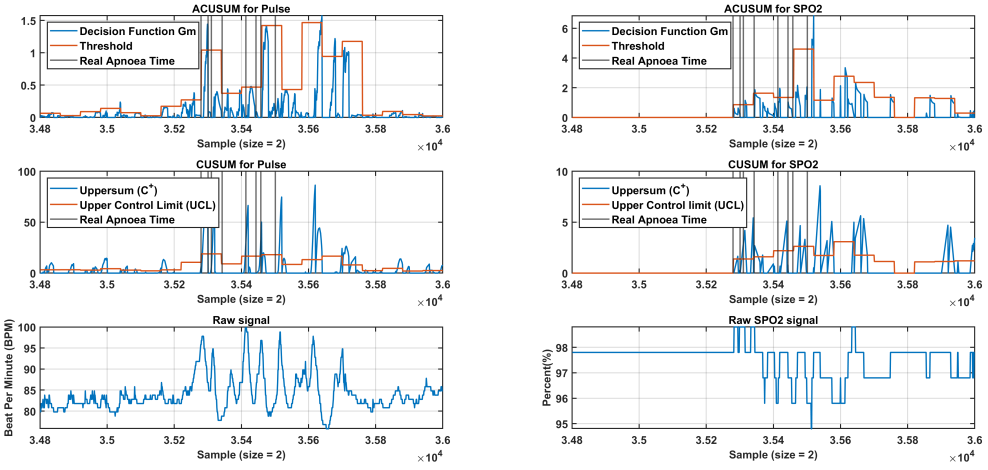
The data utilized in this study is sourced from the National Sleep Research Resource (NSRR) [21], which provides a repository of 387 EDF files. Each file contains diverse types of data, such as pulse and oxygen levels, which are pertinent to this research. The initial step involves extracting data from each file individually and converting it from EDF to MAT format to facilitate analysis. Given the potential for erroneous data due to device movement while the subject is asleep, any dataset containing more than half erroneous values must be discarded. For example, a pulse data file may be considered unreliable and thus excluded if over half of its values are erroneously read 300. This research aims to evaluate both individual and combined detection methods. Consequently, if any data file—be it oxygen or pulse data—exhibits an excessive rate of error, that patient's entire set of data will be considered compromised and excluded from further analysis. Moreover, any erroneous readings identified within the selected datasets will be replaced with the correct subsequent values available to ensure data integrity. Finally, accurate recording of the sampling frequency is essential, as it aids in setting the threshold for each analysis window, ensuring that the data segmentation aligns correctly with physiological events observed during sleep. (equation (3.20)).

Based on AASM, the window size of this experiment is 30 seconds. In order to make the experiment more accurate, overlap is set when moving the window so that each epoch can be compared before and after.

3.3.2 Detection Results for Oximetry and Pulse Data

In this detection, the window size (WL) for calculation is one epoch (30 seconds), which means $WL = 30 \times fs(sample\ size)$. In order to get higher accuracy of detection, an overlap was added in this algorithm, which is set to 10. In addition to this, the mean shift is set to default 1. Furthermore, as Subsection 3.3.1 mentioned, after data processing, there are 367 data sets used in the test.

Figure 3.1 gives the sleep signal change detection results and a comparison between the adaptive CUSUM and classical CUSUM algorithms. The first two of each figure are the results of two algorithms. The blue line in the adaptive CUSUM is the decision function (G_m) from equation 3.16 and the blue line in CUSUM is the upper sum (C^+) from equation 3.3. The red line is the threshold in the different algorithms above, which the signal is out of control, considered sleep apnoea. The black vertical line is the actual sleep apnoea occurrence time, which is used to compare with the detection results. The third figure of each is the original sleep signal. By comparing with the original signal, it can be seen that both algorithms can detect signal changes to varying degrees. However, compared with the actual time of apnoea occurrence time, the CUSUM algorithm



(a)

(b)

Fig. 3.1 Comparison between Adaptive CUSUM Algorithm and classical CUSUM on Oxygen Signal.

can detect the moment of signal change, while the adaptive CUSUM algorithm can more effectively distinguish whether the apnoea occurs in an epoch.

Table 3.1 Average Performance of CUSUM.

	acc	sens	spec	F1-N	F1-A	kappa
Combine	**89.0%**	11.8%	**90.9%**	**0.9398**	0.0210	-0.002
SPO2	74.8%	**11.9%**	75.7%	0.8451	**0.0330**	**0.002**
Pulse	73.7%	6.1%	74.9%	0.8420	0.0326	-0.002

Table 3.2 Average Performance of ACUSUM.

	acc	sens	spec	F1-N	F1-A	kappa
Combine	**91.8%**	**38.2%**	**93.9%**	**0.9536**	0.045	0.017
SPO2	85.7%	26.3%	87.5%	0.9147	**0.075**	**0.039**
Pulse	79.6%	17.3%	81.1%	0.8742	0.042	0.003

Table 3.1 and Table 3.2 gives the average detection results of CUSUM and ACUSUM. The adaptive CUSUM algorithm demonstrated superior overall performance compared with the classical CUSUM. The combined detection strategy, implemented through an AND operation across individual detections, yielded higher accuracy than single detections, consistent with expectations. Nevertheless, the sensitivity of both methods remained below 50%, which is

unsatisfactory for medical applications. This limitation arises because CUSUM primarily detects changes in time series, whereas variations in respiratory signals may reflect other events, such as hypopnoea, rather than apnoea. Notably, both CUSUM and ACUSUM achieved higher F1-scores (Apnoea) and Kappa values with SPO2 data than with Pulse, indicating that SPO2 is more informative for apnoea detection.

In terms of efficiency, adaptive CUSUM was considerably slower than classical CUSUM, requiring approximately ten minutes compared with one second on average, with computation time increasing further for larger sample sizes.

3.4 Summary

This chapter introduces an automatic sleep apnoea detection tool, the adaptive CUSUM, which analyzes blood oxygen and pulse data to detect abnormalities. The main goal is to reduce the reliance on traditional polysomnography (PSG), reduce diagnostic costs, and improve detection results. This method is a more effective and economical sleep apnoea detection method than PSG because PSG requires doctors to manually process data and go to the hospital for night monitoring, which is time-consuming and labour-intensive. To solve this problem, this chapter proposes an improved CUSUM algorithm to improve detection accuracy by adaptively adjusting the threshold. Compared with the traditional CUSUM method, the adaptive CUSUM performs better in detecting abnormal points in sleep signals, especially for non-stationary time series data, such as physiological signal fluctuations during sleep. Experimental results show that the adaptive CUSUM has a higher detection accuracy than the traditional CUSUM method. The adaptive CUSUM has a detection accuracy of 79.57% for pulse data and 85.74% for blood oxygen data. CUSUM has a pulse signal recognition accuracy of 73.68% and an SPO2 detection accuracy of 74.85%. The combination of pulse and blood oxygen data results shows that the adaptive CUSUM has a detection accuracy of 91.78%, exceeding CUSUM 88.97%. In terms of accuracy, this algorithm demonstrates superior performance compared to existing approaches. For reference, Jimenez-García et.al. [105] applied their method to the CHAT dataset and reported an accuracy of 84.64% for apnoea detection. However, the sensitivity of their method was 92.5%, which is much higher than ours. The research indicated that the adaptive CUSUM had superior anomaly detection capabilities and could more efficiently identify sleep apnoea occurrences. Nonetheless, this approach has a considerable computational cost. Analysing each data set requires 10 minutes, while conventional CUSUM requires just 1 second.

Future works will concentrate on enhancing the computational efficiency of the algorithm for real-time detection while also broadening the study's scope to differentiate various types of sleep apnoea (obstructive OSA, central CSA, mixed MSA) and integrating machine learning techniques to augment the model's generalisation capability. Furthermore, other physiological data (such as electrocardiograms and breathing) can be added to enhance the precision and usefulness of automated sleep monitoring. The results indicate that this approach may serve as an alternative to PSG, enabling patients to do more convenient sleep apnoea assessments at home and promoting the advancement of telemedicine.

Chapter 4

Machine Learning Methods for Apnoea Detection

Sleep signals are time series signals that contain a lot of time and frequency information. The change point detection method, Adaptive CUSUM, in Chapter 3 focuses on time information and ignores frequency changes. This Chapter presents a novel framework that uses the wavelet transform to extract features from sleep signals and the RUSBoost algorithm to address the challenge of imbalanced data in detecting sleep apnoea, which enables home self-monitoring. Patient data features short apnoea epochs and long periods of normal breathing, creating imbalances that challenge classification algorithms. The framework was tested on three public datasets with varying imbalance ratios. Significantly, the CHAT dataset [21] with an 'apnoea' to 'normal' period ratio of 1:15 effectively reflects actual sleep apnoea signals from children. The proposed framework with the CHAT dataset achieved a maximum accuracy of 91.54%, a sensitivity of 72.06%, a specificity of 92.39%, and an AUC of 0.923, surpassing state-of-the-art home screening models. This study compared several machine learning techniques for the classification task, including SVM, KNN, and DPGMM algorithms. It is found that the RUSBoost algorithm provides the most accurate results when the ratio of the 'apnoea' to the 'normal' period reaches an imbalance of 1:3 or greater.

4.1 Classification Based on Dirichlet Process Gaussian Mixture Model

4.1.1 Gaussian Mixture Model

A Gaussian Mixture Model (GMM) is a parametric probability density function represented by a weighted sum of Gaussian component distributions [127]. GMMs are commonly applied as parametric models to represent the probability distribution of continuous measurements or features in biometric systems, such as the frequency characteristics of blood oxygen levels in sleep-related signals.

A GMM assumes that the observed data X is generated from a mixture of K Gaussian distributions. The probability density function (PDF) of the mixture model is given by:

$$p(X) = \sum_{k=1}^{K} \pi_k \mathcal{N}(X|\mu_k, \Sigma_k), \tag{4.1}$$

where K is the number of Gaussian components. π_k denotes the mixing coefficients, which are satisfied $\sum_{k=1}^{K} \pi_k = 1$ and $0 \leq \pi_k \leq 1$. $\mathcal{N}(X|\mu_k, \Sigma_k)$ is a Gaussian distribution with mean μ_k and covariance matrix Σ_k. It is defined as:

$$\mathcal{N}(X|\mu_k, \Sigma_k) = \frac{1}{(2\pi)^{d/2}|\Sigma_k|^{1/2}} exp\left(-\frac{1}{2}(X-\mu_k)^T \Sigma_k^{-1}(X-\mu_k)\right), \tag{4.2}$$

where d is the dimensionality of the data. T and -1 represent the transpose and inverse of the matrix.

4.1.2 Dirichlet Process

A Dirichlet Process (DP) is a Bayesian non-parametric model providing a distribution between distributions [128], therefore allowing an infinite range of possible mixing components. The DP is usually used when the number of clusters is unknown. For a random distribution G to follow a DP, its marginal distributions must be a Dirichlet distribution. Mathematically, assume H as a distribution over θ and α as a positive real-valued parameter. for any finite partition $\{A_1, ..., A_K\}$

of the space, the probability mass assigned to these partitions follows a Dirichlet distribution, which is written as $G \sim DP(\alpha, H)$. The expansion equation is:

$$(G(A_1), G(A_2), ..., G(A_k)) \sim \mathcal{D}\rangle\nabla(\alpha H(A_1), \alpha H(A_2), ..., \alpha H(A_k)) \qquad (4.3)$$

The DPMM enhances Gaussian Mixture Models by enabling the number of clusters to be inferred from the data instead of predetermined. This adaptability renders the Dirichlet Process a potent instrument in Bayesian nonparametrics, facilitating the automatic identification of latent patterns inside data.

4.1.3 Stick-breaking Process

The definition of the DP is challenging to apply directly because it is not observable. Researchers use various methods for sampling from DPs to address this issue. The stick-breaking process is a strong and straightforward way to define DP by successively producing probability weights for an infinite number of components. It involves sequentially drawing samples from marginal beta distributions to construct a sample from a Dirichlet distribution. Figure 4.1 gives an intuitive conceptual diagram of the stick-breaking process. From the figure, the weight of the component after each stick-breaking is π. v is based on the

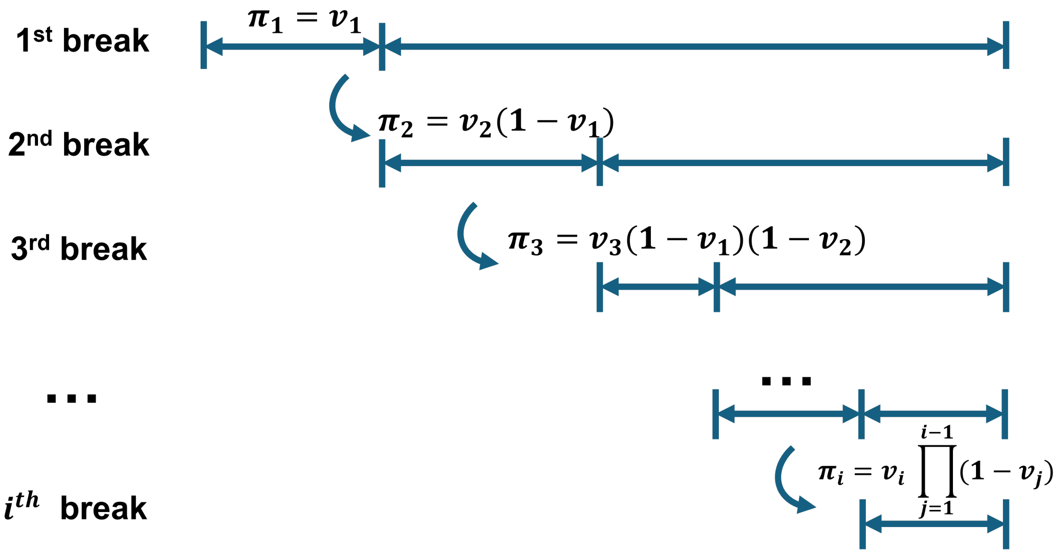


Fig. 4.1 An overview of stick-breaking process

weight of the previous stick, and v follows the beta distribution.

$$v_i \sim Beta(1, \alpha), \quad i = 1, 2, ... \tag{4.4}$$

where α is the concentration parameter that controls the number of effectively used components. The weight of each component is computed as:

$$\pi_i = v_i \prod_{j=1}^{i-1} (1 - v_j), \tag{4.5}$$

The stick-breaking process provides a straightforward method for sampling from a DP and is commonly used in Bayesian nonparametrics, particularly in DPMM. It facilitates adaptable and practical clustering by autonomously identifying the number of clusters from the data, eliminating the need for prior specification.

4.1.4 Variational Dirichlet Process Gaussian Mixture Model

Blei and Jordan originally proposed a variational Dirichlet Process Gaussian Mixture Model (DPGMM) [129] and subsequently optimized by Kurihara et al. [130]. Building on these foundational methods, Li et al. [40] developed a clustering-based classification framework specifically used for sleep apnoea detection. The model in this framework uses the LLR for data classification.

The Dirichlet Process mixture model in the stick-breaking [131] representation involves an infinite number of components, where each component is associated with a set of parameters drawn from a base distribution H. The mixing proportions are determined via a stick-breaking process governed by a parameter α. For more information, please see the first three subsections of this section. The specific process is as follows:

$$\eta_i \sim H, \tag{4.6}$$

$$v_i \sim Beta(1, \alpha), \tag{4.7}$$

$$\pi_i = v_i \prod_{j=1}^{i-1} (1 - v_j), \tag{4.8}$$

where $\{\pi_i\}_{i=1}^{\infty}$ is the mixing weight. $v_i \in [0, 1]$ which is an infinite collection of 'stick lengths' $V = \{v_i\}_{i=1}^{\infty}$. In this research, features are extracted and combined from each segment of the SPO2 and pulse signal, categorized as either 'apnoea' or 'normal.' It is posited that the distribution of features extracted from apnoea and normal segments differs. Consequently, decisions can be made by comparing the probabilities of each test segment under the 'apnoea' and 'normal' models. Features from 'apnoea' and 'normal' segments can be represented using two GMMs, as a GMM can accurately approximate any distribution when configured with the appropriate number of components and parameter adjustments. The likelihood for a data point x in a model is expressed as:

$$p(x|\{\pi_i, \eta_i\}_{i=1}^{\infty}) = \sum_{i=1}^{\infty} \pi_i \mathcal{N}(x; \eta_i), \tag{4.9}$$

where $\mathcal{N}(\cdot)$ is the Gaussian distribution. Denote the training features $X = \{x_n\}_{n=1}^{N}$, and $Z = \{z_n\}_{n=1}^{N}$ is the set of all labels. The main problem is to compute the posterior $p(z_n|X, \theta)$ over the labels and the predictive density $p(x|X, \theta)$, which expression is:

$$p(x|X, \theta) = \int_{H,V} p(x|H, V) \int_{Z} p(W|X, \theta), \tag{4.10}$$

where $\theta = \{\alpha, \lambda\}$ is the hyperparameters from prior. $W = \{H, V, Z\}$ is the set of all latent variables of the DP mixture. Since $p(W|X, \theta)$ is difficult to compute analytically, a parametric family of variational distribution $q(W; \phi)$ is utilized to approximate the posterior [130].

$$q(W;\phi) = \prod_{i=1}^{L}[q_{v_i}(v_i;\phi_i^v)q_{\eta_i}(\eta_i;\phi_i^\eta)]\prod_{i=1}^{L} q_{z_n}(z_n), \tag{4.11}$$

where $q_{v_i}(v_i;\phi_i^v)$ and $q_{\eta_i}(\eta_i;\phi_i^\eta)$ are parametric models with parameters ϕ_i^v and ϕ_i^η, which means one parameter per i. $q_{z_n}(z_n)$ are discrete distributions over the component labels, n means one distribution per n. Blei and Jordan [129] establish a specific truncation level $L \equiv T$ for the variational mixture in equation (4.11) by defining $q_{v_T}(v_T = 1) = 1$ and with the assumption that data observations allocate no responsibility to the components with index beyond the truncation level T, which means $q_{z_n}(z_n > T) = 0$.

