TH_ECE 18_42

Convolution Neural Network (CNN) Based Sleep Stage Classification Using Single Channel EEG

Name: Nasim Mahbub
Roll: 1809042

February, 2024

Department of Electronics and Communication Engineering
Khulna University of Engineering & Technology
Khulna- 9203, Bangladesh

Convolution Neural Network (CNN) Based Sleep Stage Classification Using Single Channel EEG

This thesis is submitted to the Department of Electronics and Communication Engineering, Khulna University of Engineering & Technology, in partial fulfillments for the requirements of the degree of

"Bachelor of Science in Electronics and Communication Engineering"

Supervised by

Dr. A. B. M. Aowlad Hossain
Professor,
Department of Electronics and Communication Engineering,
KUET, Khulna.

Submitted by …

Name: Nasim Mahbub
Roll: 1809042
Department of Electronics and Communication Engineering,
KUET, Khulna.

February, 2024

Department of Electronics and Communication Engineering
Khulna University of Engineering & Technology
Khulna- 9203, Bangladesh

Convolution Neural Network (CNN) Based Sleep Stage Classification Using Single Channel EEG

Author

Nasim Mahbub
1809042
Dept. of Electronics and Communication Engineering
Khulna University of Engineering & Technology

Supervisor

Dr. A. B. M. Aowlad Hossain
Professor
Dept. of Electronics and Communication Engineering
Khulna University of Engineering & Technology

……………………………………….
Signature

External

Dr. Sheikh Md. Rabiul Islam
Professor
Dept. of Electronics and Communication Engineering
Khulna University of Engineering & Technology

………………………………………
Signature

February, 2024

Department of Electronics and Communication Engineering
Khulna University of Engineering & Technology
Khulna- 9203, Bangladesh

Statement of Originality

I, [Nasim Mahub], hereby declare that this thesis book, titled [Convolution Neural Network (CNN) Based Sleep Stage Classification Using Single Channel EEG], represents my own work. All sources used in this thesis book have been duly acknowledged. Any assistance I received in its preparation is fully acknowledged and disclosed in the thesis. This thesis has not been partially or fully published anywhere. Every item on the thesis, including any subheadings, has a reference where necessary. Without the author's consent, copies of the thesis are prohibited. Additionally, this thesis has never before been sent in as an undergraduate thesis.

The statements said above are accurate. In light of this, this work has been sent in for assessment as part of an undergraduate thesis.

Date: February, 2024 **Nasim Mahbub**

Acknowledgement

I am grateful to ALLAH for the grace that enables me to face every challenge head-on and finish my thesis with initiative and determination. I would want to thank my parents for their unwavering emotional support throughout my life. I express my gratitude to my parents for their unwavering support of my mental well-being.

I am grateful to my supervisor **Prof. Dr. A. B. M. Aowlad Hossain**, Dept. of ECE, KUET for guiding me throughout the entire time. Without his earnest advice, counsel, and support, I could not have continued with my study project. His expressive and logical approach to issue analysis has always given me a clear path to overcome any obstacles I've encountered while conducting research for my thesis.

I am also grateful to my thesis external **Prof. Dr. Sheikh Md. Rabiul Islam**, Dept. of ECE, KUET for his time to evaluate the thesis. In addition, I want to express my gratitude to all the teachers for supporting me and encouraging the completion of my thesis work.

DEDICATION

Dedicated to my beloved Ammu and Abbu, whose unwavering love, guidance, and support have been the cornerstone of my journey. Your sacrifices and encouragement have fueled my aspirations, and I am forever grateful for your boundless love.

To my Nani, though you are no longer with us, your memory remains a beacon of inspiration. Your kindness and advice are still resonating with me, directing my steps as I follow my studies.This work stands as a tribute to the legacy you left behind, forever etched in my heart.

ABSTRACT

The categorization of sleep stages is crucial for precise diagnosis and management of sleep-related diseases. Electromyograms (EMG), electroencephalograms (EEG), electrooculograms (EOG), and electrocardiograms (ECG) are typically conducted by a sleep specialist who uses eye assessment of each 30-second signal to establish the sleep state. The first deep learning-based method for classifying sleep stages that may train continuously without requiring spectrogram calculation or the extraction of manually-created features. It can also benefit from all multivariate and multimodal Polysomnography (PSG) signals, including EEG, EMG, and EOG, as well as the temporal context of each 30-second data window. Making use of the variety of sensors, the first layer learns linear spatial filters to improve the signal-to-noise ratio for each modality. After that, a softmax classifier receives the learnt representation from the final layer. In order to facilitate wearability and minimize disruptions during sleep, this research has developed an automated sleep staging system that employs a single channel electroencephalogram (EEG) output to automatically identify different phases of sleep. To achieve the automatic sleep staging system, this study proposes a convolutional neural network (CNN) model. In order to evaluate the performance of the proposed model, experiments are performed on Sleep-EDF Expanded database (SEDFEDB). There are four types of algorithm that have been used in this thesis work. These are SVM, BPNN, DT and Proposed CNN algorithm. In SVM model, phase W had the highest accuracy and phase N1 had the lowest accuracy, with an overall accuracy of 79.45%. In BPNN model, stage W had the greatest precision rate at 90%, followed by stage N2 at 84%. With an overall accuracy of 78.33%, the N1 stage have the lowest f1-score rate at 66%, while the N3 and REM phases have f1-score rates close to 75%. In DT model, the W period has the highest precision rate (88%), followed by the N3 period (86%), and the N1 period (61%), which has the lowest recognition rate, with an overall accuracy rate of 76.25%.In the SEDFEDB, the overall accuracy (ACC), Cohens Kappa co-efficient (Kappa) obtained by proposed CNN model are 80%, 0.81 respectively. Experimental results show that the performance of the proposed model is better in precision, re-call, F1-score, specificity of wake sleep stage which are 0.99, 0.95, 0.97 and 0.99.The N1, N2, REM stage has also improved in some performance. This work offers a fresh approach to clinical EEG monitoring mobility.

TABLE OF CONTENTS

List of Figures

List of Tables

Table

CHAPTER 1

INTRODUCTION

1.1 Overview

For humans, sleep is a crucial physiological phenomena that helps the body reorganize itself [1]. The majority of the body's physiological processes stop when a person goes into sleep. In order to eliminate human weariness, promote tissue repair and cell rearrangement, and get the body ready for physiological activities when awake, the pituitary gland secretes more growth hormones and prohormones during this period [2], [3].

It's crucial to keep in mind that sleep is a process that can be divided into different sleep intervals based on its depth [4], [5]. Recent studies have shown that specific brain wave ratios can be used to distinguish between the three primary stages of sleep staging: waking (W), no-rapid eye movement (NREM), and rapid eye movement (REM) [6], [7]. The American Academy of Sleep Medicine (AASM) divided NREM into three stages in 2007: NREM 1 (N1), NREM 2 (N2), and NREM 3 (N3). As a result, the AASM standard divides the sleep duration into five phases: W, N1, N2, N3, and REM. Understanding sleep mechanics and providing clinical diagnosis and therapy for sleep disorders are predicated on accurate sleep staging.

Sleep specialists classify sleep stages traditionally by visually examining signals like EEG, EOG, ECG, and EMG. The suggested deep learning method utilizes all PSG data and learns end-to-end without generating spectrograms or removing custom features. The model classifies sleep stages using a softmax classifier and linear spatial filters. In this thesis study, an effective convolutional neural network (CNN) is used to classify the sleep stage. Three well-known machine learning classifiers—Random Forest (RF), Support Vector Machines (SVM), and Back Propagation Neural Network (BPNN)—are used to examine the performance of the suggested CNN model. The Cohens Kappa co-efficient (Kappa) of the suggested model is 0.81 and it attained an accuracy of 80%.

The thesis work is completely summarized in this chapter. The problem identified for this particular research project, as well as the problem statement and purpose of this effort, are all defined in detail by examining the literature review. The segment includes a brief explanation of the approach taken to solve the challenge. The chapter ends with a summary of the work done and the thesis outline.

1.2 Background and motivation

Sleep is one of our basic daily needs and is critical to preserving brain health. Any sleep problem can affect everyday tasks, security, and overall well-being. Daily activities have an influence on sleep amount and quality. Undiagnosed and untreated sleep disturbances can also result in ischemia, arrhythmias, myocardial infarction, high blood pressure, and brain attacks [8]. PSG is a method that doctors use to assess sleep issues. A few of the signals that comprise PSG include the electroencephalogram (EEG), electrooculography (EOG), electromyogram (EMG), electrocardiogram (ECG), airflow, thoracic and abdominal movements, and oximetry based on modifications to EEG. Rechtschaffen and Kales standardized the classification rules for the different stages of sleep and divided non-rapid eye movement (NREM) sleep into four new stages. The combining of stage 3 and stage 4 into stage N3 was one of the most crucial criteria set forward by the American Academy of Sleep Medicine (AASM) in 2004 [9]. The classification of sleep stages manually takes time and is prone to inaccuracy. PSG recordings use numerous signal channels, and all-night PSG recordings are visually scored by a professional [10].As a result, numerous studies have been carried out to develop an automatic method of classifying sleep stages [11]. In recent years, an increasing number of researchers have attempted to address the problem of sleep staging by using artificial intelligence approaches like machine learning and deep learning. EEG, EOG, and EMG data are examples of physiological signals from which machine learning-based algorithms often extract the appropriate features [12-15]. Then, more representative signal features are chosen using the feature selection algorithm. The classifier then classifies various stages of sleep using the features that were chosen. These methods have some issues even though they have produced some successes. For example, choosing the most discriminative sleep features necessitates that researchers have professional experience in the field of sleep medicine, which can be quite difficult to acquire. Another issue is that temporal physiological signals are poorly modeled by conventional machine learning algorithms.