Kurihara et al. [130] proposed a variational model for q that facilitates the nesting of families across T. In their theory, the truncation level L is infinity, but the parameters of all models are tied after a specific level $T(T \ll L)$. In addition, they impose extra conditions: $q_{v_i}(v_i;\phi_i^v) = p_v(v_i|\alpha)$ and $q_{\eta_i}(\eta_i;\phi_i^\eta) = p_\eta(\eta_i|\lambda)$. $q_{z_n}(z_n)$ are discrete distributions of the component labels.

The optimal $q_{z_n}(z_n)$ is [130]:

$$q_{z_n}(z_n = i) = \frac{\exp(S_{n,i})}{\sum_{j=1}^{\infty}\exp(S_{n,i})}, \tag{4.12}$$

where

$$S_{n,i} = E_{q_V}[\log p_z(z_n = i|V)] + E_{q_{\eta_i}}[\log p_x(x_n|\eta_i)], \tag{4.13}$$

In this case, the probabilities $q_{z_n}(z_n = i)$ computed using:

$$E_{q_{v_i}}[\log v_i] = \Psi(\phi_{i,1}^v) - \Psi(\phi_{i,1}^v + \phi_{i,2}^v), \tag{4.14}$$

$$E_{q_{v_i}}[\log(1 - v_j)] = \Psi(\phi_{i,2}^v) - \Psi(\phi_{i,1}^v + \phi_{i,2}^v), \tag{4.15}$$

$$E_{q_{\eta_i}}[\log p_x(x_n|\eta_i)] = E_{q_{\eta_i}}[\eta_i]^T x_n - E_{q_{\eta_i}}[a(\eta_i)], \tag{4.16}$$

where $\Psi(\cdot)$ is the digamma function, and the parameter can be found to be

$$\phi^v_{i,1} = \alpha_1 + \sum_{n=1}^{N} q_{z_n}(z_n = i) \tag{4.17}$$

$$\phi^\eta_{i,1} = \lambda_1 + \sum_{n=1}^{N} q_{z_n}(z_n = i)x_n \tag{4.18}$$

$$\phi^v_{i,2} = \alpha_2 + \sum_{n=1}^{N} \sum_{j=i+1}^{\infty} q_{z_n}(z_n = i) \tag{4.19}$$

$$\phi^\eta_{i,2} = \lambda_2 + \sum_{n=1}^{N} q_{z_n}(z_n = i) \tag{4.20}$$

The necessary posterior across data labels can be estimated by $p(x|X,\theta) \approx q_{z_n}(z_n)$, using the variational $q(W)$ as an approximation to the genuine posterior $p(W|X,\theta)$. In practice, it is enough to apply the individual $q_{z_n}(z_n = i)$ for the finite part $i \leq T$ and the cumulative $q_{z_n}(z_n > T)$ for the infinite part, even though $q_{z_n}(z_n)$ has infinite support. Lastly, applying the identity $\sum_{i=1}^{\infty} \pi_i(V) = 1$ and the parameter-tying assumption for $i > T$, the predictive probability $p(x|X,\theta)$ can be be approximated by:

$$p(x|X,\theta) = \sum_{i=1} E_{q_V}[\pi_i(V)]E_{q_{\eta_i}}[p_x(x|\eta_i)] + [1 - \sum_{i=1} E_{q_V}[\pi_i(V)]]E_{p_\eta}[p_x(x|\eta)] \tag{4.21}$$

4.1.5 Decision of Sleep Apnoea Application

This research example uses two Gaussian mixture models to model the feature distributions of the 'apnoea' and 'normal' segments. Therefore, two sets of features are given: $X^1 = \{x^1_i\}_{i=1}^{N^1}$ as 'apnoea' and $X^0 = \{x^0_i\}_{i=1}^{N^0}$ as 'normal'. The LLR is then computed for each segment by taking the difference between the log-likelihoods of the data under the 'apnoea' model and the 'normal' model based on the probability computation by equation (4.21). Then the decision [40] will be equation (4.22).

$$log \frac{p(x'|X^1)}{p(x'|X^0)} \geq c, \tag{4.22}$$

where c is a threshold that affects how well sensitivity and specificity are balanced. The whole framework is shown in figure 4.2.

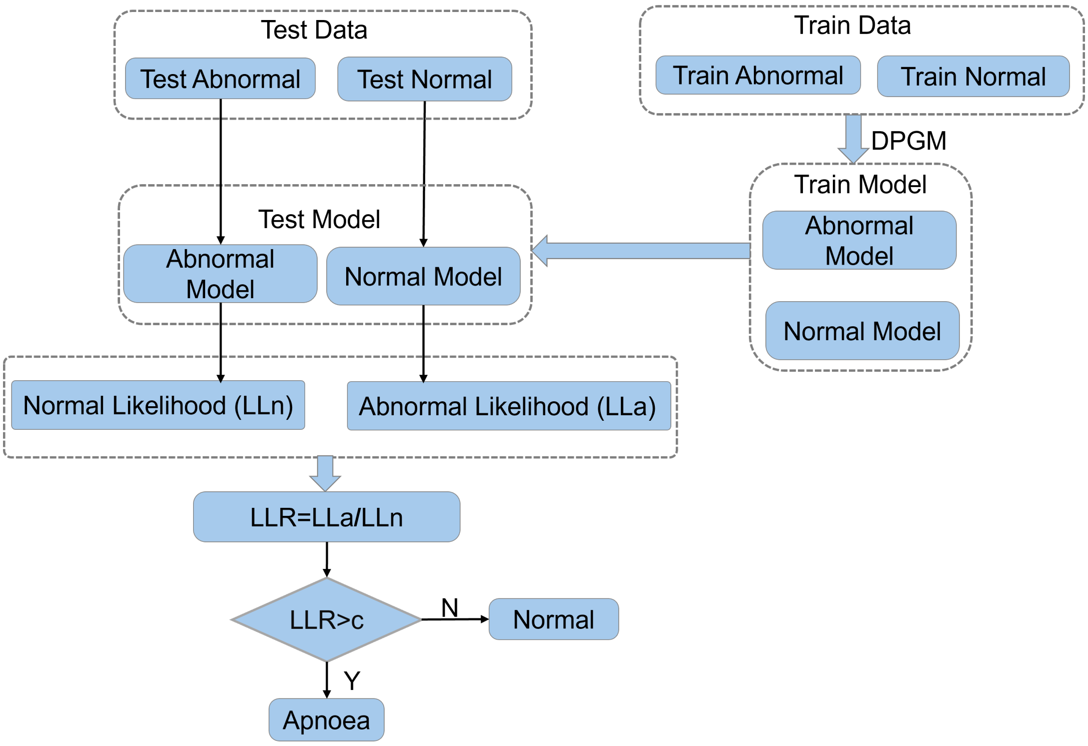


Fig. 4.2 Framework for DPGMM to detect sleep apnoea

4.2 RUSBoost Algorithm for Imbalanced Data Classification

The RUSBoost [132] classification algorithm combines two methods: RUS and boosting techniques. This method mainly solves the class imbalance problem in machine learning tasks. This happens when the data categories are unevenly skewed or biased. For example, in this study, it is necessary to train the classification of the "apnoea" category. However, this category is a minority in the overall data, which may cause bias in the training model and negatively affect its performance. This section mainly introduces the mathematical theory of RUSBoost [132].

First, assume all the features $X = \{x_i\}$ and its corresponding labels $Y = \{y_i\}$, so the dataset can be express as $S = (x_i, y_i)$, where $y_i \in \{0, 1\}$ and $i = 1, 2, 3..., m$. In this study, 0 is defined as a 'normal' class, and 1 is a 'apnoea' class. 'normal' is the category with a larger proportion. Set D_t as the weight of each weak learner and $D_1(i) = \frac{1}{m}$, which means that the weight of each instance is equal at the beginning of the model training. t here is the iteration value and $t = 1, 2, 3, ..., T$. In the RUSBoost framework, a weak learner is a simple classifier that performs only slightly better than random guessing. In practice, decision stumps (shallow decision trees of depth one) are most commonly used due to their efficiency and suitability for boosting. Then, the 'normal' samples are randomly eliminated to modify the class distribution in the training set until the dataset reaches the necessary balance between classes, thus obtaining a temporary training dataset $S_t^{'}$ and new weight $D_t^{'}$. This is the meaning of RUS. Next, the temporary dataset $S_t^{'}$ and its corresponding weights $D_t^{'}$ are fed into a base learning model ($WeakLearn$), which trains a weak hypothesis $h_t(x_i) = \{p_{y=0}(i), p_{y=1}(i)\}$. p is the predicted probabilities of x_i from weak learner. After this, the pseudo-loss ϵ_t which measures the error of h_t weighted by D_t can be calculated as:

$$\begin{aligned} \epsilon_t &= \sum_{(i,y):y_i \neq y} D_t(i)(1 - h_t(x_i, y_i) + h_t(x_i, y)), \\ &= \sum_{(i,y):y_i \neq y} D_t(i)(1 - p_{y=0}(i) + p_{y=1}(i)) \end{aligned} \tag{4.23}$$

This error can be used to compute the weight update factor $\alpha_t = \frac{\epsilon_t}{1-\epsilon_t}$, which adjusts the influence of h_t in the final model. Subsequently, weights D_{t+1} are updated.

$$D_{t+1}(i) = D_t(i)\alpha_t^{\frac{1}{2}(1+p_{y=0}(i)+p_{y=1}(i))} \tag{4.24}$$

Equation (4.24) shows that the weights assigned to misclassified instances increase, while those assigned to correctly classified instances decrease. This adjustment allows the model to focus on more challenging cases in future iterations. At the beginning of the next iteration, the weights D_{t+1} will be normalized as

$$D_{t+1}(i) = \frac{D_{t+1}(i)}{Z_t}, \tag{4.25}$$

where $Z_t = \sum\limits_{i} D_{t+1}(i)$. After T iterations, the algorithm forms the final hypothesis, denoted by $H(x)$, which is constructed by combining the T weak hypotheses through weighted voting. The weight of each vote corresponds to the accuracy of the hypothesis, which is inversely proportional to α_t. The expression of $H(x)$ is shown in equation (4.26).

$$H(x) = \arg\max_{y \in Y} \sum_{t=1}^{T} h_t(x, y) log \frac{1}{\alpha_t} \tag{4.26}$$

By repeatedly concentrating on fixing its prior errors, this approach ensures that the machine learning algorithm gradually enhances its capacity to categorise increasingly challenging, frequently minority, samples, efficiently managing the imbalance in the dataset.

4.3 Experimental Results and Discussion

4.3.1 Initial Conditions of the Algorithms

The RUSBoost model training set has 100 weak learners, 1000 ensemble learning cycles, and a learning rate of 0.1. All DPGMM settings are default. The three publicly available databases are trained and tested independently with 10-fold cross-validation. All training and testing programs are run on Matlab 2023b and above.

4.3.2 Datasets

To enable near-real-time detection of SDB events, the SPO2 and pulse signal are systematically partitioned into overlapping subsequences, each of which is the same length. Multiple informative features are extracted from each subsequence for advanced analysis. The following subsections will clarify the source of the experimental datasets and the specific methodologies applied for their preprocessing. Moreover, the features used in the classification process, which are defined within both the time and wavelet domains, will be explained.

4.3.3 Features Visualisation

To evaluate the classification ability of the 18 extracted features, we generated 18 normalised histograms (see Figure 4.3) and box plots (see Figure 4.4) for the normal class and the apnoea class, respectively. Histograms show varying degrees of classification ability regarding shape, central tendency, and skewness [133]. The box plot further reveals the median, interquartile range (IQR) and outliers of the features in different categories from a statistical perspective [134]. This section takes SPO2 in the St. Vincent dataset as an example to show the distribution of each feature in the two categories. Table 4.1 gives the specific meaning of each feature name. The histograms from figure 4.3 show that several features

Table 4.1 Explanation of feature name

varad	Variation of level 4 detail coefficients
varbd	Variation of level 5 detail coefficients
rangead	Range of level 4 detail coefficients
rangebd	Range of level 5 detail coefficients
Powerad	Mean Energy of level 4 detail coefficients
Powerbd	Mean Energy of level 5 detail coefficients
maxad	Maximum level 4 detail coefficients
maxbd	Maximum level 5 detail coefficients
kursig	Kurtosis of signal
meansig	Mean of signal
varsig	Variation of signal
rangesig	Signal Range
CTM	Central Tendency measure of signal
minsig	Minimum signal
shannonEn	Shannon entropy of signal
TsallisEn	Tsallis entropy of signal
WavL2Num	Number of the large coefficients (level 4 detail coefficients)
WavL3Num	Number of the large coefficients (level 5 detail coefficients)

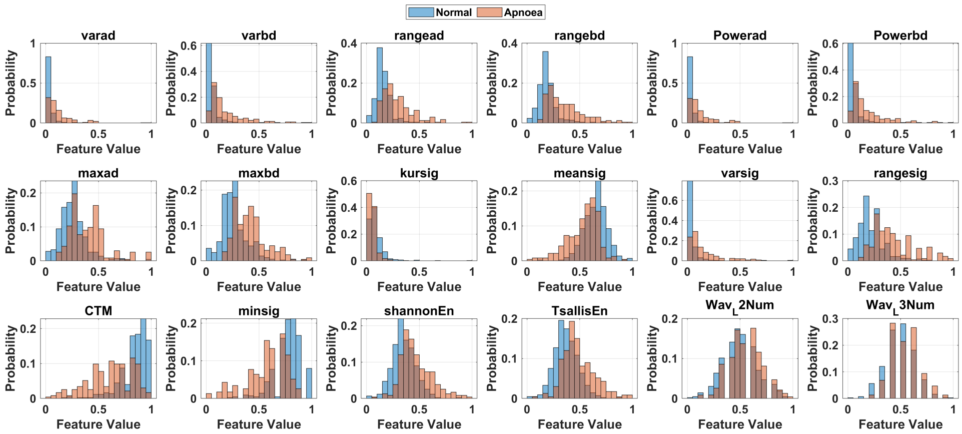


Fig. 4.3 Features visualisation by histogram

exhibit highly asymmetric distributions, including varad, varbd, rangebd, and Powerbd. There is a significant shift in the peak positions between the two classes. For example, the normal samples exhibit a sharp peak close to zero, while the apnoea samples exhibit a wider right-skewed distribution, indicating that the variability and power-related measurements in the apnoea event are higher, which means that the SPO2 signal fluctuates more violently in the apnoea state, and abnormal oscillations or signal disturbances occur. Several features, including CTM, minsig, maxad, and rangesig, exhibit a significantly skewed distribution in the normal class with a single peak. In contrast, the distribution is wider and has two peaks in the apnoea class. Besides, they are distributed on both sides with less overlap, indicating that these features can effectively distinguish between normal and apnoea. In addition to practical distinguishing features, some features with similar distributions, such as ShannonEn, TsallisEn, WavL2Num, and WavL3Num, have large overlapping areas, indicating insufficient classification ability. However, the differences in the peak positions of the distributions allow these features to serve as auxiliary features for classification.

The histogram illustrates the distribution of features, while the box plot visually indicates the classification suitability of features. The blue box in Figure 4.4 is the IQR. Features such as Powerbd, rangebd, and rangesig show non-overlapping IQR between classes, while the apnoea group consistently shows higher medians and wider distributions. CTM and Meansig also show a significant

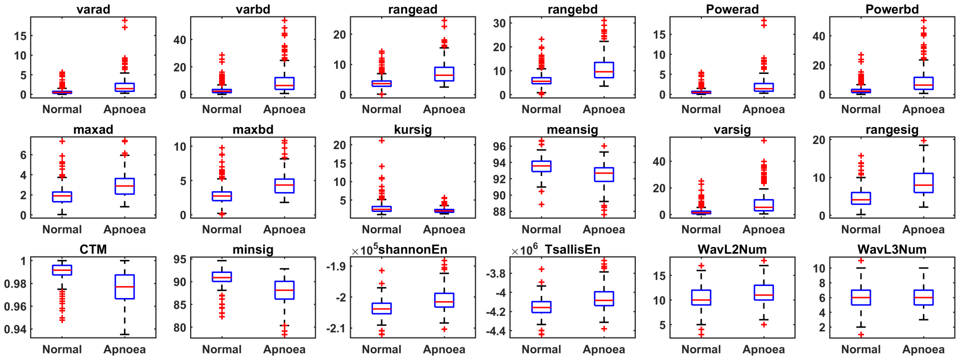


Fig. 4.4 Features visualisation by boxplot

downward shift in the apnoea class, reflecting the loss of signal consistency during apnoea episodes. Non-overlapping IQR mean that the feature has very strong classification ability. In contrast, features such as WavL2Num and WavL3Num have overlapping medians and IQR, which means limited classification power when used alone. This is the same conclusion as the histogram.

As observed from Figure 4.3 and Figure 4.4, features rangebd, powerbd, maxad, rangesig, CTM, and minsig demonstrate strong discriminative capability. However, this does not imply that the remaining features are uninformative. Table 4.2 reports the classification performance obtained using either the whole feature set or only the aforementioned six features. The fact that feature selection does not improve classification accuracy suggests that the excluded features contribute complementary information, enhancing overall model performance.

Table 4.2 Results of StVincent Database (SPO2) Based on Different Features Selected

	acc	sens	spec	F_1-N	F_1-A	κ
After feature select	74.76%	47.91%	81.14%	0.839	0.422	0.263
All feature select	84.32%	72.07%	87.12%	0.90	0.67	0.57

4.3.4 Comparison of Different Database

This study compared the detection results of three different public datasets. The first is the Childhood Adenotonsillectomy Trial (CHAT) [2] from the National Sleep Research Resource (NSRR) [21], and the second is from St. Vincent's

University Hospital [3]. The third one is the Apnea-ECG Database [1]. The whole dataset and the data used for the experiment are shown in table 4.3. In the names on the left side of the table, ‘60’ refers to a 60-second segment, while ‘10,’ ‘20,’ and ‘30’ represent different overlaps.