Deep learning has become popular in several disciplines, including bio-signal processing [16], emotion identification [17], [18], and medical imaging [19], [20]. It also offers a novel method for assessing sleep. Convolutional neural networks (CNNs) have been employed by certain researchers to create models of sleep stages [21]. The effectiveness of sleep staging can be improved using these techniques. Traditional CNNs, however, ignore context-related information and solely concentrate on information that is already in the receptive field. They cannot accurately extract essential local features and can readily interfere with non-key features. Recurrent neural networks (RNNs) have attracted several academics' attention in recent years [22].

Certain automated sleep staging algorithms are intricate and imprecise, while manual sleep staging by experts takes a lot of time and is prone to subjective influences. The goal of this thesis study is to address the drawbacks of both manual and current automatic sleep staging techniques by presenting a single-channel EEG-based approach. This work intends to construct an algorithm that can effectively categorize sleep stages using single-channel EEG data, thereby offering dependable technical support for diagnosing sleep disorders. Clinical EEG monitoring is more portable when single-channel EEG data is used, which increases its usefulness and accessibility for sleep assessment.

1.3 Literature Survey

The initial phase in the literature review section is to place the findings in the larger context of sleep medicine, highlighting the importance of classifying sleep stages for comprehending sleep physiology and identifying sleep disorders. It emphasizes how common sleep-related problems are and how they affect people's health and wellbeing. The goals and parameters of the literature review, giving readers a road map and directing them through the main topics and areas of interest that will be covered in the parts that follow. It lays up in detail the subjects that will be covered and the precise research questions that will be looked at.

Below table is about Literature Summary on sleeping stage classification related to this work.

Table 1.1 : Literature Summary on sleeping stage classification using single channel EEG related to this work.

SL	Reference (Journal/Conference papers)	Outcome	Limitations/Problems/ Challenging Issues	Proposed Solution
[23]	Junming Zhang, Ruxian Yao, Wengeng Ge, Jinfeng Gao(Orthogonal convolutional neural networks for automatic sleep stage classification based on single-channel EEG) https://doi.org/10.1016/j.cmpb.2019.105089 0169-2607/© 2019 Published by Elsevier B.V.	1. To increase performance, the squeeze-and-excitation network (SENet) and autoencoder are fused. 2. On two available datasets, the overall classification accuracy was 79.4% and 81.6%.	It is challenging to evaluate the suggested method's performance in contrast to other cutting-edge sleep stage categorization techniques because the paper does not compare it with them.	Future study might compare the suggested method with other cutting-edge sleep stage classification techniques, taking metrics like accuracy, sensitivity, and specificity into account, in order to overcome the lack of comparison with current methods.
[24]	Zhao, S.; Long, F.; Wei, X.; Ni, X.; Wang, H.; Wei, B. (Evaluation of a Single-Channel EEG-Based Sleep Staging Algorithm). Int. J. Environ. Res. Public Health 2022, 19, 2845. https://doi.org/10.3390/ ijerph19052845	1.With an average accuracy of 80.61%, the random forest classifier outperformed the other four algorithms in terms of sleep staging. 2. The Wake phase had the highest recognition rate (92.13%), whereas the N1 phase had the lowest (73.46%).	The possible effects of noise or artifacts in the single-channel EEG data on the sleep staging algorithm's accuracy are not covered in the research.	Examining how noise and distortions in the single-channel EEG data affect the algorithm's accuracy might reveal possible problems and strengthen the algorithm's resilience.

[25]	Fan Li, Rui Yan, Reza Mahini, Lai Wei, Zhiqiang Wang, Klaus Mathiak, Rong Liu, and Fengyu Cong(End-to-end sleep staging using convolutional neural network in raw single-channel EEG) https://doi.org/10.1016/j.bspc. 2020.102203 Received 11 April 2020; Received in revised form 14 August 2020; Accepted 29 August 2020	1. In two public datasets, SHHS1 (88.1%) and Sleep-EDFx (85.3%), the suggested model had good overall accuracies, demonstrating its effectiveness and generalizability in EEG sleep staging. 2. In addition to addressing the issue of class imbalance in sleep staging, the research suggests a method to raise stage N1 accuracy while taking overall accuracy into account.	The article does not fully address the issue of class imbalance in sleep staging because the recommended strategy to improve the accuracy of stage N1 may lower the accuracy overall.	Future study might investigate sophisticated strategies like data augmentation, oversampling, or ensemble methods to increase the accuracy of stage N1 while keeping a high overall accuracy in order to solve the issue of class imbalance in sleep staging.
[26]	Santosh Kumar Satapathya, Hari Kishan Kondaveetib, S R Sreejac,Hiral Madhania, Nitinsingh Rajput , Debabrata Swaina(A Deep Learning Approach to Automated Sleep Stages Classification Using Multi-Modal Signals) (https://creativecommons.org/licenses/by-nc-nd/4.0) 1877-0509 © 2023 The Authors. Published by Elsevier B.V.	1.Introducing a robust Deep Learning classifier for automated sleep stage classification using CNN and LSTM. 2. Using CNN + LSTM, the suggested model attained an accuracy of 87.4%.	1.The paper uses the Sleep-EDF dataset which contains data from only 30 subjects. A larger and more diverse dataset could help improve the model's generalizability. 2. Lack of long-term validation: The models were tested on a held-out test set from the same dataset, but their performance on new unseen long-term data is not evaluated.	Ensemble models: Combining multiple classifiers through techniques like stacking can help reduce errors from any single model and achieve more robust performance than individual models alone

[27]	Akara Supratak, Hao Dong, Chao Wu, and Yike Guo(DeepSleepNet: a Model for Automatic Sleep Stage Scoring based on Raw Single-Channel EEG) (http://ieeexplore.ieee.org/document/7961240/.) DOI: 10.1109/TNSRE.2017.2721116	1. In comparison to state-of-the-art techniques, the outcomes demonstrated that DeepSleepNet obtained comparable overall accuracy and macro F1-score on both datasets.	The paper does not provide a detailed analysis of the learned features and their interpretability	Additional analysis may be carried out to comprehend the link between the extracted characteristics and sleep stage categorization in order to improve the interpretability of the learnt features.
[28]	Wei Pei , Yan Li , Siuly Siuly(Sleep Stage Classification using deep learning using double channel EEG) Computers, Materials & Continua DOI:10.32604/cmc.2022.021830Received: 16 July 2021; Accepted: 03 September 2021	When compared to other datasets, this technique produced better results for accuracy, precision, recall, and F1-score values on the SHHS1-700 dataset.	The paper does not discuss the potential impact of imbalanced class distribution on the performance of the sleep stage classification	More pertinent variables that could influence the load change could be included to the suggested model to make it better.
[29]	Emadeldeen Eldele, Chee-Keong Kwoh, and Cuntai Guan(An Attention-Based Deep Learning Approach for Sleep Stage Classification With Single-Channel EEG) Digital Object Identifier 10.1109/TNSRE.2021.3076234 Accepted April21, 2021. Date of publication April 28, 2021	The AttnSleep model also showed better performance when compared to other approaches, such as DeepSleepNet, SleepEEGNet, ResnetLSTM, and MultitaskCNN. It achieved better classification performance, benefited from reduced training complexity compared to other models.	The paper does not discuss the potential impact of hyperparameter tuning on the performance of the AttnSleep model, which could affect its generalizability to different datasets and settings.	More research might examine the AttnSleep model's sensitivity to various hyperparameter settings and carry out a methodical study to find the best configurations for various datasets and classification tasks in order to address the possible influence of hyperparameter tweaking.

[30]	Sagar Santaji ,Veena Desai(Analysis of EEG Signal to Classify Sleep Stages Using Machine Learning) https://doi.org/10.1007/s41782-020-00101-9Received: 3 April 2020 / Revised: 5 July 2020 / Accepted: 25 July 2020 © Springer Nature Singapore Pte Ltd. 2020	When comparing the paper's accuracy and viability to other recent research on the classification of sleep stages, there is no doubt that the former is superior.	The size and diversity of the dataset used to train and evaluate the machine learning algorithms are not disclosed in the research.	larger and more varied datasets should be used in additional validation experiments to evaluate the sensitivity, specificity, and accuracy of the suggested methods.

1.4 Objective

1. To collect the data which involves gathering relevant data sources and preprocessing ensures data quality and consistency.