Table 4.3 Comparison of two different datasets

	CHAT			StVincent			ECG-Apnoea		
	label	Count	percentage	label	Count	percentage	label	Count	percentage
60_10_ovlp	0	255906	93.38%	0	9094	75.94%	0	2044	60.89%
	1	18141	6.62%	1	2882	24.06%	1	1313	39.11%
60_20_ovlp	0	323690	94.50%	0	11674	78.02%	0	2036	60.99%
	1	18824	5.50%	1	3289	21.98%	1	1302	39.01%
60_30_ovlp	0	437639	95.81%	0	16137	80.86%	0	2030	61.29%
	1	19159	4.19%	1	3820	19.14%	1	1282	38.71%

In these datasets, the StVincent dataset samples at 8Hz and the Apnoea-ECG at 100Hz, while the CHAT dataset has a variable sampling frequency. Figure 4.5 provides a statistical chart showing the number of files corresponding to each sampling frequency.

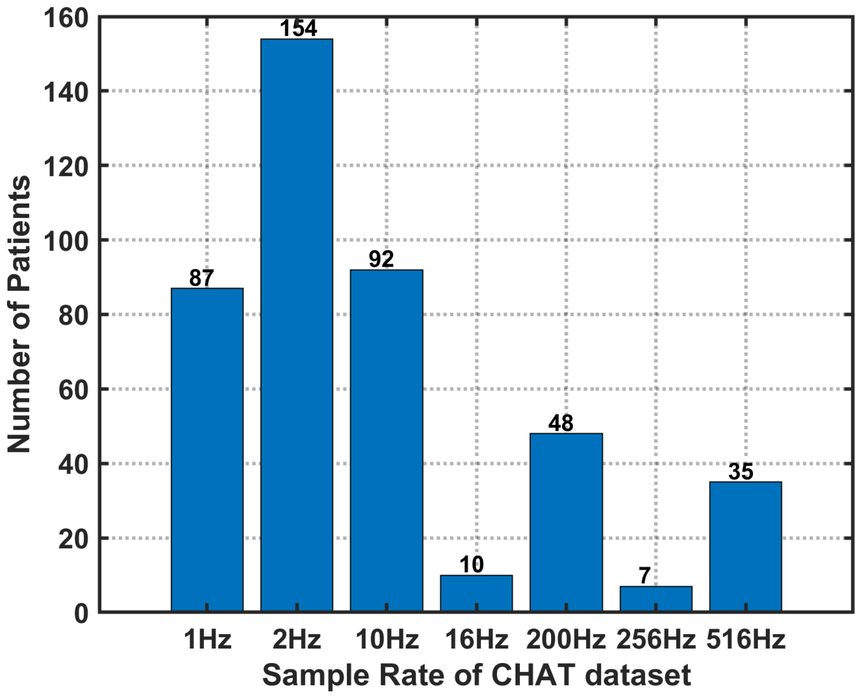


Fig. 4.5 Histogram of sampling frequency of patient data from CHAT dataset

4.3.5 Data Processing

Figure 4.5 illustrates the non-uniform sampling frequencies in the CHAT dataset, with a high count of files at 1 Hz and 2 Hz. Low sampling rates can lead to data being classified as noise, impacting further analyses, while very high rates may result in redundant information. Consequently, we excluded files with low (1 Hz and 2 Hz) and high (512 Hz) sampling frequencies from the CHAT dataset.

All data will be segmented into overlapping one-minute intervals for analysis to detect sleep-disordered breathing (SDB) events. Segments containing SDB events are classified as 'apnoea', while those without respiratory disturbances are labelled 'normal'. A nuanced classification is applied when an apnoea event spans two consecutive segments. If respiratory disturbances occur but last less than five seconds in any segment, it is labelled 'normal', as such brief fluctuations are unlikely to cause clinically significant desaturation or arousal and do not substantially alter the overall respiratory pattern. According to the AASM [17], apnoea events are defined as lasting at least 10 seconds. Therefore, shorter disturbances are unlikely to cause meaningful oxygen desaturation or arousal. Setting the cut-off at five seconds helps avoid misclassifying minor signal noise or brief irregularities as pathological events, while still maintaining sensitivity to clinically relevant disturbances. Conversely, if disturbances exceed this duration in any segment, it is marked as 'apnoea', indicating a significant disruption in respiratory activity. This approach ensures precise categorization of each minute based on the severity and duration of disruptions.

Poor contact with the pulse oximeter, often due to body movement at night, can generate artefacts. SPO2 and pulse readings below 50 and pulse readings above 300 are considered artefacts. Segments with these values are excluded from training and testing to ensure the reliability of the data.

4.3.6 Results with Single Data

This experiment evaluated and contrasted multiple algorithms using different training sets to examine the effects of data imbalance on training outcomes and to offer algorithmic recommendations for the medical field. Tables 4.4, 4.5, and 4.6 present the results of several machine learning techniques employing the identical wavelet (db1), segment duration (30-second overlap), and features. In this context, ‘F1-N’ represents the F1 score for the normal category classification, while ‘F1-A’ signifies the F1 score for the apnoea category classification. RUSBoost performs well on the StVincent dataset, attaining an acceptable F1-A score and high sensitivity. The significant imbalance in the CHAT dataset leads to an F1-A score of less than 0.5. The algorithms demonstrate exceptional performance on the ECG dataset, with a maximum accuracy of 97%. In conclusion, if the ratio of the ‘apnoea’ moments to the ‘normal’ moment is less than 1:3, the RUSBoost algorithm is advisable. For ratios over 1:3, the performance of different algorithms remains consistent, with SVM or Gaussian Naive Bayes being advisable.

Table 4.4 Results of ECG Database (SPO2) Based on Different Machine Learning Methods with the Same Wavelet, Segment Length, and Features

db1_30_ovlp	acc	sens	spec	F1-N	F1-A	κ
Fine Tree	95.16%	95.86%	94.71%	0.960	0.939	0.899
Gaussian Naive Bayes	**97.31%**	**96.55%**	**97.80%**	**0.978**	**0.966**	**0.943**
Linear SVM	97.31%	95.17%	98.68%	0.978	0.965	0.943
Fine KNN	95.97%	93.79%	97.36%	0.967	0.948	0.915
Boosted Tree	96.77%	95.17%	97.80%	0.974	0.958	0.932
Bagged Tree	96.24%	93.79%	97.80%	0.969	0.951	0.920
Medium Neural Network	94.89%	91.72%	96.92%	0.959	0.933	0.892
RUSBoost	88.95%	75.47%	89.54%	0.94	0.36	0.32
DPGMM	94.20%	97.90%	91.78%	0.95	0.93	0.88

Table 4.5 Results of StVincent Database (SPO2) Based on Different Machine Learning Methods with the Same Wavelet, Segment Length, and Features

db1_30_ovlp	acc	sens	spec	F1-N	F1-A	κ
Fine Tree	86.87%	58.38%	93.61%	0.920	0.630	0.551
Gaussian Naive Bayes	86.89%	52.62%	94.99%	0.921	0.605	0.529
Linear SVM	87.42%	46.07%	97.21%	0.926	0.584	0.516
Fine KNN	82.21%	54.97%	88.65%	0.890	0.542	0.432
Boosted Tree	87.77%	56.02%	95.29%	0.926	0.637	0.565
Bagged Tree	87.37%	56.02%	94.79%	0.924	0.629	0.555
Medium Neural Network	87.37%	56.02%	94.79%	0.924	0.629	0.555
RUSBoost	**85.86%**	**80.37%**	**87.17%**	**0.91**	**0.69**	**0.60**
DPGMM	84.51%	42.15%	94.54%	0.91	0.51	0.42

Table 4.6 Results of CHAT Database (SPO2) Based on Different Machine Learning Methods with the Same Wavelet, Segment Length, and Features

db1_30_ovlp	acc	sens	spec	F1-N	F1-A	κ
Fine Tree	95.90%	9.34%	99.69%	0.979	0.160	0.150
Gaussian Naive Bayes	92.30%	28.18%	95.10%	0.959	0.235	0.195
Linear SVM	95.81%	0.00%	100.00%	0.979	NaN	0.000
Fine KNN	93.78%	24.95	96.79	0.968	0.252	0.219
Boosted Tree	95.89%	8.09%	99.73%	0.979	0.142	0.133
Bagged Tree	95.90%	14.98%	99.44%	0.979	0.234	0.220
Medium Neural Network	95.96%	12.27%	99.62%	0.979	0.203	0.191
RUSBoost	**88.95%**	**75.47%**	**89.54%**	**0.94**	**0.36**	**0.32**
DPGMM	61.61%	85.42%	60.54%	0.75	0.16	0.09

Section 2.5.1 introduces an innovative feature termed ‘Number of Large Coefficients’. This study aimed to validate the utility of this feature in model training. Employing the pulse signal from the St. Vincent dataset as a case study, models were systematically trained under various conditions: with and without resampling and incorporating the new feature. The results presented in Table 4.7 confirm the initial hypothesis that models trained without resampling attained greater accuracy. The accuracy of models trained without resampling was 2% to 3% higher than that of models trained with resampling. Furthermore, the accuracy of the model in disease identification was improved by approximately 2% as a result of the new feature’s integration, which also increased training sensitivity.

Table 4.7 Results of St. Vincent’s University Hospital Sleep Apnoea Database Based on Different Features in Same Wavelet and Segment Length

db4_30_ovlp	acc	sens	spec	F1-N	F1-A	κ
no sample	73.93%	62.83%	76.57%	0.83	0.48	0.32
sample	71.24%	47.38%	76.91%	0.81	0.39	0.21
no resample, add new feature	**74.24%**	**64.92%**	**76.44%**	**0.83**	**0.49**	**0.33**
resample, add new feature	71.19%	47.38%	76.85%	0.81	0.39	0.21

In addition to comparing algorithms and datasets, this experiment further assesses the efficacy of different wavelet features in training models. The training results, employing the CHAT dataset for illustration, are detailed in Tables 4.8 and 4.9. The left column of each table specifies the wavelet type and segment length; for example, 'db1_10_ovlp' denotes the use of wavelet db1 with a 10-second overlap. According to the results, pulse data performs better with db3, whereas SPO2 data is best handled among the four wavelets under investigation with db1. These presentations emphasise the need to select specific wavelet features for different types of data in order to improve the efficacy of the model. Moreover, the best results are always obtained with a 30-second overlap when using identical wavelet settings.

Table 4.8 Results of CHAT Database (Pulse) Based on Different Wavelet and Segment Length.

	acc	sens	spec	F1-N	F1-A	κ
db1_10_ovlp	67.69%	69.15%	67.58%	0.80	0.22	0.12
db1_20_ovlp	69.64%	72.14%	69.50%	0.81	0.21	0.13
db1_30_ovlp	72.34%	73.50%	72.29%	0.83	0.18	0.12
db2_10_ovlp	68.18%	73.90%	67.78%	0.80	0.23	0.14
db2_20_ovlp	70.38%	72.29%	70.27%	0.82	0.21	0.13
db2_30_ovlp	72.19%	75.21%	72.05%	0.83	0.18	0.12
db3_10_ovlp	67.94%	72.84%	67.59%	0.80	0.23	0.13
db3_20_ovlp	70.58%	72.87%	70.45%	0.82	0.21	0.13
db3_30_ovlp	**72.41%**	**74.36%**	**72.33%**	**0.83**	**0.18**	**0.12**
db4_10_ovlp	67.94%	73.14%	67.57%	0.80	0.23	0.13
db4_20_ovlp	70.77%	72.43%	70.67%	0.82	0.21	0.13
db4_30_ovlp	72.49%	74.93%	72.38%	0.83	0.19	0.12

4.3.7 Results Based on Fusion Technique

In this experiment, three types of early feature fusion, as described in Section 2.7.1, were employed. Using the CHAT database as an example for concatenation fusion, the performance results of the RUSBoost algorithm are presented in Table 4.10. From the table, wavelet db3 and 30-second overlap give the best performance and employing a 30-second overlap in this experiment produces favourable results. Among the assessed wavelets, db3 exhibited the best efficacy throughout the investigation. Table 4.11 compares the training results obtained from the StVincent and CHAT databases using the same feature processing method. The CHAT database demonstrates superior overall accuracy, whereas the StVincent

Table 4.9 Results of CHAT Database (SPO2) Based on Different Wavelet and Segment Length.

	acc	sens	spec	F1-N	F1-A	κ
db1_10_ovlp	83.67%	65.65%	84.95%	0.91	0.35	0.28
db1_20_ovlp	84.93%	69.23%	85.85%	0.91	0.34	0.28
db1_30_ovlp	**89.97%**	**74.79%**	**90.63%**	**0.95**	**0.38**	**0.34**
db2_10_ovlp	83.98%	64.19%	85.37%	0.91	0.34	0.28
db2_20_ovlp	85.16%	66.13%	86.25%	0.92	0.33	0.27
db2_30_ovlp	89.82%	75.50%	90.44%	0.94	0.38	0.34
db3_10_ovlp	84.21%	64.19%	85.62%	0.91	0.35	0.28
db3_20_ovlp	85.17%	66.86%	86.23%	0.92	0.33	0.27
db3_30_ovlp	89.88%	75.07%	90.53%	0.94	0.38	0.34
db4_10_ovlp	84.04%	64.64%	85.40%	0.91	0.35	0.28
db4_20_ovlp	85.15%	66.86%	86.20%	0.92	0.33	0.27
db4_30_ovlp	89.89%	75.36%	90.53%	0.94	0.38	0.34

Table 4.10 Results of CHAT Database in Concatenation Features

	acc	sens	spec	F1-N	F1-A	κ
db1_10_ovlp	88.25%	58.51%	90.35%	0.93	0.40	0.34
db1_20_ovlp	86.64%	68.69%	87.69%	0.93	0.36	0.31
db1_30_ovlp	91.90%	69.91%	92.86%	0.96	0.42	0.38
db2_10_ovlp	87.72%	58.57%	89.77%	0.93	0.39	0.33
db2_20_ovlp	88.75%	60.12%	90.39%	0.94	0.37	0.32
db2_30_ovlp	91.78%	70.20%	92.72%	0.96	0.42	0.38
db3_10_ovlp	87.54%	57.36%	89.66%	0.93	0.38	0.32
db3_20_ovlp	88.82%	61.00%	90.42%	0.94	0.37	0.32
db3_30_ovlp	**91.54%**	**72.06%**	**92.39%**	**0.95**	**0.42**	**0.38**
db4_10_ovlp	87.62%	55.99%	89.84%	0.93	0.37	0.31
db4_20_ovlp	89.33%	60.56%	90.98%	0.94	0.38	0.33
db4_30_ovlp	91.39%	72.06%	92.23%	0.95	0.41	0.37

database exhibits elevated F1 scores and κ values. This is because the different imbalance ratios of the two data sets can impact model fit significantly. Figure 4.6 illustrates the associated ROC curve, demonstrating an AUC value of 0.923.

The average performance indicators for each training dataset were calculated to assess the impact of feature fusion relative to non-fused features. The CHAT database exhibits an average accuracy of 86.3% for single signal classification, whereas fusion classification demonstrates an enhancement of 89.3% accuracy.

Table 4.11 Results of Different Database with the Processing Features

	acc	sens	spec	F1-N	F1-A	κ
CHAT	91.54%	72.06%	92.39%	0.95	0.42	0.38
StVincent	86.72%	74.87%	89.52%	0.92	0.68	0.60

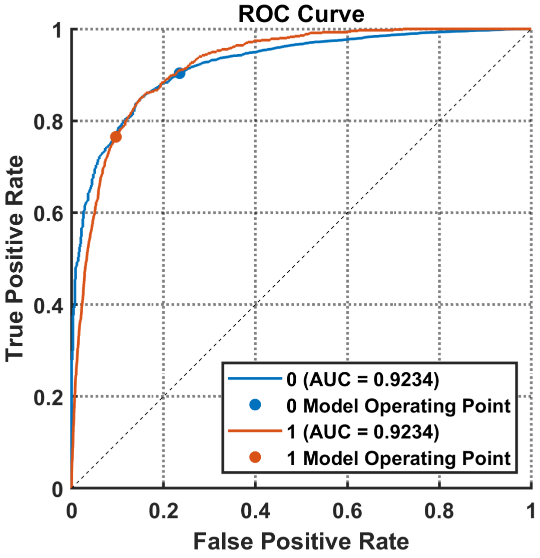


Fig. 4.6 RUSBoost ROC curve of CHAT Database based on db3 and 30s overlap and fusion technique

Table 4.12 Mean Performance Based on 157 Patients of Single Signal and Concatenation Feature Fusion

		acc	sens	spec	F1-N	F1-A	κ
CHAT	SPO2	86.32%	69.04%	87.33%	0.92	0.35	0.30
	Pulse	70.21%	**73.06%**	70.04%	0.82	0.21	0.13
	Fusion	**89.27%**	63.75%	**90.78%**	**0.94**	**0.39**	**0.34**
St Vincent	SPO2	84.32%	72.07%	87.12%	0.90	0.67	0.57
	Pulse	72.79%	60.47%	76.27%	0.81	0.49	0.31
	Fusion	**85.80%**	**74.45%**	**89.61%**	**0.91**	**0.69**	**0.60**

In the St. Vincent database, single signal detection achieves 84.3% accuracy, while feature fusion classification enhances accuracy to 85.8%. This means that, even in instances of mild data imbalance, specificity, the metric for accurately identifying disease-free segments, is enhanced by 1.5%. In cases of substantial data imbalance, RUSBoost accuracy with integrated features is enhanced by 3% compared to single-signal features. Furthermore, data from Table 4.12 indicates that training data post-feature fusion fits more effectively with the model, as demonstrated by elevated F1 scores and κ values.

4.4 Summary

This chapter discusses a framework to detect sleep apnoea using pulse and SpO2 data based on machine learning. This framework aims to provide an effective alternative to expensive and time-consuming PSG. The suggested approach is very reliable and can be used in many situations, which makes it perfect for tracking sleep at home. Furthermore, unlike PSG, which requires multiple sensors to be monitored all night, this approach only requires a simple pulse oximeter for non-invasive detection, which makes the patient more comfortable and saves money.