2. To extract the feature which involves identifying meaningful patterns and generating representations from the data.

3. To design the model architecture which includes selecting suitable algorithms and structuring the flow of information within the model.

4. To train the model , Optimization & Performance Evaluation

1.5 Research methodology

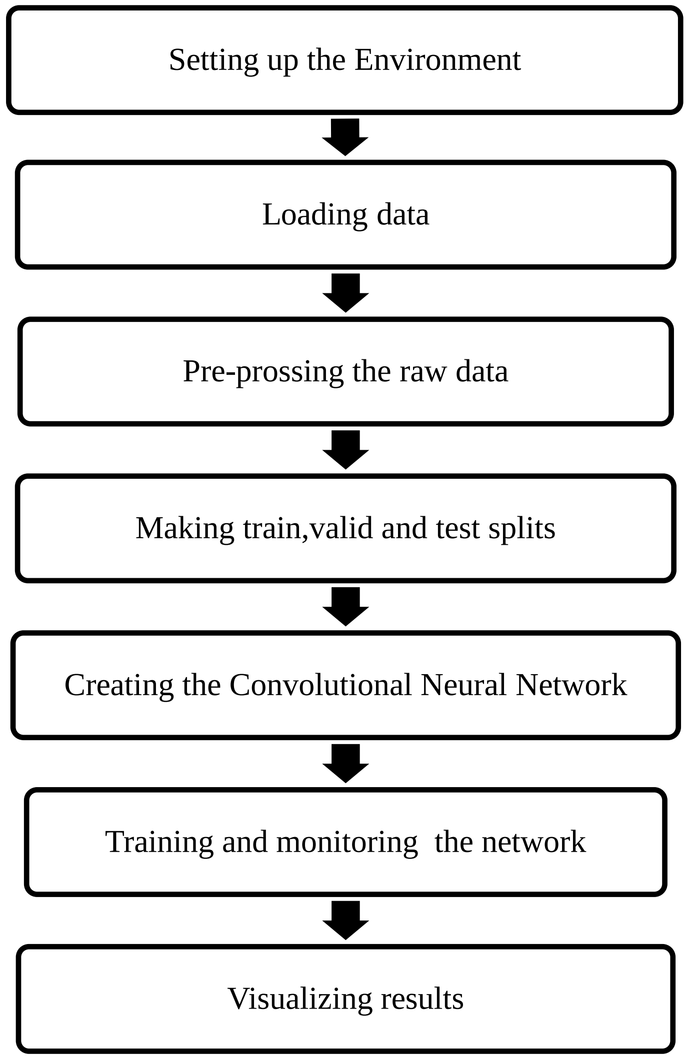


Fig 1.1: Demonstrate working procedure of research methodology

Loading data & Pre-processing raw data: Importing the raw EEG data and cropping the duration of wake events prior to and following sleep events from the Sleep Physionet dataset. The raw data is then preprocessed, and recordings are filtered using a lowpass filter with a cutoff frequency of 30 Hz. This is because the majority of the significant information in sleep EEG data is recorded below this frequency.

Making train, valid and test splits: The windowed and preprocessed data is currently divided into multiple sets that are needed: (1) the training set, which is used to find the required ConvNet parameters; (2) the validation set, which tracks the training process and decides when to stop it; and (3) the test set, which measures the generalization performance of the desired model.

Creating the neural network: The ConvNet architecture will be explained in this section. One kind of machine learning algorithm utilized for a range of tasks is the neural network. It may be used for regression analysis, pattern recognition, classification, and data-driven decision-making. Artificial neurons, or linked nodes, are arranged into layers to form neural networks. Here are the key components and concepts of a neural network:

- **Artificial Neurons (Nodes):** These are the fundamental processing units of a neural network. Each neuron takes multiple input values, performs computations on them, and produces an output.
- **Layers:** Neurons are organized into layers. The three primary types of layers in a neural network are: Input Layer, Hidden Layers, Output Layer

Train and monitor network & visualizing results: Two critical pieces of the training process are the optimizer and the criterion.

- The optimizer implements the parameter update procedure.
- The criterion, or loss function, is used to measure how well the neural network performs on an example. And lastly, inspecting the results.

1.6 Project planning

1.6.1 Work plan-RACI Matrix

: A project management tool called a RACI matrix may be used to define roles and responsibilities for decisions and activities inside an organization or project. The four essential responsibilities involved in accomplishing tasks are represented by the abbreviation RACI, which stands for Responsible, Accountable, Consulted, and Informed.

Responsible(R)	The person or group responsible for performing the task or completing the work.
Accountable(A)	The person who is ultimately answerable for the completion and success of the task. While they may not perform the task themselves, they oversee its progress, ensure it meets requirements, and make final decisions.
Consulted(C)	Individuals or groups whose input or expertise is sought before the task can be completed. They provide relevant information, guidance, or feedback to support task execution.
Informed(I)	People or organizations that must be updated on the status, choices, or results of the work. Although they are not actively involved in carrying out the assignment, they ought to be informed of any updates.

Table 1.2: RACI Matrix of working plan

Task	Student	Supervisor	External Supervisor
1. Define Research Objectives	**R**	**A**	**I**
2. Literature Review	**R**	**C**	**I**
3. Data Collection Planning	**R**	**C**	**I**
4. Preprocessing Data and feature extraction	**R**	**A**	**I**
5.Model Selection and Training	**R**	**A**	**I**
6.Model Evaluation	**R**	**I**	**I**
7.Performance Analysis	**R**	**A**	**I**
8.Writing Methodology Section	**R**	**I**	**I**
9.Discussion of Findings and Writing Discussion Section	**R**	**A**	**I**
10.Conclusion and Future Work and Writing Conclusion Section	**R**	**I**	**I**
11.Thesis Defense Preparation	**R**	**C**	**I**
12.Thesis Presentation	**R**	**I**	**I**

- Responsible(R)
- Accountable(A)
- Consulted(C)
- Informed(I)

1.6.2 Work plan – Ghatt Chart

	1st Term													2nd Term												
Task Name	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
Topic Selection																										
Communication with Supervisor																										
Acquire field Knowledge																										
Paper Selection																										
Literature Review																										
Pre-defence Preparation																										
Pre-defence Presentation																										
Model Selection																										
Creating Neural Network																										
Model Finalizing																										
Final Model Evalution																										
Thesis Report Writing																										
Defence Preparation																										
Thesis Defence																										

Figure 1.2: Ghatt Chart of working plan

1.7 Project Budget

1.7.1 Overall Budget

Table 1.3: Overall budget for implementing project

SI	Item	Justification	Price(BD)
1	Binding and printing of report	Requires for printing the thesis final report and binding	1000
2	Google Colab Pro subscription	Requires high-memory VMs with 25 GB RAM	1000
Total(BDT)			2000
In word: Two thousand taka only			

1.7.2 Software Budget:

Table 1.4: Software budget for implementing project

SI	Item	Justification	Price(BD)
1	Google Colab Pro subscription	Requires high-memory VMs with 25 GB RAM	1000
Total(BDT)			1000
In word: One thousand taka only			

1.8 Organization of the report:

Fundamental of Convolution Network is briefly discussed in “**Chapter 2”**.Temporal and spatial convolutional layers are specifically designed for sequential data processing, convolutional layers use learnable filters to extract characteristics from raw data. Max pooling layers introduce non-zero gradients for negative inputs, preventing "dying ReLU" problems, while leaky ReLU activation functions downsample feature maps.

Design of Proposed Scheme is briefly explained in “**Chapter 3”**. The proposed CNN model architecture is explained detail in this chapter. The design requirement of the proposed architecture is also described in this chapter. Also the value of hyper-parameter of the proposed model and 3-D design of the proposed architecture is also described.

Methodology of this thesis work is briefly explained in “**Chapter 4”**. Dataset loading, cropping waking period, applying low pass filter, scaling transform to the dataset, extracting 30-s windows, concating all the new dataset and simulation of proposed CNN architecture all topics are explained with detail in this chapter.

Investigations, Results and Discussion are briefly discussed in “**Chapter 5”**. In this chapter various method is applied in the dataset. The performance of these all method are compared with the performance of proposed model. Precision, recall, F1-score values are plotted in graph and observed the performance with the proposed model. All model’s predicted results are compared with the expert manual results by representing hypnogram.

Socio-Economic Impact and Sustainability of this thesis work is discussed in “**Chapter 6”**. In this chapter, impact of the project on societal, health, legal and cultural issues and impact of project on the environment and sustainability of this thesis explained.

Addressing Complex Engineering Problems and Activities are discussed in “**Chapter 7”**. Complex Engineering Problems and Complex Engineering Problems are identified and explained in this chapter.

Conclusions, Limitations and Future Works are discussed in “**Chapter 8”**.

1.9 Summary:

This chapter has covered background of the sleep stage classification using single channel EEG. This chapter has reviewed many literature. The study's goals have been established. Project planning and project budget all are discussed in this chapter. The structure of this thesis book has then been briefly presented.