This study uses the wavelet transform to capture valuable information from sleep signals and the RUSBoost algorithm to simplify classifying unbalanced data. Three public datasets are used in this study: the apnoea-ECG, CHAT, and St Vincent datasets. The CHAT dataset has a highly imbalanced apnoea-to-normal ratio of 1:15, similar to the sleep apnoea situations in children. To evaluate this framework, this study compared multiple machine learning methods, including SVM, KNN, and DPGMM. The study found that RUSBoost performs best when the ratio of apnoea events to normal events exceeds 1:3, which means that RUSBoost becomes the most suitable choice for processing unbalanced sleep apnoea data. In addition, the experiment also compared the effects of single signal and feature fusion on binary classification. Feature fusion can improve detection accuracy by about 3% compared to using either signal alone. This highlights the advantages of feature fusion and enhances the model's ability to distinguish between normal and apnoea events. The study also investigated the impact of different wavelet transform types, especially Daubechies wavelets (db1–db4), to determine the most compelling feature extraction method. The results indicated that the db3 wavelet was optimal for pulse signal processing, and the db1 wavelet was best suited for SpO2 data, highlighting the significance of selecting the appropriate wavelet for various physiological signals.

Before the experiment, sleep data was divided into overlapping one-minute segments to capture significant changes in sleep patterns in real time accurately. The experiment used 10-fold cross-validation and evaluated the performance through evaluation indicators such as accuracy, sensitivity, specificity, F1 value, Cohen's kappa, and AUC. It has an AUC of 0.923, a maximum accuracy of 91.54%, a sensitivity of 72.06% and a specificity of 92.39%. In addition, the experiments found that one of the main challenges of automatic sleep apnoea detection is the imbalance of the dataset. Because the frequency of apnoea events is much lower than the normal breathing epoch, this will cause bias in the machine learning model.

To the best of our knowledge, the most recent study on sleep apnoea detection employed a CNN–LSTM model [135] and evaluated it on the ECG dataset, which was also included in our experiments. Their reported accuracy, sensitivity, and specificity were 91.50%, 94.37%, and 86.89%, respectively. In comparison, our implementation of RUSBoost achieved 88.95%, 75.47%, and 89.54%. Although the sensitivity is lower, RUSBoost is inherently more suitable for handling imbalanced datasets. For instance, Zovko et.al. [136] also addressed imbalanced data, however, their reported accuracy and sensitivity were only 79% and 68%, whereas our method attained 88.95% and 75.47%. This further demonstrates that RUSBOOST is very suitable for the classification of imbalanced data.

In conclusion, this study introduces an effective and practical machine-learning framework for sleep apnoea detection using pulse and blood oxygen data, focusing on overcoming the challenges of imbalanced datasets. The results confirm that the RUSBoost is the most effective classification algorithm for highly imbalanced sleep apnoea data. Feature fusion can improve detection accuracy. The proposed framework achieves state-of-the-art performance in sleep apnoea classification and enables widespread non-invasive home screening. Future studies could incorporate other physiological signals (such as ECG and respiratory effort) to improve diagnostic accuracy.

Chapter 5

A Convolutional Neural Network for Apnoea Detection

Machine learning requires manual feature extraction, which increases computational costs. Deep learning can make up for this. This Chapter proposes a deep learning approach for contactless detection of sleep apnoea using pulse and SPO2 data. Three convolutional neural network architectures are adopted for apnoea classification purposes by fusing different features of the available time series signals. A CNN model, a CNN-SVM model, and a CNN-RNN model are compared. The RNN includes GRU and BiGRU. The CNN is utilised to extract features, whilst the SVM and RNN are used for classification. In addition, we compare two different fusion methods, signal-level and feature-level fusion. The performance is validated and evaluated on a public dataset St. Vincent University Hospital. The results show that the concatenation of SPO2 and pulse signal at the signal level enhances the classification performance compared to using the individual signal. In addition, the classification sensitivity with signal-level fusion is higher than that with feature-level fusion. Overall, the proposed CNN-GRU architecture gives the best performance with an accuracy of 85.4%, a sensitivity of 61.5%, a specificity of 91.9%, an F_1 score of 0.64, and a κ score of 0.551 with a dropout rate of 0.5 and a 20-second overlap. The results demonstrate that the proposed deep learning approach offers a promising solution for non-invasive detection of sleep apnoea using affordable physiological signals.

5.1 Advantages and Limitations of Deep Learning

Traditional machine learning methods refer to the classic algorithms for pattern recognition, predictive modelling, and decision-making tasks. These methods include supervised learning, unsupervised learning, and reinforcement learning methods. Although machine learning techniques have been successful in many fields, they have significant limitations [137] when dealing with complex, high-dimensional data. First, these methods rely heavily on structured data and require a lot of feature engineering to achieve optimal performance. The automatic learning ability of the model often depends on features manually extracted by experts. In addition, traditional machine learning algorithms have high computational costs when facing large-scale data. They cannot effectively process unstructured data, such as images and text, resulting in limited model performance.

Deep learning was developed to address these challenges. It minimises reliance on manual feature engineering, effectively analyses image data, and automatically learns features via a layered structure. Its emergence has increased the accuracy and generalisation capability of computer vision tasks. Besides automatically extracting characteristics and lowering human involvement, deep learning techniques, such as long short-term memory networks (LSTM), may efficiently represent long-term dependencies. Furthermore, time series data often contains complex nonlinear relationships. Deep learning techniques, such as convolutional neural network (CNN) and recurrent neural networks (RNN) can perform well when handling highly nonlinear or complicated periodic time series and automatically learn nonlinear relationships in the data.

Although deep learning has dramatically advanced in several areas, multiple limitations remain. First, deep learning relies on massive quantities of labelled data and requires a lot of processing capacity. The intrinsic opacity of the model limits interpretability. Furthermore, it is difficult to adjust to changes in data distribution and is influenced by noisy data. The training duration is too long, and hyperparameter tuning is complex, significantly increasing deployment and optimisation costs. In addition to the above-mentioned concerns, unbalanced data will also impact deep learning models. When the majority class samples in the data set are far more than the minority class, the model tends to learn the majority class features and ignore the minority class. This will lead to a decrease in recall. It may seriously affect key applications such as fraud detection and medical diagnosis. Although it can be optimised using data augmentation, resampling, and loss function adjustment, removing the bias caused by unbalanced data is still difficult. Hence, improving the robustness and optimising its unbalanced data processing method remain significant problems in deep learning applications.

5.2 Deep Learning Architecture

Deep learning architectures are structural designs of neural networks that allow computers to learn intricate patterns and representations from input. Unlike conventional machine learning models that depend on hand feature extraction, deep learning uses several layers of artificial neurons to extract, manipulate, and enhance characteristics automatically. Fields, including computer vision, natural language processing, and autonomous systems, have been transformed by methods including convolutional neural networks (CNNs) and recurrent neural networks (RNNs). Especially useful in processing high-dimensional and unstructured data, these structures describe complex nonlinear interactions using neurone layers with activation functions. Deep learning research and practical applications depend on the particular task, data characteristics, and computational resources accessible, which determines the architectural decision and hence emphasises its relevance. This section mainly introduced three deep learning architectures based on CNN.

5.2.1 Design of the CNN architecture

The CNN model was trained and validated with the extracted blood oxygen signal (SpO2) and pulse signal segments to detect apnoea events. The CNN architecture consists of N convolutional blocks that consecutively process the input data [138], as shown in Figure 5.1. N can take any integer value. In the experiments, values of N=2,3,4 and 5 were tested. However, the results showed little improvement or even degradation as N increased. Therefore, N is chosen 2. Each convolutional block includes five sequential layers: convolution, batch normalisation, activation, max pooling, and dropout. The convolutional layer generates a 3D feature map using a 2D convolution operation [105], which is expressed below:

$$x_i^j[m,n] = \sum_{k=1}^{ksize} \sum_{l=1}^{2} w_i^j[k,l] \cdot a_i[m-k+1, n-l+1] + b_i^j, \tag{5.1}$$

where $i = 1, ..., N$ and $j = 1, ..., M$. N is the number of convolutional blocks. M is the number of filters. x_i^j is the feature map generated in the convolutional block i. w_i^j denotes the weight of the filter. b_i^j denotes bias. a_i is the input to the i^{th} convolutional block. The convolutional layer comprises a set of M 2D filters with a kernel. Since the experiment uses dual and single signals separately, the kernel size will be set to a single column $ksize \times 1$ to meet all task inputs. The batch normalization layer normalised the previously generated feature maps [89]. The rectified linear unit (ReLU) [90] activation function was then applied:

$$ReLU(x_i^j) = max(0, x_i^j), \tag{5.2}$$

where x_i^j is the value of each sample of the feature map. The ReLU activation function is commonly employed in deep architectures to accelerate the training

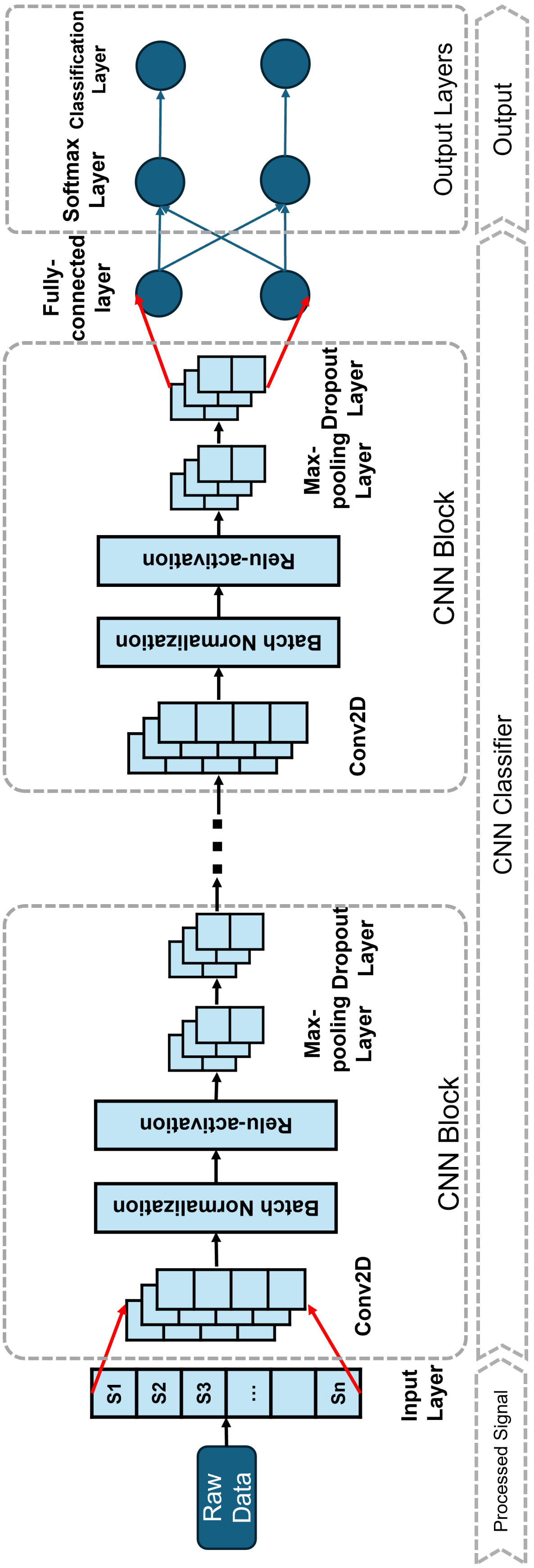


Fig. 5.1 Convolutional Neural Network for Sleep Apnoea Detection

process of CNN without relying on pre-optimized weights[139]. The activation layer is dimensionally reduced using a max pooling layer with a pool factor of 2 × 1. This step reduces the length of the feature map while maintaining both width and depth directions. The last layer of the convolutional block is a dropout layer. The purpose of this layer is to reduce over-fitting. It removes a small part of the activations randomly at each training interval. The drop probabilities usually range from 0 to 0.5 [91].

5.2.2 CNN and SVM framework

A hybrid classification approach, CNN-SVM, combines CNN with SVM. This method [140] uses CNN to extract feature representations from raw input data. The extracted features are then used as input for the SVM classifier. CNN enables the automated extraction of features in a hierarchical structure. As discussed in Chapter 2, Section 2.6.2, the SVM method maximises the margins between various classes to generate appropriate decision limits for classification tasks. The CNN feature extraction combined with SVM classification could raise the model's general performance for different applications. This hybrid model efficiently leverages the representational capabilities of CNNs in conjunction with the robust generalisation properties of SVMs. The CNN-SVM architecture has been successfully applied in several research fields, such as medical imaging, facial recognition remote sensing, and object classification.

Figure 5.2 gives a basic architecture of CNN-SVM, which is also the architecture used in this experiment. Because 2D CNN is used, the processed signal needs to be reshaped to meet the high-dimensional output requirements of CNN. The detailed introduction of the CNN block is in the last section. The output of the fully connected layer of the CNN block will be used as the input of the SVM model. SVM can be changed to any other machine learning method.

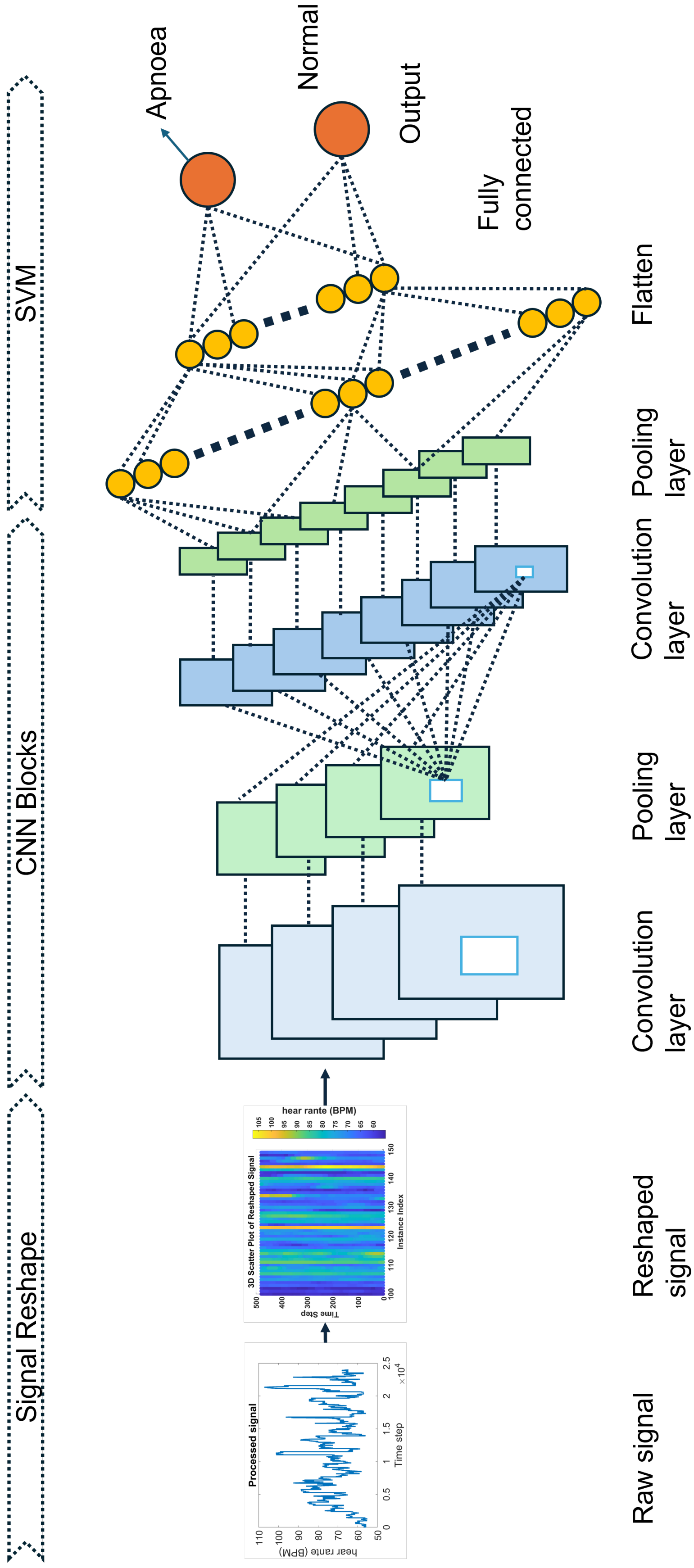
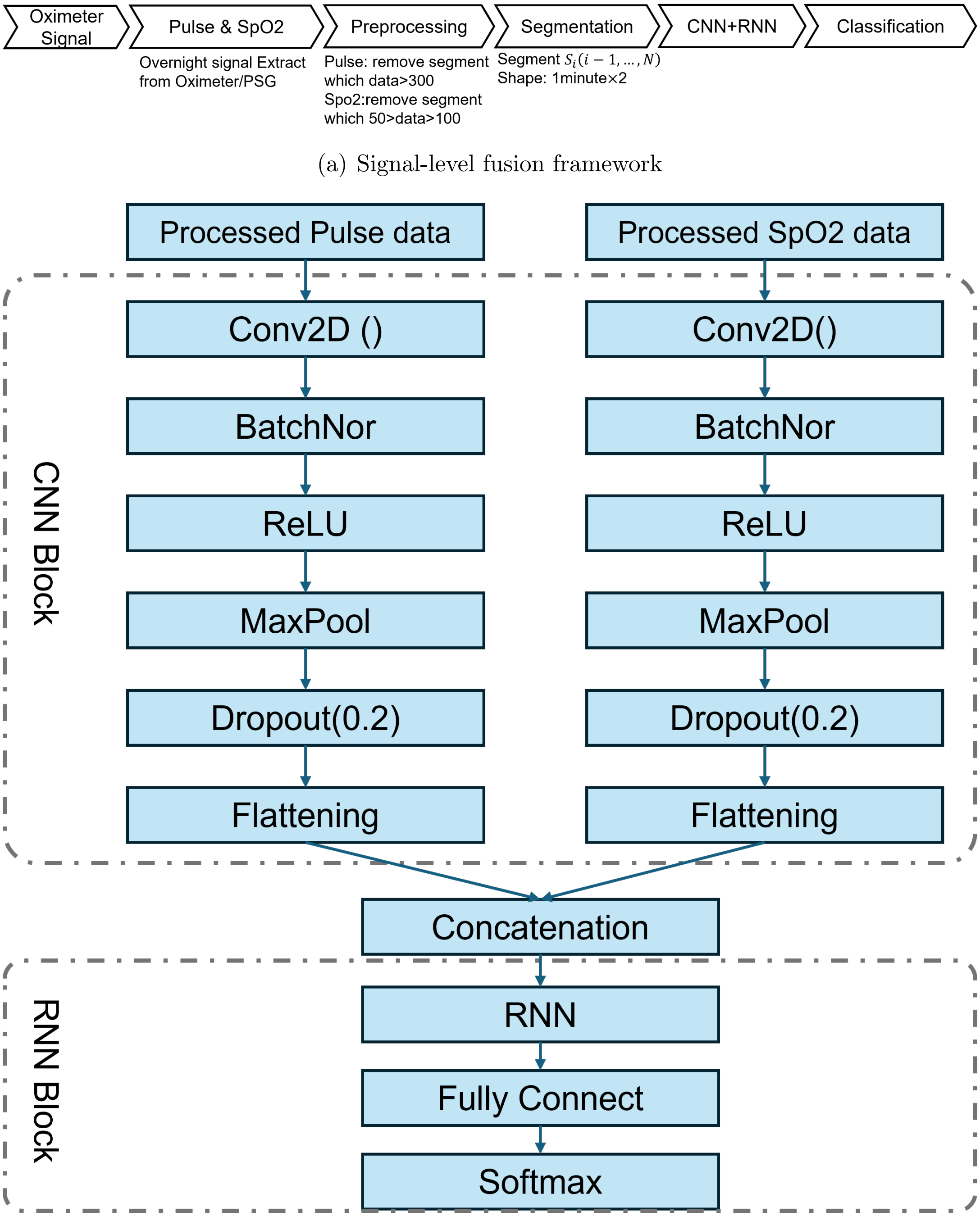


Fig. 5.2 CNN+SVM architecture

5.2.3 CNN and RNN framework

The RNN is a deep learning model that is trained to process and convert sequential data inputs into specific sequential data outputs. Sequential data [141] refers to ordered observations where each element is related to others through temporal or structural dependencies. Examples of sequential data include words, sentences, and time-series data.

A combination of CNN and RNN models is designed and trained using a dataset of 1-minute segments of Pulse and SPO2 signals, labelled with whether it is apnoea. Figure 5.3(a) is a deep learning framework for concatenation in the signal stage and the overall CNN+RNN idea. Both signals are processed similarly to Section 4.3.5 in Chapter 4. The data is divided into 1-minute segments. Assuming the sampling frequency of the data is 8 Hz, the length of a signal is 480. If the concatenation is performed at the signal stage, the input size is 480×2. If concatenation is performed at the CNN feature stage, the architecture should have two input sizes of 480×1 (see Figure 5.3(b)). The CNN block is the same as the one introduced in the subsection 5.2.1). Unlike CNN, the dropout layer is followed by a flattening layer, not a fully connected layer. This is because the flattening layer is used to convert a multi-dimensional tensor into a one-dimensional vector without doing any additional processing [142]. This operation is a reshaping function with no learnable parameters. In the CNN+RNN architecture, we usually want to convert the CNN-extracted features straight into a sequence format suitable for RNN processing without further mixing or transformation. The flattening layer translates the multi-dimensional feature map into a one-dimensional vector free of parameters, preserving the original feature information obtained by CNN. A fully connected layer, on the other hand, will add more weights and biases and combine and change features in linear and nonlinear ways. This makes it more difficult for RNN to get time series information because it adds more model factors and computational work. It may also change the original feature structure.



(a) Signal-level fusion framework

(b) Feature-level fusion framework

Fig. 5.3 An overview of CNN+RNN framework

The RNN block contains an RNN layer, a fully connected layer and a softmax layer. The last two layers are for classification. Since GRU has the same performance as LSTM and lower computational cost [106], two layers related to GRU are selected for the RNN layer: GRU layer and Bidirectional GRU

(BiGRU) layer. The BiGRU layer integrates the capabilities of GRU with bidirectional processing, enabling the model to learn past and future details about the input sequence [143]. The BiGRU layer has two GRU layers(see figure 5.4), each of which concurrently processes the input sequence in both forward and backwards directions. In the forward pass, the GRU layer captures information from previous time steps, while the backwards pass acquires information from subsequent time steps. This bidirectional processing allows the model to capture long-term dependencies in the input sequence precisely. Finally, the outputs are concatenated and sent into a fully connected layer and a softmax layer for classification.

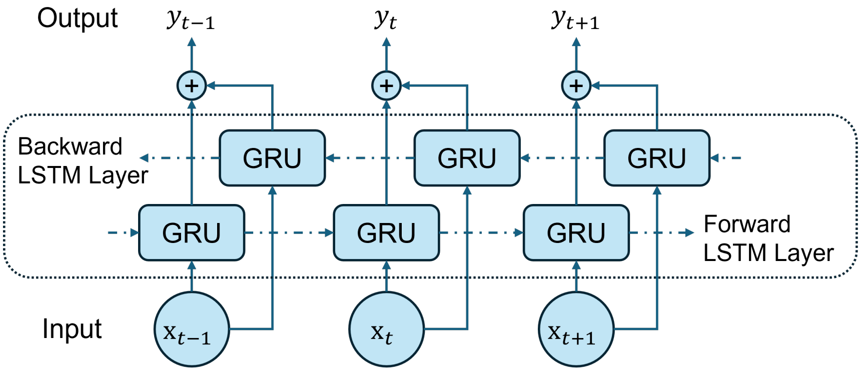


Fig. 5.4 The BiGRU architecture

5.3 Experimental Results and Discussion

5.3.1 Initial Conditions of the Algorithms

The experiments are performed with the MATLAB 2024 b version. The parameter settings for the CNN model of each task are shown in Table 5.1. The Adaptive Moment Estimation (Adam) optimiser was selected as it is widely used in binary classification tasks and has demonstrated robust convergence properties in related studies. The mini-batch size was set to 128, which represents a balance between computational efficiency and training stability. The maximum number of epochs was fixed at 500 to allow sufficient iterations for convergence while

CNN	Conv2D_1	number	32
		size	[16,1]
		stride	[1,1]
	Conv2D_2	number	64
		size	[16,1]
		stride	[1,1]
	Maxpooling_1	size	[2,1]
		stride	[1,1]
	Maxpooling_2	size	[2,1]
		stride	[1,1]
	dropout	probability	0.3
CNN+RNN	Conv2D_1	number	32
		size	[16,1]
		stride	[1,1]
	Maxpooling_1	size	[2,1]
		stride	[1,1]
	dropout	probability	0.2
CNN+SVM	Conv2D_1	number	32
		size	[16,1]
		stride	[1,1]
	Conv2D_2	number	64
		size	[16,1]
		stride	[1,1]
	Maxpooling_1	size	[2,1]
		stride	[1,1]
	dropout	probability	0.2

Table 5.1 Parameters setting for the CNN model of each task

preventing excessive training time. The initial learning rate was set to 0.001, a standard value recommended in the literature for Adam, and was found in preliminary experiments to provide stable convergence without overshooting. These parameter choices are thus informed by both established practice in deep learning and empirical tuning based on the dataset characteristics.

5.3.2 Data Preparation

Considering the impact of data imbalance on training models and single-signal and multi-signal on experimental results, the dataset from St. Vincent's University Hospital [3] are used in this experiment. St. Vincent dataset has 25 cases, and all the data are split into overlapping one-minute segments. SPO2 and pulse readings below 50 and above 300 are considered artefacts, and this segment will be removed. The segments containing apnoea events are classified as ‘apnoea’ while those without respiratory disturbances are labelled ‘normal’. When an apnoea event spanned two consecutive segments, fine-scale classification was applied. Respiratory disturbances lasting less than five seconds in any segment are classified as ’normal,’ as such brief interruptions do not significantly affect the overall respiratory pattern. If disturbances surpass this duration in any segment, it is classified as ’apnoea’, signifying a substantial interruption in respiratory function. This method guarantees accurate classification of each minute according to the intensity and length of interruptions. The detail of the data in different overlaps(ovlp) is shown in Table 5.2.

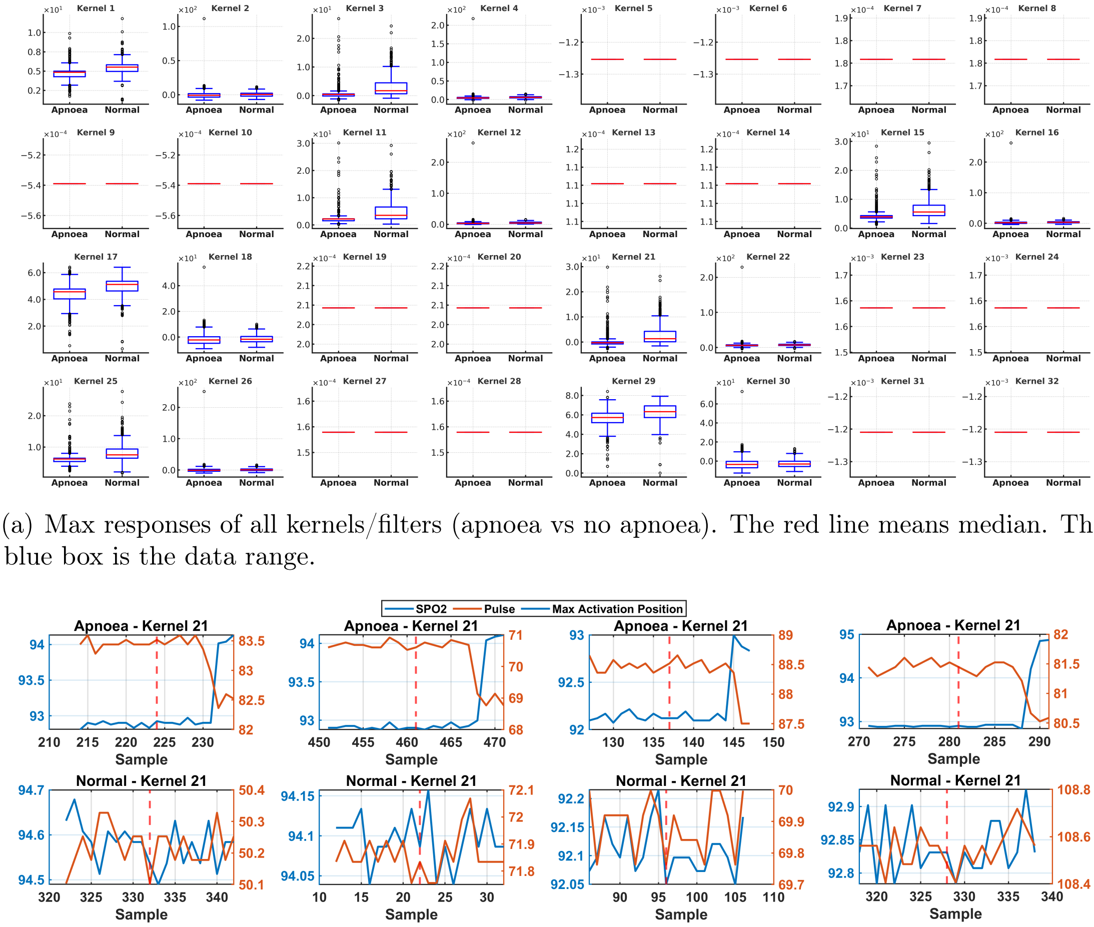
Table 5.2 Detail of the data in different overlaps(ovlp)

	apnoea	normal
0_ovlp	2569	7406
10_ovlp	2882	9094
20_ovlp	3289	11674
30_ovlp	3820	16137

The data is divided into a training set and a test set in a ratio of 8:2. Since the convolutional layer used is a 2D convolutional layer, the data must be reshaped into four-dimensional data to meet the model input requirements. The training data is divided into 2000 segments and one segment has 480 samples. The input data can be expressed as [480 2000]. In order to meet the input conditions of CNN, the input matrix is reshaped in the order of $[S\ C\ B\ T]$ to become 4-D data [480 1 1 2000]. These four numbers represent samples, channels, batches, and time respectively.

5.3.3 Feature Map from CNN

Figure 5.5(a) shows the maximum activation values of the 32 convolution kernels of the CNN model for the apnoea and normal classes. Each box plot shows



(a) Max responses of all kernels/filters (apnoea vs no apnoea). The red line means median. The blue box is the data range.

(b) Data around the strongest activation location in the input

Fig. 5.5 Feature map visualisation from CNN block

the statistical characteristics of the maximum response value of the convolution kernel to the input sequence of the two classes. The central line in the boxplot means the median, while the two edges of the box mean the interquartile range. Distinct disparities exist in the activation distribution for some convolution kernels between the two classes, which means that these kernels effectively capture the temporal structural attributes associated with apnoea and significantly influence the classification decision of the model.

Figure 5.5(b) shows how the kernel 21 captures the local structure of the original input signal at its maximum response position. The figure shows the signal pattern within a specific range before and after the response position. It can be observed from the figure that in the apnoea sample, the kernel tends to detect local structures with downward trends in SPO2 and upward trend patterns in pulse. These patterns may correspond to the sudden change in flow before apnoea.

5.3.4 Results for Different Model

This section mainly gives the results of different models based on CNN. The left side of the table represents different inputs, for example, 'SPO2_30_ovlp' means the input data is SPO2 signal in 30 seconds overlap. 'Feature' here means feature concatenation. 'SigConcate' means signal concatenation. The difference between these two is the location of the concatenation. Concatenate at the full-connection layer is feature concatenation, while signal concatenation is concatenated at the signal stage. 'F1-N' and 'F1-A' means the F1 score of the normal class and the F1 score of the apnoea class. Since this experiment is a medical classification, the sensitivity (sens) is mainly used to evaluate the model.

Results for CNN architecture

Table 5.3 Results for CNN model

	acc	sens	spec	F1-N	F1-A	κ
SPO2_0_ovlp	82.1%	53.7%	91.4%	0.88	0.60	0.48
SPO2_10_ovlp	82.3%	56.1%	90.2%	0.89	0.59	0.48
SPO2_20_ovlp	84.3%	57.9%	91.8%	0.90	0.61	0.51
SPO2_30_ovlp	85.6%	52.4%	93.3%	0.91	0.58	0.49
Pulse_0_ovlp	69.4%	30.7%	84.0%	0.80	0.36	0.16
Pulse_10_ovlp	71.8%	30.0%	85.0%	0.82	0.34	0.16
Pulse_20_ovlp	74.2%	28.3%	87.6%	0.84	0.33	0.18
Pulse_30_ovlp	78.2%	25.8%	91.8%	0.87	0.33	0.21
Featrue_0_ovlp	80.5%	60.2%	87.2%	0.87	0.61	0.48
Feature_10_ovlp	81.4%	56.2%	89.1%	0.88	0.59	0.47
Feature_20_ovlp	84.0%	55.7%	92.0%	0.90	0.61	0.51
Feature_30_ovlp	84.7%	53.5%	92.4%	0.91	0.58	0.49
SigConcate_0_ovlp	80.5%	56.7%	88.7%	0.87	0.60	0.47
SigConcate_10_ovlp	82.0%	58.3%	89.7%	0.88	0.61	0.50
SigConcate_20_ovlp	**84.0%**	**60.1%**	**90.4%**	**0.90**	**0.61**	**0.51**
SigConcate_30_ovlp	85.0%	53.4%	92.3%	0.91	0.57	0.48

Table 5.3 shows the results of the CNN model. In this table, the signal concatenation data with a 20-second overlap performs well. The feature concatenation with no overlap has 60.2% sensitivity, but theκvalue is not as good as signal concatenation. Compared to the same 20-second overlap input, the sensitivity of the connection is higher than that of the single signal result. However, there is still room for improvement in the overall performance, which can be achieved by optimizing network parameters or enhancing data preprocessing.

Results for CNN+SVM

Table 5.4 Results for CNN-SVM model

	acc	sens	spec	F1-N	F1-A	κ
SPO2_0_ovlp	79.0%	45.1%	90.9%	0.87	0.53	0.40
SPO2_10_ovlp	83.4%	47.9%	94.3%	0.90	0.58	0.48
SPO2_20_ovlp	84.1%	45.6%	95.5%	0.90	0.57	0.48
SPO2_30_ovlp	85.0%	40.4%	96.2%	0.91	0.52	0.44
Pulse_0_ovlp	68.0%	28.0%	82.0%	0.79	0.31	0.11
Pulse_10_ovlp	74.8%	18.8%	92.3%	0.85	0.26	0.14
Pulse_20_ovlp	76.8%	16.2%	93.4%	0.86	0.23	0.12
Pulse_30_ovlp	79.0%	14.9%	95.0%	0.88	0.22	0.13
SigConcate_0_ovlp	81.4%	50.7%	92.2%	0.88	0.59	0.47
SigConcate_10_ovlp	83.8%	54.7%	92.9%	0.90	0.62	0.52
SigConcate_20_ovlp	83.5%	58.8%	90.2%	0.90	0.60	0.50
SigConcate_30_ovlp	84.7%	51.1%	93.1%	0.91	0.57	0.48

Table 5.4 shows the results of the CNN-SVM model. In this model, using a single signal as input is unsatisfactory, but the fusion of two signals performs relatively well. Based on previous studies, SPO2 has always performed well in apnoea classification. This is because apnoea directly affects oxygen intake, which means the oxygen content in the blood [6]. However, in this CNN-SVM model, the classification sensitivity of the SPO2 signal is less than 50%, which is not ideal. This may be related to the unbalanced signal and may also be due to the parameter setting.

Results for CNN+RNN

Based on different RNN models, this experiment designed two CNN-RNN models, the CNN-GRU and the CNN-BiGRU models. Table 5.5 shows the results of the CNN-GRU model. The results show that based on the CNN-GRU model, the comprehensive performance of the signal concatenation fusion input with a 20-second overlap is better than that of other inputs. However, due to the randomness of the experiment (such as the existence of the dropout layer), the experimental parameters can be further optimized to obtain better performance.