Chapter 2

Fundamentals of Convolutional Neural Network

2.1 Introduction

One kind of deep learning neural networks called convolutional neural networks (CNNs) is mostly utilized for the analysis of visual vision. Through a technique known as convolution, they are made to automatically and adaptively learn the spatial hierarchies of information from input pictures. Because CNNs are so good at capturing spatial relationships in data, they have become the mainstay of many computer vision jobs and have seen tremendous growth in popularity. Because of the specific layers in their architecture designed for convolving and pooling, they are able to learn patterns locally, generalize features, and provide precise predictions .Convolutional Neural Networks are a potent type of deep learning models that have transformed computer vision by enabling previously unheard-of performance in tasks involving the interpretation and comprehension of visual input.

2.2 Architecture of CNN model

The architecture of a basic Convolutional Neural Network (CNN) typically consists of several key components:

1. **Input Layer:** Raw values are sent to the input layer. This layer's dimensions—such as its width, height, and number of color channels—match those of the input picture.

2. **Convolutional Layers:** CNNs' fundamental building components are these layers. A collection of learnable filters, also known as kernels, are applied to the input picture by each convolutional layer. By exchanging over the input data, these filters produce feature maps that draw attention to particular patterns or characteristics and multiply the data elements. Several filters are used simultaneously to capture various aspects. The number of filters employed determines the resulting feature map's depth.

3. **Activation Function:** To add non-linearities to the network, an activation function is applied element-by-element following each convolutional operation. Many activation function such as ReLU, Leaky ReLu function are widely utilized because of its efficiency in encouraging sparse activation and ease of implementation.

4. **Pooling Layers:** The feature maps produced by the convolutional layers are downsampled by the pooling layers. Two popular forms of pooling procedures are max pooling and average pooling. Pooling makes the network more computationally efficient and less prone to overfitting by lowering the spatial dimensions of the feature maps while maintaining the most crucial information.

5. **Fully Connected Layers:** In most network architectures, these layers are utilized at the very end. Every neuron in the layer above is completely linked to every neuron in these levels. Prediction and the learning of high-level characteristics are aided by fully linked layers. Leaky ReLU and other activation functions frequently come after them.

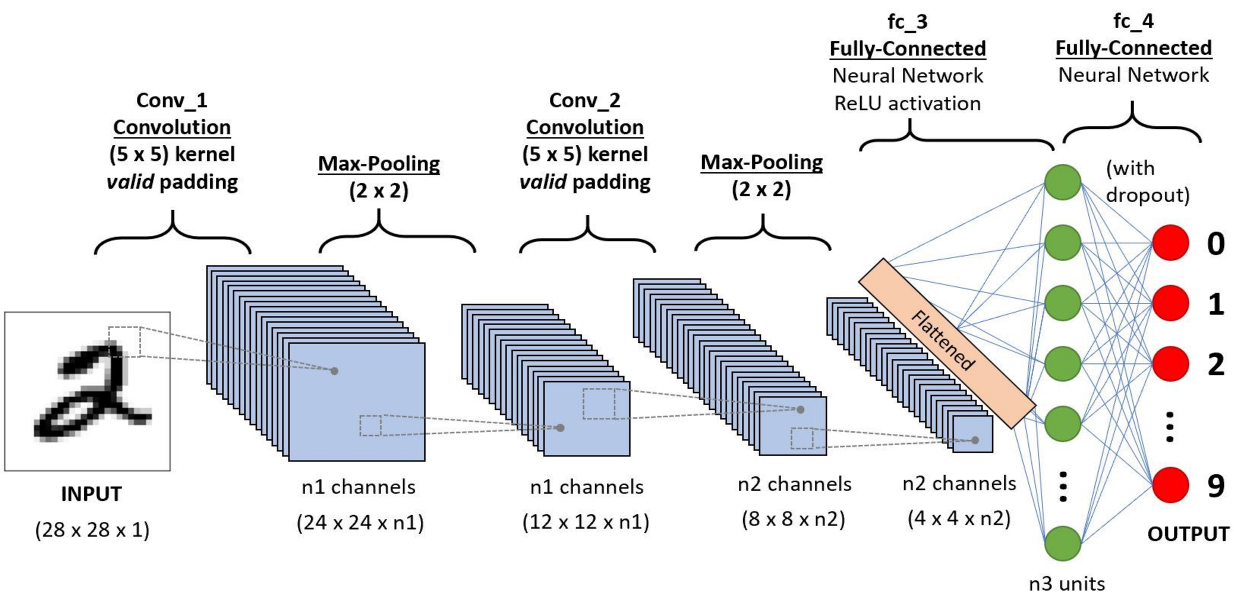


Figure 2.1: Demonstrate a CNN model with Fully Connected Layer

6. **Output Layer:** The network's last layer generates the output predictions. Depending on the job at hand, this layer has a different number of neurons. For instance, the number of neurons in the output layer may correspond to the number of classes, and each neuron would reflect the likelihood that the input data falls into a certain class. It is customary to employ the softmax activation function to transform the raw output values into probabilities.

2.3 Convolution Layer

The convolutional layer is a fundamental building block in Convolutional Neural Networks (CNNs) and plays a crucial role in extracting features from input data. The convolution process is the central component of the convolutional layer. A tiny window known as a filter or kernel is slid over the incoming data during this process. In order to extract relevant features, the filter's learnable parameters are modified during training. The filter multiplies elements-by-elements the values it has and the values of the input data inside the window as it moves across the input. The output feature map has a single value that is the total of the outcomes of these multiplications. The convolutional layer contains a set of filters, each with its own set of learnable parameters. The network learns to adjust the parameters of the filters to detect various patterns and features in the input data. The convolutional layer can be configured with padding and stride parameters. There are many types of convolution layer such as spatial layer, temporal layer etc.

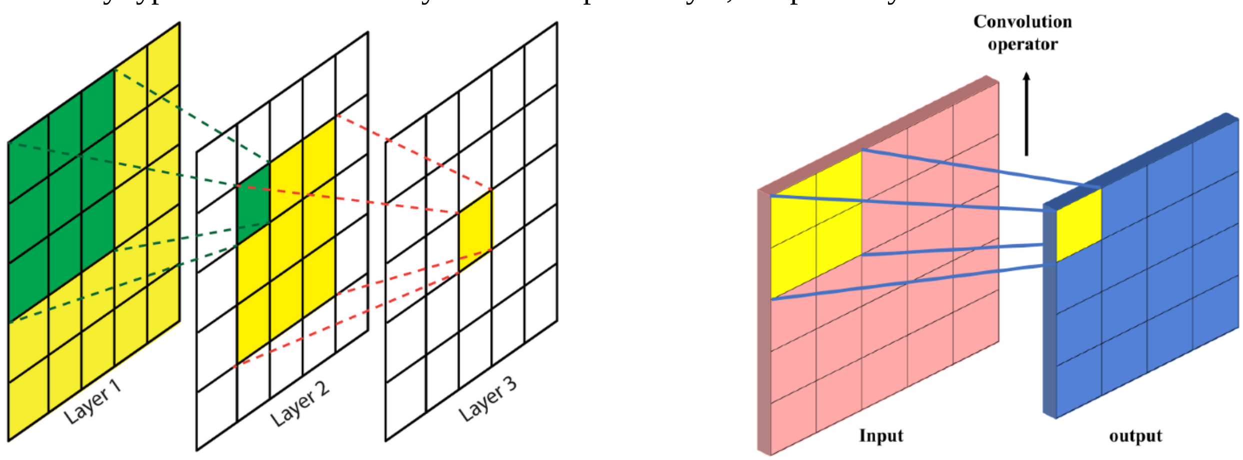


Figure 2.2: Demonstrate Convolution layer and its Convolution operation.

2.3.1 Spatial convolution layer

One essential part of convolutional neural networks (CNNs), which are used to extract characteristics from input data, is the spatial convolutional layer. In this layer, a collection of learnable filters, or kernels, slides over the input and multiplies elements with local areas element by element to create feature maps that emphasize certain patterns or characteristics. Convolutional layers are arranged hierarchically to enable the network to learn progressively more complicated representations of the input data. Early layers capture low-level properties like as edges and textures, while deeper layers capture more abstract elements such as forms and objects.