Table 5.5 Results for CNN-GRU model

	acc	sens	spec	F1-N	F1-A	κ
SPO2_0_ovlp	81.1%	55.3%	90.2%	0.88	0.60	0.48
SPO2_10_ovlp	81.3%	56.0%	89.2%	0.88	0.59	0.47
SPO2_20_ovlp	83.4%	58.3%	91.5%	0.90	0.61	0.51
SPO2_30_ovlp	85.2%	51.6%	93.0%	0.91	0.57	0.48
Pulse_0_ovlp	70.5%	24.8%	86.1%	0.81	0.30	0.12
Pulse_10_ovlp	72.7%	20.5%	88.6%	0.83	0.26	0.11
Pulse_20_ovlp	73.5%	19.7%	88.6%	0.84	0.25	0.10
Pulse_30_ovlp	76.1%	20.0%	90.0%	0.86	0.25	0.12
Featrue_0_ovlp	81.3%	54.5%	90.6%	0.88	0.60	0.48
Feature_10_ovlp	82.3%	55.9%	90.7%	0.89	0.60	0.49
Feature_20_ovlp	85.4%	57.7%	92.9%	0.91	0.63	0.54
Feature_30_ovlp	83.7%	53.8%	91.0%	0.90	0.56	0.46
SigConcate_0_ovlp	80.2%	50.3%	90.7%	0.87	0.57	0.44
SigConcate_10_ovlp	82.9%	57.0%	91.0%	0.89	0.61	0.51
SigConcate_20_ovlp	**83.6%**	**61.2%**	**89.7%**	**0.90**	**0.61**	**0.51**
SigConcate_30_ovlp	85.2%	50.5%	93.0%	0.91	0.56	0.47

Table 5.6 shows the results of the CNN-BiGRU model. The overall trend of this result is similar to that of CNN-GRU. However, a 10-second overlap is more suitable for this model. By comparing the detection results after fusion of the two stages, the results of fusion in the signal stage are slightly higher than others.

It can be seen from these four tables that the blood oxygen signal can directly detect apnoea, while the pulse signal is slightly insufficient. This may be due to the limitation of the pulse signal for apnoea classification. Although the pulse

Table 5.6 Results for CNN-BiGRU model

	acc	sens	spec	F1-N	F1-A	κ
SPO2_0_ovlp	80.6%	51.0%	90.1%	0.88	0.56	0.44
SPO2_10_ovlp	81.6%	55.0%	90.6%	0.88	0.60	0.48
SPO2_20_ovlp	83.6%	56.6%	91.1%	0.90	0.60	0.50
SPO2_30_ovlp	84.6%	52.7%	92.2%	0.91	0.57	0.48
Pulse_0_ovlp	70.3%	26.8%	84.8%	0.81	0.31	0.13
Pulse_10_ovlp	71.2%	23.0%	86.4%	0.82	0.28	0.11
Pulse_20_ovlp	73.2%	27.5%	86.7%	0.83	0.32	0.16
Pulse_30_ovlp	76.4%	24.1%	88.5%	0.86	0.28	0.14
Featrue_0_ovlp	80.9%	56.7%	89.3%	0.87	0.61	0.48
Feature_10_ovlp	83.2%	56.1%	91.6%	0.89	0.61	0.51
Feature_20_ovlp	82.4%	54.9%	89.8%	0.89	0.57	0.46
Feature_30_ovlp	83.7%	55.5%	90.7%	0.90	0.58	0.48
SigConcate_0_ovlp	79.8%	54.3%	88.8%	0.87	0.58	0.45
SigConcate_10_ovlp	**81.7%**	**58.4%**	**89.0%**	**0.88**	**0.60**	**0.49**
SigConcate_20_ovlp	83.6%	56.2%	91.8%	0.90	0.61	0.51
SigConcate_30_ovlp	84.1%	57.4%	90.6%	0.90	0.58	0.49

signal also changes when apnoea occurs, the pulse signal still has limitations in diagnosing apnoea. Sleep apnoea is mainly caused by airway obstruction or central nervous system abnormalities. However, the pulse signal does not directly indicate the occurrence of apnoea like the airflow sensor but is indirectly inferred through the secondary effects of the cardiovascular system [144]. This means that if some short or mild apnoea does not cause an obvious heart rate response, the pulse signal may not have an apparent change, which may cause missed events. But even so, the result after signal fusion is still better than that of a single signal, which shows that pulse signal is still helpful in indirectly detecting apnoea.

5.3.5 Comparison of Performance and Discussion

Table 5.7 gives the results of different models based on signal concatenation with 20-second overlap. CNN-GRU shows the best performance among these models. Given that the outcomes of the CNN model closely resemble those of the CNN-GRU model, we choose to perform a more detailed evaluation based on

Table 5.7 Results of different models based on signal concatenation with 20-second overlap

	acc	sens	spec	F1-N	F1-A	κ
CNN	84.0%	60.1%	90.4%	0.90	0.61	0.51
CNN+SVM	83.5%	58.8%	90.2%	0.90	0.60	0.50
CNN+GRU	**83.6%**	**61.2%**	**89.7%**	**0.90**	**0.61**	**0.51**
CNN+BiGRU	83.6%	56.2%	91.8%	0.90	0.61	0.51

different dropout values. Table 5.8 and table 5.9 show the performance of the CNN and the CNN-GRU models based on different dropout probabilities, using signal concatenation with a 20-second overlap as input. From these tables, both models perform well when the probability is 0.5. CNN-GRU is the best among them, which can be shown in figure 5.6. This figure shows theκvalues obtained on the validation set for different dropout values using CNN and CNN-GRU. The maximum performance on the validation set was $kappa$ = 0.5332 with *dropout probability* = 4 and the CNN-GRU model. The other configurations performed slightly lower, so this best model was ultimately chosen to continue evaluating the test data.

Table 5.8 Performance of the CNN model for signal concatenation inputs with a 20-second overlap based on different dropout probabilities

Dropout probability	acc	sens	spec	F1-N	F1-A	κ
0.1	82.5%	58.7%	88.9%	0.89	0.59	0.4750
0.2	83.4%	57.9%	90.3%	0.90	0.60	0.4919
0.3	84.0%	60.1%	90.4%	0.90	0.61	0.5131
0.4	84.0%	57.9%	91.1%	0.90	0.61	0.5058
0.5	**84.7%**	**57.3%**	**92.1%**	**0.90**	**0.61**	**0.5184**

Table 5.9 Performance of the CNN-GRU model for signal concatenation inputs with a 20-second overlap based on different dropout probabilities

Dropout probability	acc	sens	spec	F1-N	F1-A	κ
0.1	83.7%	54.1%	91.7%	0.90	0.59	0.4857
0.2	84.5%	56.5%	92.1%	0.90	0.61	0.5127
0.3	83.6%	61.2%	89.7%	0.90	0.61	0.5100
0.4	84.8%	58.8%	91.9%	0.90	0.62	0.5296
0.5	**84.4%**	**62.4%**	**90.4%**	**0.90**	**0.63**	**0.5332**

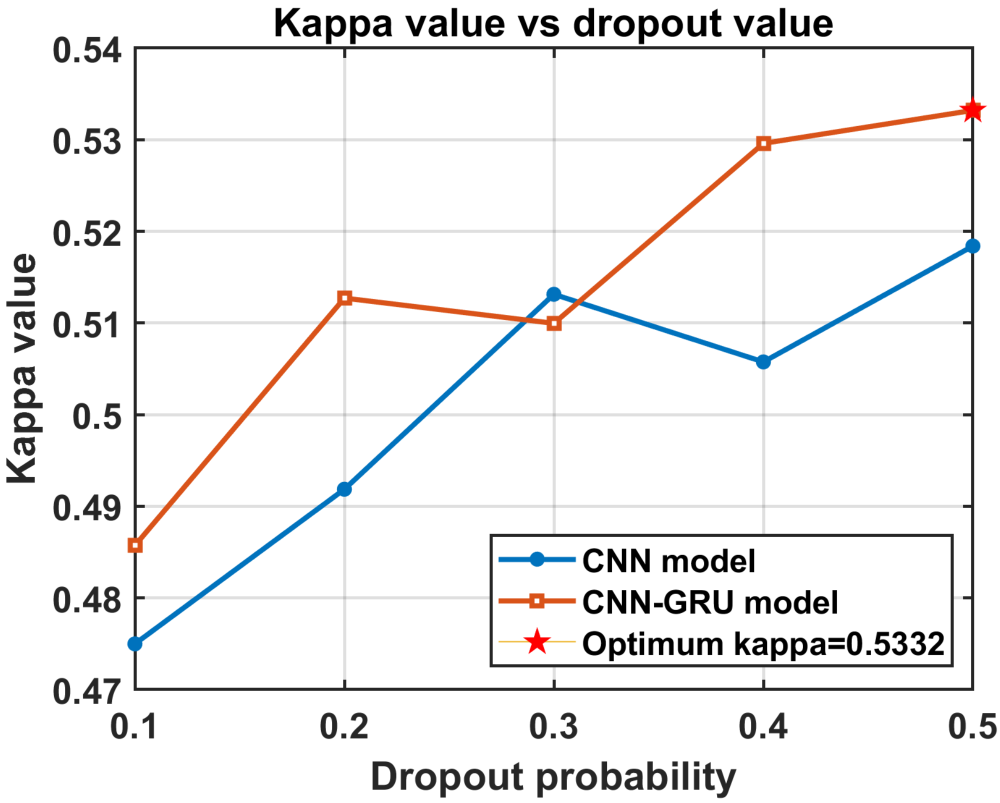


Fig. 5.6 Diagnostic performance of convolutional and recurrent neural network (CNN) and convolutional and recurrent neural network (CNN + RNN) architectures for different numbers of dropout probabilities.

Table 5.10 Performance of the CNN-GRU model for signal concatenation inputs with a 20-second overlap based on different numbers of neurons in the GRU layer (NG).

NG	acc	sens	spec	F1-N	F1-A	κ
1	85.5%	53.4%	94.3%	0.91	0.61	0.5258
2	85.2%	56.3%	93.1%	0.91	0.62	0.5290
4	**85.4%**	**61.5%**	**91.9%**	**0.91**	**0.64**	**0.5510**
8	84.3%	60.2%	90.8%	0.90	0.62	0.5219
16	84.2%	56.6%	91.8%	0.90	0.61	0.5080
32	85.0%	60.7%	91.7%	0.91	0.63	0.5407
64	83.6%	61.2%	89.7%	0.90	0.61	0.5100

Figure 5.7 shows theκvalues obtained in the validation set using different numbers of neurons in the GRU layer (NG) values for CNN and RNN. The maximum performance in the validation set is $kappa = 0.5510$ with $NG = 4$.

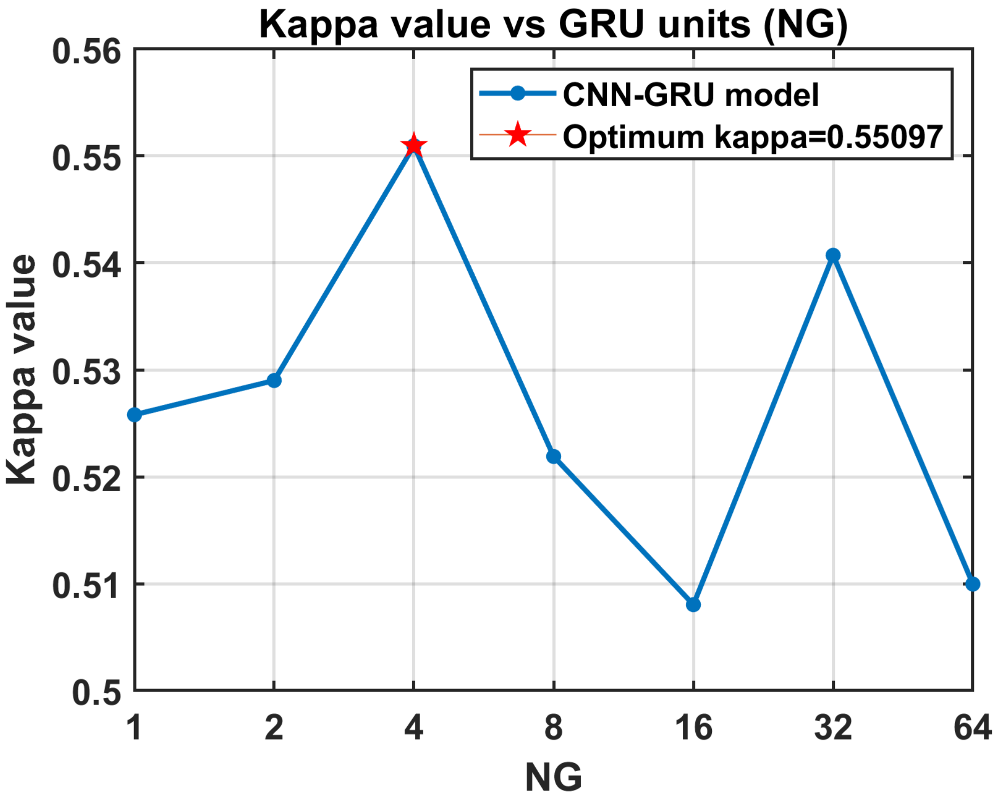


Fig. 5.7 Diagnostic performance of convolutional and recurrent neural network (CNN + RNN) architectures for different numbers of neurons in the GRU layer (NG).

It can be seen from table 5.10 that when NG=4, the model sensitivity reaches 61.5%, and the F1 score of the apnoea category is 0.64, which is the highest value in all experiments.

5.4 Summary

This chapter thoroughly investigates deep learning techniques, with a particular focus on CNN, aimed at detecting sleep apnoea using pulse and oximetry data. Initially, the benefits of deep learning are examined, highlighting its capacity to autonomously extract crucial features from raw signals. Furthermore, the discussion includes its limitations, such as the requirement for extensive datasets and considerable processing capabilities.

The subsequent section presents the core architecture of CNN along with two enhanced hybrid approaches. The hybrid model combines CNN and SVM, where the CNN component extracts descriptive features from the data, and the SVM acts as a high-level classifier to distinguish apnoea events from normal segments. The second hybrid model, CNN-RNN, integrates convolutional layers that are designed to identify specific signal features with recurrent models such as the GRU or BiGRU, which are adept at capturing long-range temporal relationships in time series data.

This chapter introduces data preparation, including splitting the signal into overlapping windows and initialisation for model validation. Experiments are validated not only with a single signal but also with a fusion of two signals. In addition, each model's architecture and hyperparameter configuration are listed, giving the choice of layer arrangement, kernel dimension, activation function, and dropout rate.

The experimental results show that while independent CNNs can show robust performance, integrating SVM or RNN can further improve detection accuracy. The integrated CNN-RNN method is good at detecting subtle patterns in the signal because RNN can simulate long-term dynamics. In addition to comparing the validation results of different inputs in the model, the experiment also experimentally selected the optimal dropout rate and the number of GRU neurons. Among them, when dropout = 0.5, NG=4, the classification effect is the best, reaching 85.4% accuracy and 61.5% sensitivity. Theκvalue reached 0.5510, and the F1 score of apnoea was 0.64. These results show that deep learning architectures, especially hybrid models, have great potential to improve the accuracy and reliability of automatic sleep apnoea recognition. However, this chapter indicates that these methods require careful parameter optimisation and sufficient training data to achieve maximum effectiveness.

When compared to existing approaches in the literature, these results are competitive and highlight the potential of hybrid architectures. For instance, recent CNN-LSTM methods applied to the Apnea-ECG dataset achieved accuracies above 90%, with sensitivities exceeding 90% as well [135]. However, those methods often rely on larger, more balanced datasets and are primarily validated on ECG-based signals, which may not directly translate to pulse and oximetry data. Other works employing ensemble methods such as RUSBoost report accuracies around 88–89% but typically at the expense of sensitivity [145]. In contrast, the hybrid CNN-RNN approach developed in this study demonstrates comparable overall accuracy while maintaining stronger adaptability to imbalanced physiological data. These findings therefore reinforce the view that deep learning architectures, particularly when combined with recurrent mechanisms, hold significant promise for improving the accuracy and clinical reliability of automatic sleep apnoea recognition.

Chapter 6

Conclusions and Future Work

6.1 Summary and Contributions

This dissertation represents the importance of accurate identification of sleep apnoea. The limitations of traditional clinical techniques underscore the necessity for affordable, automated, home-based alternatives. Although conventional techniques like the PSG have clinical accuracy, they still have challenges with high costs, time-consuming processes and the requirement for specialised laboratory conditions. In addition, manual interpretation of physiological data, such as those obtained from electrocardiography, pulse oximetry, and electroencephalography, is time-consuming and subject to inter-observer variability. The signals' natural non-stationarity exacerbates these issues. Furthermore, data imbalance always poses a significant challenge to standard detection algorithms, which often rely on fixed thresholds or limited classification boundaries. Based on these challenges, the research presented in Chapters 3 to 5 offers a range of innovative solutions to address these issues.

Chapter 3 reviews the limits of traditional change detection approaches, including the standard CUSUM algorithm. Classical CUSUM is constrained by its dependence on a set threshold, a limitation that is insufficient for addressing constantly fluctuating physiological inputs. The dissertation presents an Adaptive CUSUM technique to address this issue. This innovative approach uses the log-likelihood ratio as a selection criterion, providing a more detailed assessment of deviations from predicted signal behaviour. In addition, by integrating a rational sub-grouping process, the algorithm can dynamically modify the detection threshold in real-time. This demonstrates its ability to adapt to the changing statistical properties of the input signal. This adaptive approach improves the algorithm's sensitivity, enabling it to detect subtle anomalies that might go unnoticed while reducing the false positive rate. False positive rates are crucial in healthcare environments as overdiagnosis may result in superfluous interventions. The multi-signal fusion aspect of the ACUSUM algorithm integrates pulse and blood oximetry data, marking a substantial shift from earlier research that focused predominantly on single signal assessment. The extensive dataset, such as the NSRR, substantiates improved detection accuracy and reliability.

Chapter 4 focuses on the use of advanced machine-learning methods to detect sleep apnoea. This problem is particularly pronounced in sleep apnoea detection because the number of normal events far exceeds the number of apnoea occurrences. This chapter highlights two methods for overcoming the significant problem of data imbalance. The first framework uses the DPGMM. This clustering technique can autonomously identify the number of clusters in the data. Compared to traditional supervised learning methods that require manual tuning of hyperparameters, DPGMM can self-adjust based on the inherent structure of the dataset, which has greater flexibility and robustness. Another method introduced in Chapter 4 is the RUSBoost method, which complements this probabilistic strategy by combining random undersampling and boosting techniques to alleviate the problems caused by class imbalance effectively. This dual strategy ensures that the minority class (usually representing significant apnoea events) is appropriately recognised during classification, thereby improving sensitivity and specificity. Experimental results from three datasets: ECG-Apnoea, CHAT, and StVincent, demonstrate that the RUSBoost classification framework outperforms traditional methods, such as support vector machines and k-nearest neighbours, while also confirming the feasibility of home sleep apnoea monitoring using low-cost blood oxygen and pulse signals.

Chapter 5 explores the use of deep learning for sleep apnoea identification. A hybrid model that combines the benefits of CNN and RNN is proposed to classify sleep apnoea. CNN is to capture the spatial and temporal features of sleep signals, and RNN is to detect apnoea. Since CNNs are good at extracting spatial patterns from raw input data, they cannot detect temporal dependencies. This study formulates a customised CNN architecture designed to process time series data. Subsequently, this design is combined with a recurrent module that utilises a GRU, which excels in modelling the sequential relationships inherent in physiological signals during sleep. The resulting CNN+RNN hybrid model is designed to autonomously extract and integrate spatial features from individual signal frames and the temporal dependencies of these features. In the experiment, the dropout probability and the number of neurons in GRU are adjusted to find the best parameters to achieve the optimal classification. This chapter also introduces a hybrid model of CNN+SVM. The hybrid model can capture complex dynamic changes that are challenging for a single method, and it represents a significant innovative contribution to the field of automatic sleep apnoea recognition.

6.2 Future Work

To improve the detection and monitoring of sleep apnoea, several intriguing directions for future research might be further explored based on the contributions and findings of this thesis. One important way is the real-time implementation of the suggested algorithms, especially the CNN+BiGRU framework and the Adaptive CUSUM. These techniques' practical usefulness would be confirmed and continuous, inconspicuous monitoring of sleep problems in various settings would be made possible by moving them from an offline experimental setup to a real-time, embedded system appropriate for wearable or home-based monitoring.

Another critical area for future work is extensive validation using larger, more heterogeneous datasets. This study has shown strong performance across various established datasets. However, the additional research can incorporate data from various populations, including patients of different ages and those with sleep apnoea severities. This research aims to refine the algorithm and augment its generalisation capacity, enabling the model to accommodate individual variations and signal conditions.

In addition, the integration of multimodal physiological signals inspires further exploration. Although this paper combines pulse and blood oxygen saturation data to improve detection accuracy, multiple results indicate that pulse signals are unsuitable for detecting sleep apnoea. Future research can try to combine other data sources, such as electrocardiogram, electroencephalogram, and even acoustic signals. This multimodal fusion can capture a broader range of physiological manifestations associated with sleep apnoea, thereby improving the robustness and sensitivity of the detection system.

The introduction of the ACUSUM algorithm raised the issue of computational efficiency. The proposed method can maximise the computational efficiency of space. Although they require a lot of processing resources, the adaptive methods and complex deep learning architectures demonstrated in this study are successful. Future research should focus on algorithmic improvements, including model compression or creating more efficient training paradigms, to help implement these models in real-world settings with limited resources.

The above suggestions for future work focus on the surface of model architectures, such as changing data and optimising algorithm time. But in fact, it is more important to explore advanced deep learning architectures, such as those based on attention mechanisms or transformer models, which can provide additional improvements for capturing long-range temporal dependencies and complex signal dynamics. Compared with the traditional CNN and RNN combination, such architectures may have higher performance, especially when dealing with the high-dimensional and noisy characteristics of physiological signals. And when training the model, the parameters can be more optimized to achieve better prediction results.

In summary, future work inspired by this paper includes real-time deployment of the developed methods, extensive validation across different datasets, multimodal data fusion, computational optimisation, and research on advanced neural architectures. These research directions can advance the automatic sleep apnoea detection field and help develop accessible, efficient, and reliable home monitoring systems.

References

[1] Thomas Penzel, George B. Moody, Roger G. Mark, Ary L. Goldberger, and Jörg-Hermann Peter. The apnea-ecg database. In *Proceedings of Computers in Cardiology 2000. Vol.27 (Cat. 00CH37163)*, pages 255–258, 2000.

[2] Carole L. Marcus, Reneé H. Moore, Carol L. Rosen, Bruno Giordani, Susan L. Garetz, H. Gerry Taylor, Ron B. Mitchell, Raouf Amin, Eliot S. Katz, Raanan Arens, Shalini Paruthi, Hiren Muzumdar, David Gozal, Nina Hattiangadi Thomas, Janice Ware, Dean Beebe, Karen Snyder, Lisa Elden, Robert C. Sprecher, Paul Willging, Dwight Jones, John P. Bent, Timothy Hoban, Ronald D. Chervin, Susan S. Ellenberg, and Susan Redline. A randomized trial of adenotonsillectomy for childhood sleep apnea. *New England Journal of Medicine*, 368(25):2366–2376, 2013.

[3] Ary L. Goldberger, Luis A. N. 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. *Circulation*, 101(23):e215–e220, 2000.

[4] Jair Cervantes, Farid Garcia-Lamont, Lisbeth Rodríguez-Mazahua, and Asdrubal Lopez. A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, 408:189–215, 2020.

[5] Sheng-Fu Liang, Chin-En Kuo, Yu-Han Hu, and Yu-Shian Cheng. A rule-based automatic sleep staging method. *Journal of Neuroscience Methods*, 205(1):169–176, 2012.

[6] Andrew R Spector, Daniel Loriaux, and Alfredo E Farjat. The clinical significance of apneas versus hypopneas: is there really a difference? *Cureus*, 11(4), 2019.

[7] John Wright and Trevor Sheldon. Sleep apnoea and its impact on public health. *Thorax*, 53(5):410–413, 1998.

[8] James M. Parish. Sleep-related problems in common medical conditions. *Chest*, 135(2):563–572, 2009.

[9] Carole L Marcus, Lee J Brooks, Sally Davidson Ward, Kari A Draper, David Gozal, Ann C Halbower, Jacqueline Jones, Christopher Lehmann, Michael S Schechter, Stephen Sheldon, et al. Diagnosis and management of childhood obstructive sleep apnea syndrome. *Pediatrics*, 130(3):e714–e755, 2012.

[10] Anita Ramachandran and Anupama Karuppiah. A survey on recent advances in machine learning based sleep apnea detection systems. *Healthcare*, 9(7), 2021.

[11] M. Emin Tagluk, Mehmet Akin, and Necmettin Sezgin. Classıfication of sleep apnea by using wavelet transform and artificial neural networks. *Expert Systems with Applications*, 37(2):1600–1607, 2010.

[12] Jessica Vensel Rundo and Ralph Downey. Chapter 25 - Polysomnography. In Kerry H. Levin and Patrick Chauvel, editors, *Clinical Neurophysiology: Basis and Technical Aspects*, volume 160 of *Handbook of Clinical Neurology*, pages 381–392. Elsevier, 2019.

[13] Sirin Roomkham, Bernd Ploderer, Simon Smith, and Dimitri Perrin. *Technologies for Quantifying Sleep: Improved Quality of Life or Overwhelming Gadgets?*, pages 151–164. Springer International Publishing, 04 2022.

[14] Fábio Mendonça, Sheikh Shanawaz Mostafa, Antonio G. Ravelo-García, Fernando Morgado-Dias, and Thomas Penzel. A review of obstructive

sleep apnea detection approaches. *IEEE Journal of Biomedical and Health Informatics*, 23(2):825–837, 2019.

[15] Da Woon Jung, Su Hwan Hwang, Jae Geol Cho, Byung Hun Choi, Hyun Jae Baek, Yu Jin Lee, Do-Un Jeong, and Kwang Suk Park. Real-time automatic apneic event detection using nocturnal pulse oximetry. *IEEE Transactions on Biomedical Engineering*, 65(3):706–712, 2018.

[16] Alimohamad Asghari and Fatemeh Mohammadi. Is apnea-hypopnea index a proper measure for obstructive sleep apnea severity. *Med J Islam Repub Iran*, 27(3):161–2, 2013.

[17] Richard B Berry, Stuart F Quan, Alexandre R Abreu, Marietta L Bibos, et al. The AASM manual for the scoring of sleep and associated events:rules, terminology and technical specifications. *Darien, IL: American Academy of Sleep Medicine*, 2.6, 2020.

[18] Adult Obstructive Sleep Apnea Task Force of the American Academy of Sleep Medicine. Clinical guideline for the evaluation, management and long-term care of obstructive sleep apnea in adults. *Journal of clinical sleep medicine*, 5(3):263–276, 2009.

[19] Nur HA Rashid, Soroush Zaghi, Marcelo Scapuccin, Macario Camacho, Victor Certal, and Robson Capasso. The value of oxygen desaturation index for diagnosing obstructive sleep apnea: A systematic review. *The Laryngoscope*, 131(2):440–447, 2021.

[20] Atul Malhotra, Indu Ayappa, Najib Ayas, Nancy Collop, Douglas Kirsch, Nigel Mcardle, Reena Mehra, Allan I Pack, Naresh Punjabi, David P White, and for SRS Task Force Gottlieb, Daniel J. Metrics of sleep apnea severity: beyond the apnea-hypopnea index. *Sleep*, 44(7):zsab030, 03 2021.

[21] Guo-Qiang Zhang, Licong Cui, Remo Mueller, Shiqiang Tao, Matthew Kim, Michael Rueschman, Sara Mariani, Daniel Mobley, and Susan Redline.

The national sleep research resource: Towards a sleep data commons. In *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, BCB '18, page 572, New York, NY, USA, 2018. Association for Computing Machinery.

[22] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

[23] Peter Christen, David J. Hand, and Nishadi Kirielle. A review of the F-Measure: Its history, properties, criticism, and alternatives. *ACM Comput. Surv.*, 56(3), Oct 2023.

[24] TANG Wan, HU Jun, Hui Zhang, WU Pan, and HE Hua. Kappa coefficient: a popular measure of rater agreement. *Shanghai Archives of Psychiatry*, 27(1):62, 2015.

[25] Kareem Abu, Massoud L. Khraiche, and Jason Amatoury. Obstructive sleep apnea diagnosis and beyond using portable monitors. *Sleep Medicine*, 113:260–274, 2024.

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

[27] Atul Malhotra and Joseph Loscalzo. Sleep and cardiovascular disease: an overview. *Progress in cardiovascular diseases*, 51(4):279, 2009.

[28] U Rajendra Acharya, K Paul Joseph, Natarajan Kannathal, Choo Min Lim, and Jasjit S Suri. Heart rate variability: a review. *Medical and Biological Engineering and Computing*, 44:1031–1051, 2006.

[29] Verónica Barroso-García, Gonzalo C. Gutiérrez-Tobal, David Gozal, Fernando Vaquerizo-Villar, Daniel Álvarez, Félix del Campo, Leila Kheirandish-Gozal, and Roberto Hornero. Wavelet analysis of overnight airflow to detect obstructive sleep apnea in children. *Sensors*, 21(4), 2021.

[30] Ahnaf Rashik Hassan and Md. Aynal Haque. An expert system for automated identification of obstructive sleep apnea from single-lead ecg using random under sampling boosting. *Neurocomputing*, 235:122–130, 2017.

[31] Gonzalo C. Gutiérrez-Tobal, Alonso-Álvarez M. Luz, Daniel Álvarez, Félix del Campo, Joaquín Terán-Santos, and Roberto Hornero. Diagnosis of pediatric obstructive sleep apnea: Preliminary findings using automatic analysis of airflow and oximetry recordings obtained at patients' home. *Biomedical Signal Processing and Control*, 18:401–407, 2015.

[32] Manish Sharma, Divyash Kumbhani, Jainendra Tiwari, Kumar T. Sudheer, and Acharya U. Rajendra. Automated detection of obstructive sleep apnea in more than 8000 subjects using frequency optimized orthogonal wavelet filter bank with respiratory and oximetry signals. *Computers in Biology and Medicine*, 144:105364, 2022.

[33] Gokhan Memis and Mustafa Sert. Multimodal classification of obstructive sleep apnea using feature level fusion. In *Proceedings of the IEEE 11th International Conference on Semantic Computing (ICSC)*, pages 85–88, 2017.

[34] Xiaoyun Zhao, Xiaohong Wang, Tianshun Yang, Siyu Ji, Huiquan Wang, Jinhai Wang, Yao Wang, and Qi Wu. Classification of sleep apnea based on EEG sub-band signal characteristics. *Scientific Reports*, 11(1):5824, March 2021.

[35] V Vimala, K Ramar, and M Ettappan. An intelligent sleep apnea classification system based on EEG signals. *Journal of Medical Systems*, 43(2):1–9, 2019.

[36] Wu Huang, Bing Guo, Yan Shen, Xiangdong Tang, Tao Zhang, Dan Li, and Zhonghui Jiang. Sleep staging algorithm based on multichannel data adding and multifeature screening. *Computer Methods and Programs in Biomedicine*, 187:105253, 2020.

[37] Varun Chandola and Ranga Raju Vatsavai. A gaussian process based online change detection algorithm for monitoring periodic time series. In *Proceedings of the 2011 SIAM international conference on data mining*, pages 95–106. SIAM, 2011.

[38] Ahnaf Rashik Hassan. Computer-aided obstructive sleep apnea detection using normal inverse Gaussian parameters and adaptive boosting. *Biomedical Signal Processing and Control*, 29:22–30, 2016.

[39] Chutinan Singtothong and Thitirat Siriborvornratanakul. Deep-learning based sleep apnea detection using sleep sound, spo2, and pulse rate. *International Journal of Information Technology*, pages 1–6, 2024.

[40] Zhenglin Li, Mahnaz Arvaneh, Heather E. Elphick, Ruth N. Kingshott, and Lyudmila S. Mihaylova. A Dirichlet process mixture model for autonomous sleep apnea detection using oxygen saturation data. In *Proceedings of the IEEE 23rd International Conference on Information Fusion (FUSION)*, pages 1–8, 2020.

[41] Ahnaf Rashik Hassan and Md. Aynal Haque. Identification of sleep apnea from single-lead electrocardiogram. In *Proceedings of the IEEE International Conference on Computational Science and Engineering (CSE) and IEEE Intl Conference on Embedded and Ubiquitous Computing (EUC) and 15th Intl Symposium on Distributed Computing and Applications for Business Engineering (DCABES)*, pages 355–360, 2016.

[42] Daniel Michelsanti, Zheng-Hua Tan, Shi-Xiong Zhang, Yong Xu, Meng Yu, Dong Yu, and Jesper Jensen. An overview of deep-learning-based audio-visual speech enhancement and separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 29:1368–1396, 2021.

[43] Samaneh Aminikhanghahi and Diane J Cook. A survey of methods for time series change point detection. *Knowledge and information systems*, 51(2):339–367, 2017.

[44] Thomas Flynn and Shinjae Yoo. Change detection with the kernel cumulative sum algorithm. In *Proceedings of 2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 6092–6099, 2019.

[45] Yoshinobu Kawahara and Masashi Sugiyama. Change-point detection in time-series data by direct density-ratio estimation. In *Proceedings of the 2009 SIAM international conference on data mining*, pages 389–400. SIAM, 2009.

[46] Mirko Boettcher. Contrast and change mining. *WIREs Data Mining and Knowledge Discovery*, 1(3):215–230, 2011.

[47] Marcel Bosc, Fabrice Heitz, Jean-Paul Armspach, Izzie Namer, Daniel Gounot, and Lucien Rumbach. Automatic change detection in multimodal serial mri: application to multiple sclerosis lesion evolution. *NeuroImage*, 20(2):643–656, 2003.

[48] Alexander Selvikvåg Lundervold and Arvid Lundervold. An overview of deep learning in medical imaging focusing on mri. *Zeitschrift für Medizinische Physik*, 29(2):102–127, 2019. Special Issue: Deep Learning in Medical Physics.

[49] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Sánchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60–88, 2017.

[50] Rakesh Malladi, Giridhar P Kalamangalam, and Behnaam Aazhang. Online bayesian change point detection algorithms for segmentation of epileptic activity. In *Proceedings of 2013 Asilomar Conference on Signals, Systems and Computers*, pages 1833–1837, 2013.

[51] Martin Staudacher, Stefan Telser, Anton Amann, Hartmann H. Hinterhuber, and Monika Ritsch-Marte. A new method for change-point detection

developed for on-line analysis of the heart beat variability during sleep. *Physica A: Statistical Mechanics and its Applications*, 349(3):582–596, 2005.

[52] Ping Yang, Guy Dumont, and J. Mark Ansermino. Adaptive change detection in heart rate trend monitoring in anesthetized children. *IEEE Transactions on Biomedical Engineering*, 53(11):2211–2219, 2006.

[53] Vu Linh Le, Daewoo Kim, Eunsung Cho, Hyeryung Jang, Roben Delos Reyes, Hyunggug Kim, Dongheon Lee, In-Young Yoon, Joonki Hong, and Jeong-Whun Kim. Real-time detection of sleep apnea based on breathing sounds and prediction reinforcement using home noises: Algorithm development and validation. *J Med Internet Res*, 25:e44818, Feb 2023.

[54] Salvatore Serrano, Luca Patanè, Omar Serghini, and Marco Scarpa. Detection and classification of obstructive sleep apnea using audio spectrogram analysis. *Electronics*, 13(13), 2024.

[55] Richard J. Radke, Srinivas Andra, Omar Al-Kofahi, and Badrinath Roysam. Image change detection algorithms: a systematic survey. *IEEE Transactions on Image Processing*, 14(3):294–307, 2005.

[56] Lucrezia Giorgi, Domiziana Nardelli, Antonio Moffa, Francesco Iafrati, Simone Di Giovanni, Ewa Olszewska, Peter Baptista, Lorenzo Sabatino, and Manuele Casale. Advancements in obstructive sleep apnea diagnosis and screening through artificial intelligence: A systematic review. *Healthcare*, 13(2), 2025.

[57] Li-Wen Chiu, Yang-Ren Chou, Yi-Chiao Wu, Meng-Liang Chung, Bing-Fei Wu, and Kun-Ta Chou. Video-based contactless detection of sleep apnea with deep-learning model. *IEEE Transactions on Instrumentation and Measurement*, 73:1–13, 2024.