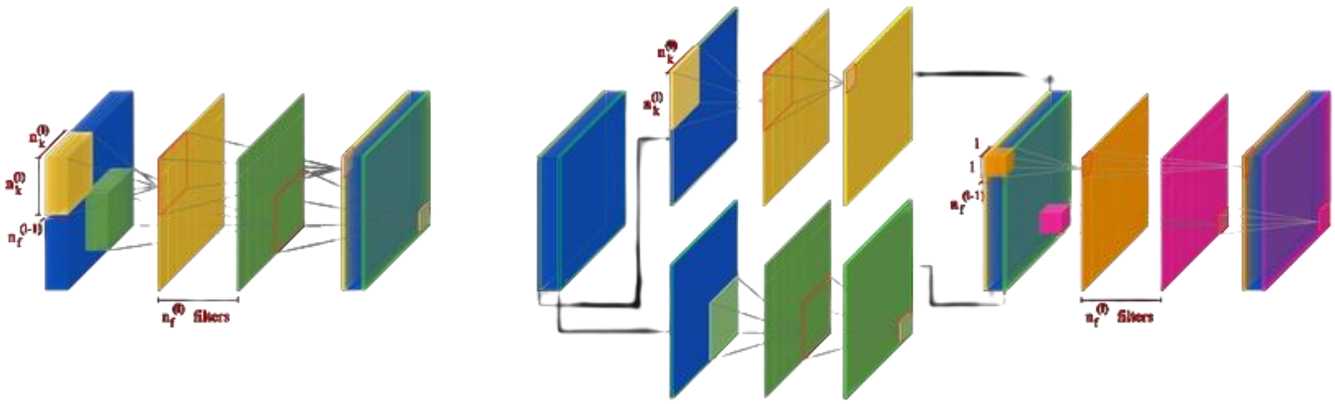


Figure 2.3: Demonstrate Spatial Convolution Layer and its working operation

2.3.2 Temporal convolution layer

One essential part of convolutional neural networks (CNNs) designed to analyze sequential input is the temporal convolutional layer. The temporal axis convolution operation done to the input sequence by the learnable filters (kernels). These filters learn to extract pertinent characteristics at different sizes and places by capturing temporal patterns and dependencies as they go through the sequence.

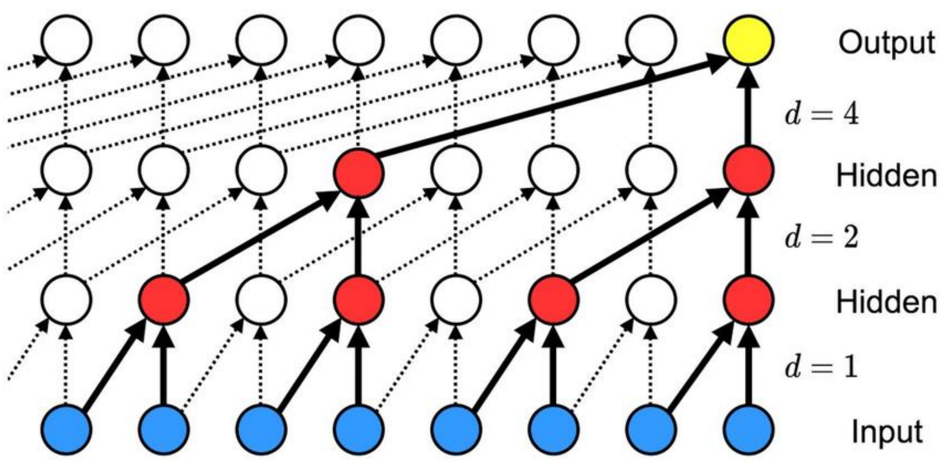


Figure 2.4: Demonstrate the structure of Temporal Convolution Layer

2.4 Max pooling layer

The max pooling layer is a crucial component of convolutional neural networks (CNNs), typically applied after convolutional layers. It reduces the spatial dimensions of the feature maps produced by convolutional layers through downsampling in order to retain the most significant information. Using max pooling, the input feature map is split into non-overlapping rectangular portions known as kernels or pooling windows. The greatest value is taken from each area. Consequently, each pooling window's highest value is included in the output feature map, which downsamples the input. Two hyperparameters are usually involved in the max pooling layer: the stride and the size of the pooling windows.

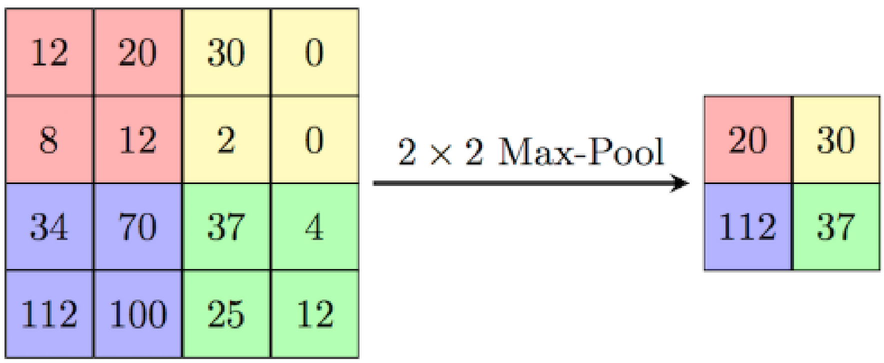


Figure 2.5: Demonstrate the operating principle of Max pooling Layer

2.5 Leaky ReLu activation function

The Leaky ReLU (Rectified Linear Unit) activation function is a variant of the traditional Rectified Linear Unit (ReLU) activation function that is widely used in neural networks, such as Convolutional Neural Networks (CNNs). Unlike the classic ReLU function, the Leaky ReLU avoids the "dying ReLU" problem, in which neurons may go inactive during training as a result of repeatedly negative inputs, which sets negative values in the input to zero (f(x) = max(0, x))
The Leaky ReLU function is defined as follows:

$$\text{Leaky ReLU} = f(x) = \max(0.01*x, x) \quad \ldots\ldots\ldots\ldots\ldots\ldots..(2.1)$$

where x represents the input to the function.

Advantages of Leaky ReLu activation function:

- The "dying ReLU" issue, which might arise with the regular ReLU function, is addressed by the Leaky ReLU.
- It preserves the ReLU sparsity trait, which allows for certain neurons to stay dormant and increases computational efficiency.
- It has a straightforward implementation, with only one additional hyperparameter (α) to tune.

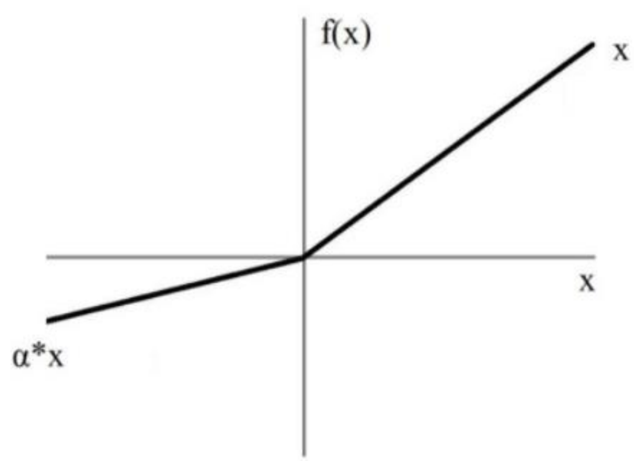


Figure 2.6: Graphical representation of Leaky ReLu activation function where α = 0.01

2.6 Summary

While temporal and spatial convolutional layers are specifically designed for sequential data processing, convolutional layers use learnable filters to extract characteristics from raw data. Max pooling layers introduce non-zero gradients for negative inputs, preventing "dying ReLU" problems, while leaky ReLU activation functions downsample feature maps, encouraging translation invariance. Convolutional Neural Networks are made up of four fundamental parts, which work together to provide efficient feature extraction, dimensionality reduction, and non-linear learning—all of which are essential for sequencing processing.

Chapter 3
Design of the Proposed Scheme

3.1 Introduction

This “Design of Proposed Scheme” provides an overview and sets the context for the proposed scheme. This chapter aims to introduce with the proposed CNN model architecture, the requirements to design the proposed architecture, value of all hyper-parameter of the model. Also in this chapter, 3-D design of the proposed architecture will be introduced.

3.2 Proposed CNN Model Architecture

Figure 3.1 shows the proposed CNN Model architecture which has two Convolution Layer (Conv), two MaxPooling Layer (MP), Two Leaky ReLU activation Layer and two Fully Connected Layer (FC).

After extracting 30 s window, the signal data is represented in 2-D array where each row represent the number of channel and each column represent sample point in time. Then this 2-D array is applied into first layer of convolution network. The network's first layer is a time-independent linear process that produces a collection of virtual channels by combining the original input channels in a linear fashion. It carries out a spatial filtering that is motivated by the categorization problem at hand. In this experiment, the first layer is a square matrix multiplication since the number of virtual channels was adjusted to equal the number of input channels. It is possible to create this first layer, which is based on spatial filters, using a 2D valid convolution with shape (C, 1) kernels. Next, in order to stabilize the learning process, aid prevent over-fitting, and make the network less sensitive to the initialization of weights and biases, the dimensions are permuted and batch normalization is performed to the spatial layer., making training more stable and robust.

After that, two blocks of temporal convolution are performed, and then non-linearity and max pooling. Each block applies a rectified linear unit, Leaky ReLU function, after convoving its input signal eight times using estimated kernels of length 64 and stride 1.

$$\mathbf{f(x) = max(\alpha * x , x)} \quad \ldots\ldots\ldots\ldots\ldots\ldots..(3.1)$$

Where alpha is a small positive value (usually 0.01).

A max pooling layer with a size of 16 and no overlap is then used to decrease the outputs along the time axis. Prior to sending the feature maps to the thick layers, they need to be flattened into a 1D vector. The spatial and temporal data are structured using this method so that they can be included in the thick layers. The final stage entails feeding the two convolution blocks' output via a dropout layer, which randomly stops updating 25% of its output neurons at each gradient step. The resultant outputs are then sent into a final layer with five neurons and a softmax non-linearity to get a probability vector that adds to one. This process creates the feature space of dimension D. A softmax classifier is the name given to this last layer. Let $\boldsymbol{\alpha} \in \mathbb{R}^5$ be the pre-activation of the last layer. The output of the network is a vector $\wp \in$ y. $\wp$ is obtained as:

$$\wp_i = \exp(\alpha_i) / \sum_{j=1}^{5} \exp(\alpha_i) \quad \text{..................(3.2)}$$

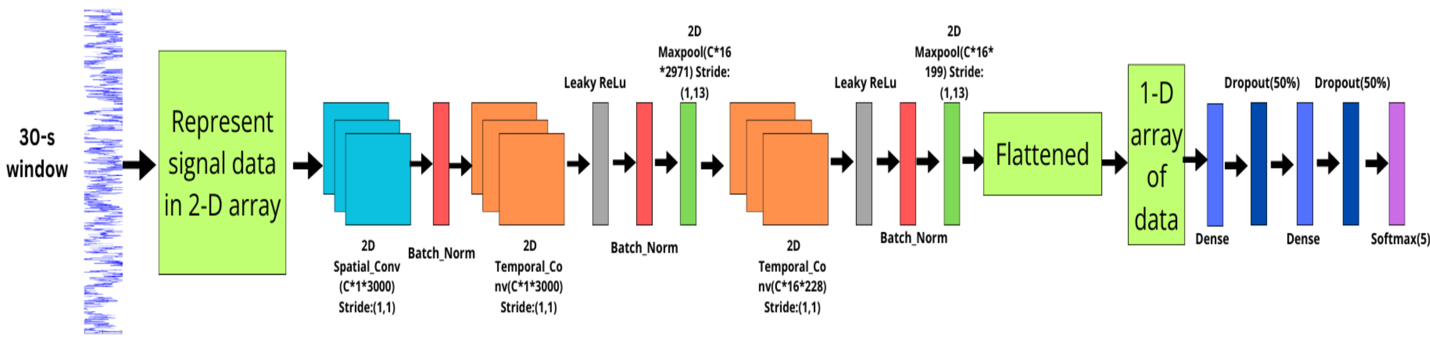


Figure 3.1: Structure of the proposed CNN model

3.3 Design Requirements of the proposed architecture

Table 3.1: The feature extractor's detailed design for C EEG channels with time series of length T

	Layer	Layer type	Filters	Size	Stride	Output dimension	Activation	Mode
Features Extractor	1	Input						
	2	Reshape						
	3	Convolution2D	C	(C, 1)	(1, 1)	(1,T,C)	Linear	
	4	Batch_Norm				(1, T, C)		
	5	Permute				(C, T, 1)		
	6	convolution2D	8	(1, 64)	(1, 1)	(C, T, 8)	ReLu	Same
	7	Batch_Norm				(C, T, 8)		
	8	maxpooling2D		(1, 16)	(1, 16)	(C, T // 16, 8)		
	9	convolution2D	8	(1, 64)	(1, 1)	(C, T // 16, 8)	ReLu	Same
	10	Batch_Norm				(C, T // 16, 8)		
	11	maxpooling2D		(1, 16)	(1, 16)	(C, T // 256, 8)		
	12	Flatten				(C* (T // 256)* 8)		
	13	Dropout (50%)				(C* (T // 256)* 8)		
Classifier	14	Dense				5	Softmax	

3.4 Hyper-parameter of the Proposed Architecture

Hyperparameters are settings made before a machine learning model is trained, and they are not changed while the model is being trained. They have an impact on how the model learns from the data and how it behaves. Hyperparameters need to be set by the practitioner prior to training starting, in contrast to model parameters, which are determined by the data. Table shows the hyper-parameter for the proposed model architecture

Table 3.2: List of all Hyper-parameter of the proposed CNN model

Hyper-parameter	Values
Number of classes	5
Number of convolution channels	16
Time convolution size per second	1
Max pooling size per second	0.25
Input size per second	30
Pad size per second	0.5
Train batch size	56
Validation batch size	256
Number of processes to use for the data loading process	0
Learning rate	0.001
Weight decay rate	0
Number of epochs	20
Dropout	0.3

3.5 Summary

The proposed CNN model architecture is explained detail in this chapter. The design requirement of the proposed architecture is also described in this chapter. Also the value of hyper-parameter of the proposed model and 3-D design of the proposed architecture is also described.

Chapter 4
Methodology

4.1 Introduction

The categorization of sleep phases based on single-channel electroencephalography (EEG) data is an essential component in the diagnosis of different sleep disorders and the study of the dynamics of sleep patterns. These phases include waking, rapid eye movement (REM) sleep, and non-REM sleep, which are further separated into stages N1, N2, and N3, which correspond to varying sleep depths. This methodology offers a methodical way to categorize different phases of sleep using just single-channel EEG data. Preprocessing, feature extraction, classification, post-processing, assessment, and iterative improvement processes are all included. This methodology seeks to precisely identify sleep phases by utilizing machine learning algorithms and signal processing techniques. This will help with the diagnosis of sleep disorders and further our understanding of sleep physiology.

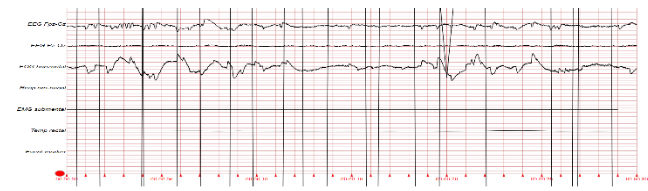
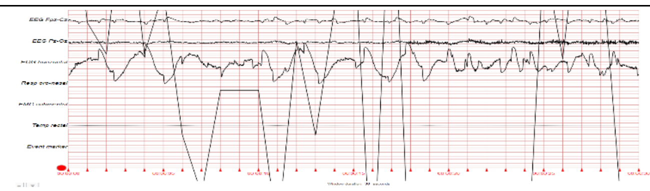
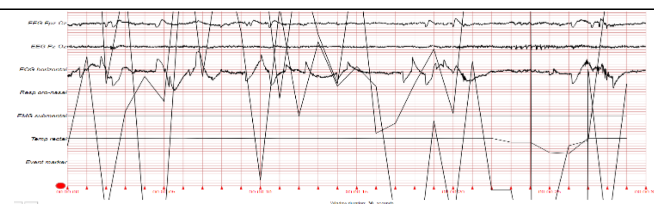
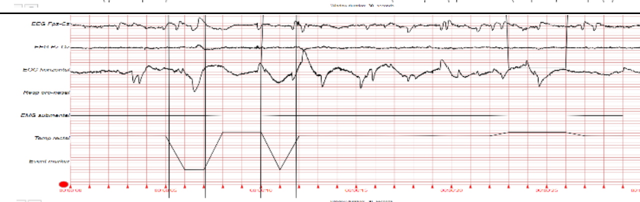
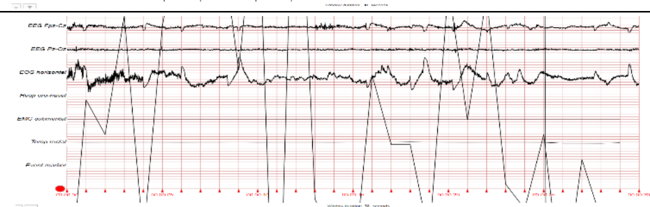
4.2 Dataset acquisition

The National Academies of General Medical Sciences and Biomedical Imaging and Bioengineering support the resource website PhysioNet. [31]. The Sleep Cassette (SC) subset of the Sleep-EDF Expanded dataset from 2013 was referred to as Sleep-EDF.197 whole-night PolySomnoGraphic sleep recordings with event markers, chin EMG, EEG, and EOG are available in the sleep-edf database.Whole-night polysmnographic sleep recordings, comprising an event marker, horizontal EOG, submental chin EMG, and EEG (from the Fpz-Cz and Pz-Oz electrode placements), are included in the *PSG.edf files. The *Hypnogram.edf files have annotations for the appropriate PSG sleep patterns. These patterns, or hypnograms, are composed of the following sleep phases: W, R, 1, 2, 3, 4, and M (movement time). A study conducted in 1987–1991 on the impact of age on sleep in 25–101-year-old healthy Caucasian people produced 153 SC* files (SC = Sleep Cassette)[32]. Two PSGs, each lasting roughly 20 hours, were recorded in the patients' homes over the course of the two subsequent day-night cycles. Every EOG and EEG signal sample was recorded at 100 Hz. Below table describes the data acquisition details.