[58] Elangovan Ramanujam, Thinagaran Perumal, and SJISJ Padmavathi. Human activity recognition with smartphone and wearable sensors using

deep learning techniques: A review. *IEEE Sensors Journal*, 21(12):13029–13040, 2021.

[59] Christian Janiesch, Patrick Zschech, and Kai Heinrich. Machine learning and deep learning. *Electronic Markets*, 31(3):685–695, 2021.

[60] Mesnil Benoit and Pierre Petitgas. Detection of changes in time-series of indicators using cusum control charts. *Aquatic Living Resources (0990-7440) (EDP Sciences)*, 22(2):187–192, 04 2009.

[61] Cesare Alippi and Manuel Roveri. An adaptive cusum-based test for signal change detection. In *Proceedings of 2006 IEEE International Symposium on Circuits and Systems*, pages 4–pp. IEEE, 2006.

[62] Vanli O. Arda and Rupert Giroux. An adaptive cumulative sum method for monitoring integer-valued time-series data. *Quality Engineering*, 34(2):230–247, 2022.

[63] Ali N. Akansu and Richard A. Haddad. Chapter 6 - wavelet transform. In Ali N. Akansu and Richard A. Haddad, editors, *Multiresolution Signal Decomposition (Second Edition)*, pages 391–442. Academic Press, San Diego, second edition edition, 2001.

[64] Kandala.N.V.P.S. Rajesh, Ravindra Dhuli, and T. Sunil Kumar. Obstructive sleep apnea detection using discrete wavelet transform-based statistical features. *Computers in Biology and Medicine*, 130:104199, 2021.

[65] Olivier Rioul and Martin Vetterli. Wavelets and signal processing. *IEEE Journal of Signal Processing Magazine*, 8(4):14–38, 1991.

[66] Fernando Vaquerizo-Villar, Daniel Álvarez, Leila Kheirandish-Gozal, Gonzalo C. Gutiérrez-Tobal, Verónica Barroso-García, Andrea Crespo, Félix del Campo, David Gozal, and Roberto Hornero. Wavelet analysis of oximetry recordings to assist in the automated detection of moderate-to-severe

pediatric sleep apnea-hypopnea syndrome. *PLOS ONE*, 13(12):1–18, 12 2018.

[67] Daniel Álvarez, Roberto Hornero, J. Victor Marcos, and Felix del Campo. Multivariate analysis of blood oxygen saturation recordings in obstructive sleep apnea diagnosis. *IEEE Transactions on Biomedical Engineering*, 57(12):2816–2824, 2010.

[68] Richard A. Groeneveld and Glen Meeden. Measuring Skewness and Kurtosis. *Journal of the Royal Statistical Society Series D: The Statistician*, 33(4):391–399, 12 2018.

[69] Jianbo Gao, Jing Hu, and Wen-wen Tung. Entropy measures for biological signal analyses. *Nonlinear Dynamics*, 68:431–444, 2012.

[70] Mohammad Reza Nazari Kousarrizi, AbdolReza Asadi Ghanbari, Mohammad Teshnehlab, Mahdi Aliyari Shorehdeli, and Ali Gharaviri. Feature extraction and classification of eeg signals using wavelet transform, svm and artificial neural networks for brain computer interfaces. In *Proceedings of 2009 International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing*, pages 352–355, 2009.

[71] Hafeez Ullah Amin, Aamir Saeed Malik, Rana Fayyaz Ahmad, Nasreen Badruddin, Nidal Kamel, Muhammad Hussain, and Weng-Tink Chooi. Feature extraction and classification for eeg signals using wavelet transform and machine learning techniques. *Australasian Physical & Engineering Sciences in Medicine*, 38:139–149, 2015.

[72] Hadi Ratham Al Ghayab, Yan Li, S. Siuly, and Shahab Abdulla. A feature extraction technique based on tunable q-factor wavelet transform for brain signal classification. *Journal of Neuroscience Methods*, 312:43–52, 2019.

[73] Shaul Hameed Syed and V. Muralidharan. Feature extraction using discrete wavelet transform for fault classification of planetary gearbox – a comparative study. *Applied Acoustics*, 188:108572, 2022.

[74] Duraisamy Ramyachitra and Parasuraman Manikandan. Imbalanced dataset classification and solutions: a review. *International Journal of Computing and Business Research (IJCBR)*, 5(4):1–29, 2014.

[75] Vaishali Ganganwar. An overview of classification algorithms for imbalanced datasets. *International Journal of Emerging Technology and Advanced Engineering*, 2(4):42–47, 2012.

[76] Zhi-Hua Zhou. *Machine learning*. Springer nature, 2021.

[77] Iqbal Muhammad and Zhu Yan. Supervised machine learning approaches: A survey. *ICTACT Journal on Soft Computing*, 5(3), 2015.

[78] Peter Dayan, Maneesh Sahani, and Grégoire Deback. Unsupervised learning. *The MIT Encyclopedia of the Cognitive Sciences*, pages 857–859, 1999.

[79] Jesper E Van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020.

[80] Leslie Pack Kaelbling, Michael L Littman, and Andrew W Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285, 1996.

[81] Vladimir Nasteski. An overview of the supervised machine learning methods. *Horizons. b*, 4(51-62):56, 2017.

[82] Sotiris B Kotsiantis. Decision trees: a recent overview. *Artificial Intelligence Review*, 39:261–283, 2013.

[83] Sourish Ghosh, Anasuya Dasgupta, and Aleena Swetapadma. A study on support vector machine based linear and non-linear pattern classification. In *Proceedings of 2019 International Conference on Intelligent Sustainable Systems (ICISS)*, pages 24–28, 2019.

[84] Vikramaditya Jakkula. Tutorial on support vector machine (SVM). *School of EECS, Washington State University*, 37(2.5):3, 2006.

[85] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. Knn model-based approach in classification. In *Proceedings of On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003.*, pages 986–996. Springer, 2003.

[86] Jinming Zou, Yi Han, and Sung-Sau So. Overview of artificial neural networks. In David J. Livingstone, editor, *Artificial Neural Networks: Methods and Applications*, pages 14–22. Humana Press, Totowa, NJ, 2009.

[87] Keiron O'Shea and Ryan Nash. An introduction to convolutional neural networks. *CoRR*, abs/1511.08458, 2015.

[88] Shadman Sakib, Nazib Ahmed, Ahmed Jawad Kabir, and Hridon Ahmed. An overview of convolutional neural network: Its architecture and applications. *Preprints*, February 2019.

[89] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ArXiv*, abs/1502.03167, 2015.

[90] Rezoana Bente Arif, Md. Abu Bakr Siddique, Mohammad Mahmudur Rahman Khan, and Mahjabin Rahman Oishe. Study and observation of the variations of accuracies for handwritten digits recognition with various hidden layers and epochs using convolutional neural network. In *Proceedings of 2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT)*, pages 112–117, 2018.

[91] Verónica Barroso-García, Gonzalo C. Gutiérrez-Tobal, Leila Kheirandish-Gozal, Fernando Vaquerizo-Villar, Daniel Álvarez, Félix del Campo, David Gozal, and Roberto Hornero. Bispectral analysis of overnight airflow to improve the pediatric sleep apnea diagnosis. *Computers in Biology and Medicine*, 129:104167, 2021.

[92] Robin M. Schmidt. Recurrent neural networks (rnns): A gentle introduction and overview, 2019.

[93] Sakib Ashraf Zargar. Introduction to sequence learning models: Rnn, lstm, gru. *Department of Mechanical and Aerospace Engineering, North Carolina State University*, 2021.

[94] Zae Myung Kim, Hyungrai Oh, Han-Gyu Kim, Chae-Gyun Lim, Kyo-Joong Oh, and Ho-Jin Choi. Modeling long-term human activeness using recurrent neural networks for biometric data. *BMC Medical Informatics and Decision Making*, 17:1–15, 2017.

[95] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[96] Sepp Hochreiter and Jürgen Schmidhuber. Lstm can solve hard long time lag problems. *Advances in Neural Information Processing systems*, 9, 1996.

[97] Greg Van Houdt, Carlos Mosquera, and Gonzalo Nápoles. A review on the long short-term memory model. *Artificial Intelligence Review*, 53(8):5929–5955, 2020.

[98] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *ArXiv Preprint ArXiv:1409.1259*, 2014.

[99] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *ArXiv Preprint ArXiv:1412.3555*, 2014.

[100] George Bazoukis, Sandeep Chandra Bollepalli, Cheuk To Chung, Xinmu Li, Gary Tse, Bethany L. Bartley, Salma Batool-Anwar, Stuart F. Quan, and Antonis A. Armoundas. Application of artificial intelligence in the diagnosis of sleep apnea. *Journal of Clinical Sleep Medicine*, 19(7):1337–1363, 2023.

[101] Sheikh Shanawaz Mostafa, Fábio Mendonça, Antonio G. Ravelo-García, and Fernando Morgado-Dias. A systematic review of detecting sleep apnea using deep learning. *Sensors*, 19(22), 2019.

[102] Lachlan D. Barnes, Kevin Lee, Andreas W. Kempa-Liehr, and Luke E. Hallum. Detection of sleep apnea from single-channel electroencephalogram (eeg) using an explainable convolutional neural network (cnn). *PLOS ONE*, 17(9):1–18, 09 2022.

[103] Lorin Werthen-Brabants, Yolanda Castillo-Escario, Willemijn Groenendaal, Raimon Jané, Tom Dhaene, and Dirk Deschrijver. Deep learning-based event counting for apnea-hypopnea index estimation using recursive spiking neural networks. *IEEE Transactions on Biomedical Engineering*, 2024.

[104] Ziyu Jia, Xiyang Cai, Gaoxing Zheng, Jing Wang, and Youfang Lin. Sleepprintnet: A multivariate multimodal neural network based on physiological time-series for automatic sleep staging. *IEEE Transactions on Artificial Intelligence*, 1(3):248–257, 2020.

[105] Jorge Jiménez-García, María García, Gonzalo C. Gutiérrez-Tobal, Leila Kheirandish-Gozal, Fernando Vaquerizo-Villar, Daniel Álvarez, Félix del Campo, David Gozal, and Roberto Hornero. A 2d convolutional neural network to detect sleep apnea in children using airflow and oximetry. *Computers in Biology and Medicine*, 147:105784, 2022.

[106] Jorge Jiménez-García, María García, Gonzalo C. Gutiérrez-Tobal, Leila Kheirandish-Gozal, Fernando Vaquerizo-Villar, Daniel Álvarez, Félix del Campo, David Gozal, and Roberto Hornero. An explainable deep-learning architecture for pediatric sleep apnea identification from overnight airflow and oximetry signals. *Biomedical Signal Processing and Control*, 87:105490, 2024.

[107] Yuan Gao and Dorota Glowacka. Deep gate recurrent neural network. In Robert J. Durrant and Kee-Eung Kim, editors, *Proceedings of The*

8th Asian Conference on Machine Learning, volume 63 of *Proceedings of Machine Learning Research*, pages 350–365, The University of Waikato, Hamilton, New Zealand, 16–18 Nov 2016. PMLR.

[108] Devdutt Baresary, Kushwant Kaur, Tanya Gera, Sanjoy Kumar Debnath, and Shiva Mehta. A comparative study of cnn and svm for sleep apnea classification: Evaluating performance and efficiency. In *Proceedings of Global Conference on Communications and Information Technologies (GCCIT)*, pages 1–5, 2024.

[109] Cees G. M. Snoek, Marcel Worring, and Arnold W. M. Smeulders. Early versus late fusion in semantic video analysis. In *Proceedings of the 13th Annual ACM International Conference on Multimedia*, MULTIMEDIA '05, page 399–402, New York, NY, USA, 2005. Association for Computing Machinery.

[110] Yimian Dai, Fabian Gieseke, Stefan Oehmcke, Yiquan Wu, and Kobus Barnard. Attentional feature fusion. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 3560–3569, January 2021.

[111] Yang Li, Lianghui Guo, Yu Liu, Jingyu Liu, and Fangang Meng. A temporal-spectral-based squeeze-and- excitation feature fusion network for motor imagery eeg decoding. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:1534–1545, 2021.

[112] Xianhui Chen, Wenjun Ma, Weidong Gao, and Xiaomao Fan. Bafnet: Bottleneck attention based fusion network for sleep apnea detection. *IEEE Journal of Biomedical and Health Informatics*, 28(5):2473–2484, 2024.

[113] Triantafyllos Afouras, Joon Son Chung, and Andrew Zisserman. My lips are concealed: Audio-visual speech enhancement through obstructions. *ArXiv Preprint ArXiv:1907.04975*, 2019.

[114] Xiangbo Shu, Jiawen Yang, Rui Yan, and Yan Song. Expansion-squeeze-excitation fusion network for elderly activity recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(8):5281–5292, 2022.

[115] Seksan Laitrakun. Merging-squeeze-excitation feature fusion for human activity recognition using wearable sensors. *Applied Sciences*, 13(4), 2023.

[116] Ewan S. Page. Cumulative sum charts. *Technometrics*, 3(1):1–9, 1961.

[117] Olivia A Grigg, VT Farewell, and DJ Spiegelhalter. Use of risk-adjusted cusum and rsprtcharts for monitoring in medical contexts. *Statistical Methods in Medical Research*, 12(2):147–170, 2003.

[118] Per Winkel and Nannan Zhang. *Statistical development of quality in medicine.* John Wiley & Sons, 2007.

[119] Ewan S Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954.

[120] Vijay Koshti. Cumulative sum control chart. *International Journal of Physics and Mathematical Sciences*, 1(1):28–32, 2011.

[121] James M Lucas. Cumulative sum (cusum) control schemes. *Communications in Statistics-Theory and Methods*, 14(11):2689–2704, 1985.

[122] Douglas C Montgomery. *Introduction to Statistical Quality Control.* John Wiley & Sons, Danvers, MA, USA, 2007.

[123] Alexander G. Tartakovsky. *Rapid Detection of Attacks in Computer Networks by Quickest Changepoint Detection Methods*, chapter Chapter 2, pages 33–70. World Scientific, 04 2014.

[124] Habeeb Olufowobi, Uchenna Ezeobi, Eric Muhati, Gaylon Robinson, Clinton Young, Joseph Zambreno, and Gedare Bloom. Anomaly detection approach using adaptive cumulative sum algorithm for controller area

network. In *Proceedings of the ACM Workshop on Automotive Cybersecurity*, AutoSec '19, page 25–30, New York, NY, USA, 2019. Association for Computing Machinery.

[125] Mustafa Sefik. Importance of the rational subgroups in designing control charts. *Computers & Industrial Engineering*, 35(1):205–208, 1998.

[126] Vijay Koshti. Cumulative sum control chart. *International Journal of Physics and Mathematical Sciences*, ISSN:2277–2111, 12 2011.

[127] Douglas A Reynolds et al. Gaussian mixture models. *Encyclopedia of Biometrics*, 741(659-663), 2009.

[128] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. Hierarchical dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

[129] David M. Blei and Michael I. Jordan. Variational inference for Dirichlet process mixtures. *Bayesian Analysis*, 1(1):121 – 143, 2006.

[130] Kenichi Kurihara, Max Welling, and Nikos Vlassis. Accelerated variational dirichlet process mixtures. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Proceedings of Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006.

[131] Jayaram Sethuraman. A constructive definition of dirichlet priors. *Statistica Sinica*, 4(2):639–650, 1994.

[132] Chris Seiffert, Taghi M. Khoshgoftaar, Jason Van Hulse, and Amri Napolitano. Rusboost: A hybrid approach to alleviating class imbalance. *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, 40(1):185–197, 2010.

[133] David W. Scott. Histogram. *WIREs Computational Statistics*, 2(1):44–48, 2010.

[134] LeRoy Bessler. *Distributions, Histograms, Box Plots, and Alternative Tools*, pages 465–502. Apress, Berkeley, CA, 2023.

[135] Lijuan Duan, Zikang Song, Yourong Xu, Yanzhao Wang, and Zhiling Zhao. An automatic sleep apnoea detection method based on multi-context-scale cnn-lstm and contrastive learning with ecg. *IET Cyber-Systems and Robotics*, 7(1):e70017, 2025.

[136] Kristina Zovko, Yann Sadowski, Toni Perković, Petar Šolić, Ivana Pavlinac Dodig, Renata Pecotić, and Zoran Đogaš. Advanced data framework for sleep medicine applications: Machine learning-based detection of sleep apnea events. *Applied Sciences*, 15(1), 2025.

[137] Momin M. Malik. A hierarchy of limitations in machine learning, 2020.

[138] Joshua Ayeni. Convolutional neural network (cnn): the architecture and applications. *Appl. J. Phys. Sci*, 4:42–50, 2022.

[139] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[140] Savita Ahlawat and Amit Choudhary. Hybrid cnn-svm classifier for handwritten digit recognition. *Procedia Computer Science*, 167:2554–2560, 2020. International Conference on Computational Intelligence and Data Science.

[141] Thomas G. Dietterich. Machine learning for sequential data: A review. In Terry Caelli, Adnan Amin, Robert P. W. Duin, Dick de Ridder, and Mohamed Kamel, editors, *Structural, Syntactic, and Statistical Pattern Recognition*, pages 15–30, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[142] Julian Kreiser, Monique Meuschke, Gabriel Mistelbauer, Bernhard Preim, and Timo Ropinski. A survey of flattening-based medical visualization techniques. *Computer Graphics Forum*, 37(3):597–624, 2018.

[143] Zhanhui Hu, Guangzhong Liu, Yanping Li, and Siqing Zhuang. Sagb: self-attention with gate and bigru network for intrusion detection. *Complex & Intelligent Systems*, 10(6):8467–8479, 2024.

[144] Thomas Penzel. Is heart rate variability the simple solution to diagnose sleep apnoea? *European Respiratory Journal*, 22(6):870–971, 2003.

[145] Dongjin Yang, Jingqiong Zhang, Zhenglin Li, Heather Elphick, Eishaan Bhargava, and Lyudmila Mihaylova. A machine learning framework for sleep apnoea detection based on imbalanced pulse and oximetry data. *Journal of Machine Learning in Fundamental Sciences*, 2025(1), Jul. 2025.