Table 4.1: Data Acquisition Details (Sleep-EDF Dataset Expanded)

Patients	Male/Female	Patients age	Sleep rule	EEG channel	Sample freq. (EEG/EOG)	Sample freq. (EMG)
78	30/48	25-101	R & K	Fpz-Cz/ Pz-Oz	100 Hz	1Hz

Table 4.2: Dataset description for some recorded signal

Serial no	Database	Record	Image of recorded signal
1	sleep-edfx/1.0.0	sleep-cassette/SC4001E0-PSG.edf	
2	sleep-edfx/1.0.0	sleep-cassette/SC4012E0-PSG.edf	
3	sleep-edfx/1.0.0	sleep-cassette/SC4032E0-PSG.edf	
4	sleep-edfx/1.0.0	sleep-cassette/SC4052E0-PSG.edf	
5	sleep-edfx/1.0.0	sleep-cassette/SC4072E0-PSG.edf	

4.3 Block Diagram of Proposed Method

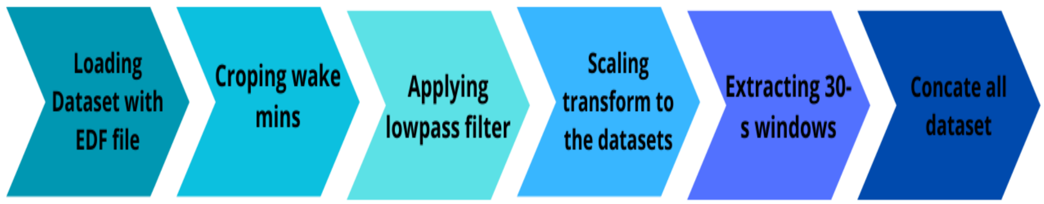


Figure 4.1: Demonstrate block diagram of Methodology

- **Croping wake mins:** The removal of waking periods from the start and finish of the raw data is one of the most important preprocessing techniques used in this part. Cropping these non-sleep states allows researchers to concentrate their investigations on times when real sleep is occurring, which improves the precision and applicability of spectral analysis or sleep stage categorization that follows. Furthermore, cropping these non-sleep situations incorporates sleep stage annotations from the supplied files in a smooth manner, guaranteeing that the EEG data segments and associated sleep phases are properly aligned.

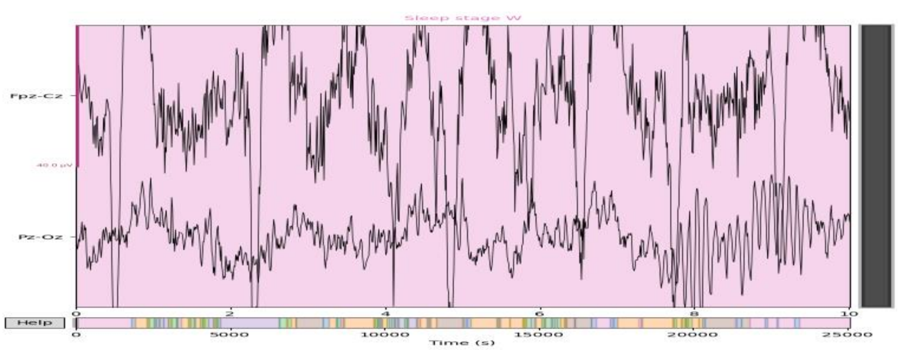


Figure 4.2: Demonstrate Crop portion of sleep event

- **Applying low pass filter:** The majority of useful information in sleep EEG data is found below 30 Hz. Therefore, to mitigate the impact of higher frequency noise, need to apply a lowpass filter with cutoff frequency of 30 Hz to recordings. With a lower frequency cutoff of None, a highpass filter that keeps frequencies above 0 Hz is indicated. With the highest frequency cutoff set to 30, frequencies up to 30 Hz are to be kept.

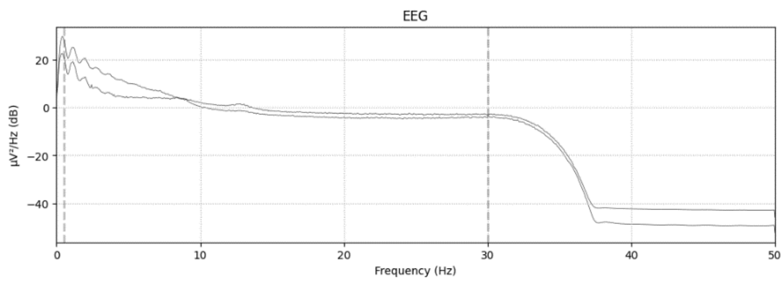


Figure 4.3: Cutoff frequency of 30 Hz of the recordings.

- **Scaling transform to datasets:** When a scaling transform is applied to EEG datasets, every EEG channel in a 30-second frame is guaranteed to have a mean of 0 and a standard deviation of 1. In order to standardize the EEG data across many channels and time intervals, this normalization technique is crucial. Any biases in the mean and variance of the EEG signals are eliminated by the scaling transform, which centers the data around zero and scales it to have a standard deviation of 1. It is now easier to compare and analyze EEG signals fairly across channels and time periods because to this normalization, which also makes it simpler to spot trends and abnormalities in the data. Additionally, scaling the data helps mitigate the effects of outliers and amplifies smaller variations.

- **Extracting 30-s windows:** A critical step in the EEG data processing method was extracting 30-second window from the continuous EEG recordings. The segmentation of the EEG data into manageable units, which made it possible to analyze particular temporal periods in detail, was made possible by this technique. The first step in the window extraction process was calculating the length of each window, which was determined to be 30 seconds based on accepted practices in EEG research. The entire time of the EEG recordings was then divided by the window length to get the total number of window. A 30-second slice of the EEG signal was taken during each repetition, beginning at the signal's present location. The retrieved windows were saved in a suitable data structure, like a list or array, to make data administration and accessibility easier. This arrangement made it possible to efficiently retrieve and work with specific windows in later phases of data processing and analysis.

- **Concating all datasets:** Concate all the dataset into a single dataset using ConcatDataset library. The concatenated dataset enhances flexibility in analysis and modeling endeavors. Also concatenating datasets ensures consistency in data representation and processing.

4.4 Making train, valid and test split

After preprocessing and windowing data, the dataset has been splited into traning, validation and testing dataset by following this below procedure:

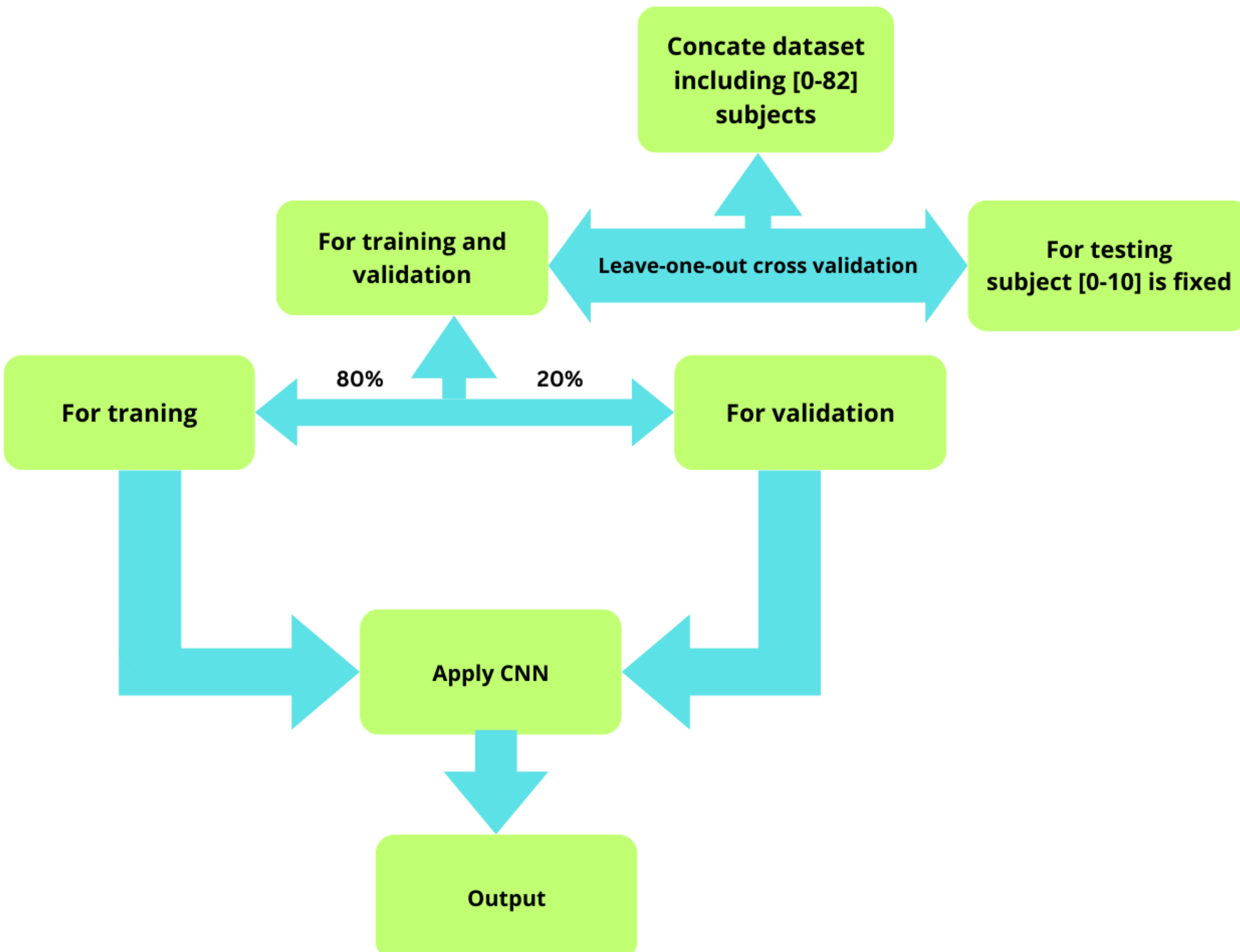


Figure 4.4: Demonstrate the process of making train, test and valid split in a block diagram

From figure 4.4, it is seen that, concate dataset has been split into training and testing dataset using leave-one-out cross validation with n-number of group. This ensures that each group is used for testing exactly once. Subject [0-10] is fixed for testing dataset and remaining datasets are used for training and validation datasets.
Then, take 20% of the remaining dataset for the validation and 80% of the remaining dataset for the training. Ensures that the split is stratified by subject ID. This means that the proportion of subjects from each category is preserved in both the training and validation sets.
After completing the split, Number of examples in each set:
Training: 101936
Validation: 23209
Test: 19148
Visualization of the splitting sample is given below:

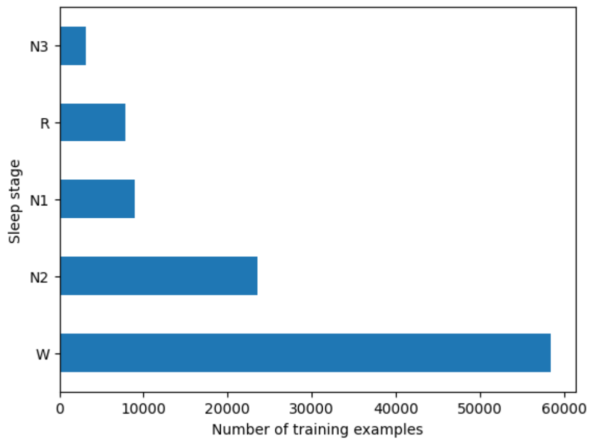


Figure 4.5: Demonstrate the class distribution of training samples where N3 sleep stage has 5673 samples, N2 sleep stage has 22163 samples, N1 sleep stage has 8719 samples, REM sleep stage has 7125 samples, W sleep stage has 58256 samples.

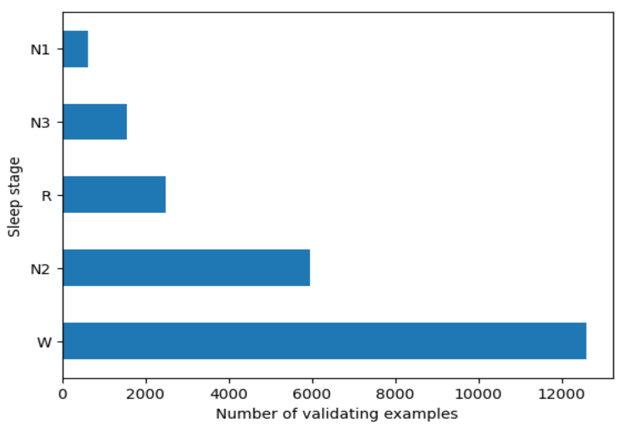


Figure 4.6: Demonstrate the class distribution of validation samples where N3 sleep stage has 1773 samples, N2 sleep stage has 5826 samples, N1 sleep stage has 593 samples, REM sleep stage has 2219 samples, W sleep stage has 12798 samples.

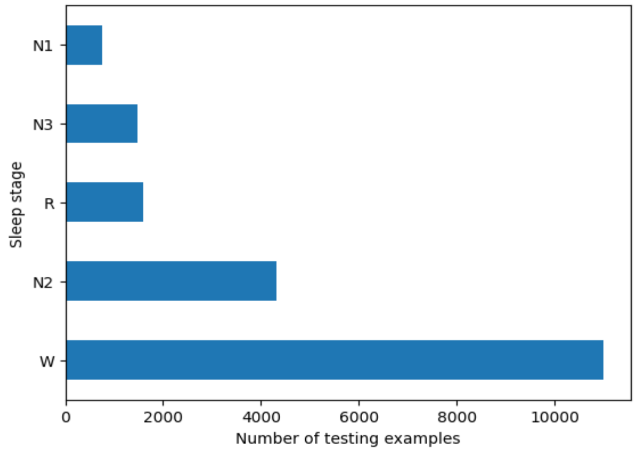


Figure 4.7: Demonstrate the class distribution of testing samples where N3 sleep stage has 1483 samples, N2 sleep stage has 4312 samples, N1 sleep stage has 764 samples, REM sleep stage has 1594 samples, W sleep stage has 10995 samples.

Here, from figure 4.5, 4.6, 4.7 it has seen that classes of sleep stage are imbalanced. Some classes being significantly more prevalent than others. This class imbalance can introduce biases during model training, potentially leading the model to favor the majority classes and perform poorly on the minority classes. To mitigate this issue, class weights are calculated to assign higher importance to minority classes and lower importance to majority classes during training. Weights that are inversely proportional to the class frequencies. Less frequent classes are assigned higher weights, and more frequent classes are assigned lower weights. The model learns to prioritize accurately categorizing examples from the minority classes by including these class weights into the training process. This improves performance across all classes and improves overall generalization.

These are the computed weights of following channel:
N1: 0.34891068
N3: 2.26751196
R: 0.86617666
N2: 6.51764706
W: 2.59742642

4.5 Working Principle of Proposed Model Architecture

- **Input Data Representation**: The EEG signal, which is commonly represented as a time-series of voltage values collected over time, is the input used in the CNN model. The electrical activity of the brain at a certain time is the input data for single-channel EEG. The windowing data is represented by 2-D array and then apply this 2-D data into 2-D convolution layer.

- **Convolution Layers:** The suggested CNN model is composed of two temporal convolution layers, which extract temporal information from the EEG data, and one spatial convolutional layer, which learns spatial patterns or features from the input EEG signal. Every convolutional layer performs convolution operations on the input signal using a collection of learnable filters, or kernels, to create feature maps. By identifying local patterns in the EEG data, these filters capture significant spatial and temporal features that are crucial for classifying sleep stages. In the suggested model, batch normalization is also implemented. During training, batch normalization stabilizes the distributions of inputs across layers. Because of its reduction in sensitivity to initiation and parameter size, this normalization speeds up convergence. Higher learning rates are made possible and model generalization is improved. CNNs get more resilient with batch normalization.

- **Activation function:** By adding non-linearity to the CNN model, activation functions help the model understand and depict the correlations seen in the EEG data. In suggested convolutional layers, Leaky ReLU is utilized to induce non-linearity by effectively thresholding the output feature maps by substituting zeros for negative values.

- **Pooling Layers**: The pooling layers downsample the feature maps that were obtained from the convolutional layers. Pooling helps retain the most important data while lowering the spatial dimensions of the feature maps. Max pooling effectively preserves the most prominent features by selecting the biggest value from each local area of the feature map.

- **Flattening and Fully Connected Layers**: A one-dimensional vector is created by flattening the feature maps following many convolutional and pooling layers. One or more fully linked layers, often referred to as thick layers, get this flattened representation and use it to teach them how to categorize the data taken from the EEG signal into various sleep phases. The model can learn intricate feature combinations because fully linked layers link every neuron in one layer to every other layer's neuron. Two 50% dropout dense layers are used, meaning that 50% of the activations are randomly set to zero during training after each 50% dropout dense layer. This is a method for keeping neural networks from overfitting. Fully connected layers, or dense layers, have connections between every neuron in one layer and every other layer. To identify complex patterns in the data, dense layers are employed. By randomly setting a portion of the input units to zero after each training phase update, the regularization approach known as "dropout" helps to prevent overfitting. A 50% dropout in this instance indicates that, on average, half of the dense layer's units are removed during training. During training, half of the units in the dense layer are randomly set to zero at each update, forcing the network to learn more robust features.

- **Output Layer and Classification:** The CNN model's output layer is made up of neurons that represent the various stages of sleep, such as waking, REM, N1, N2, and N3. The raw output scores are converted into probabilities via the output layer's activation function (softmax for multi-class classification), which shows the likelihood of each sleep stage. Using gradient descent and backpropagation, the model modifies its parameters (weights) during training in order to reduce the discrepancy between the projected probabilities and the actual sleep stage labels.

- **Training and Optimization:** The EEG signal is used as the input and the matching sleep stage label is used as the output when training the CNN model with labeled EEG data. During training, the model is iteratively given batches of EEG data, the loss (error) between the predicted and real labels is calculated, and the model parameters are updated to minimize this loss. The model parameters are effectively adjusted by the application of optimization techniques like Adam. The difference between the actual distribution of class labels and the expected class probabilities is effectively measured by cross-entropy loss, which is why it is utilized.

- **Evaluation and Performance Metrics:** After training, the CNN model's performance is assessed using a different validation or test dataset. The model's performance in accurately classifying distinct stages of sleep is measured by computation of performance measures, including accuracy, precision, recall, and F1-score. A detailed assessment of the model's classification performance is conducted using confusion matrices.

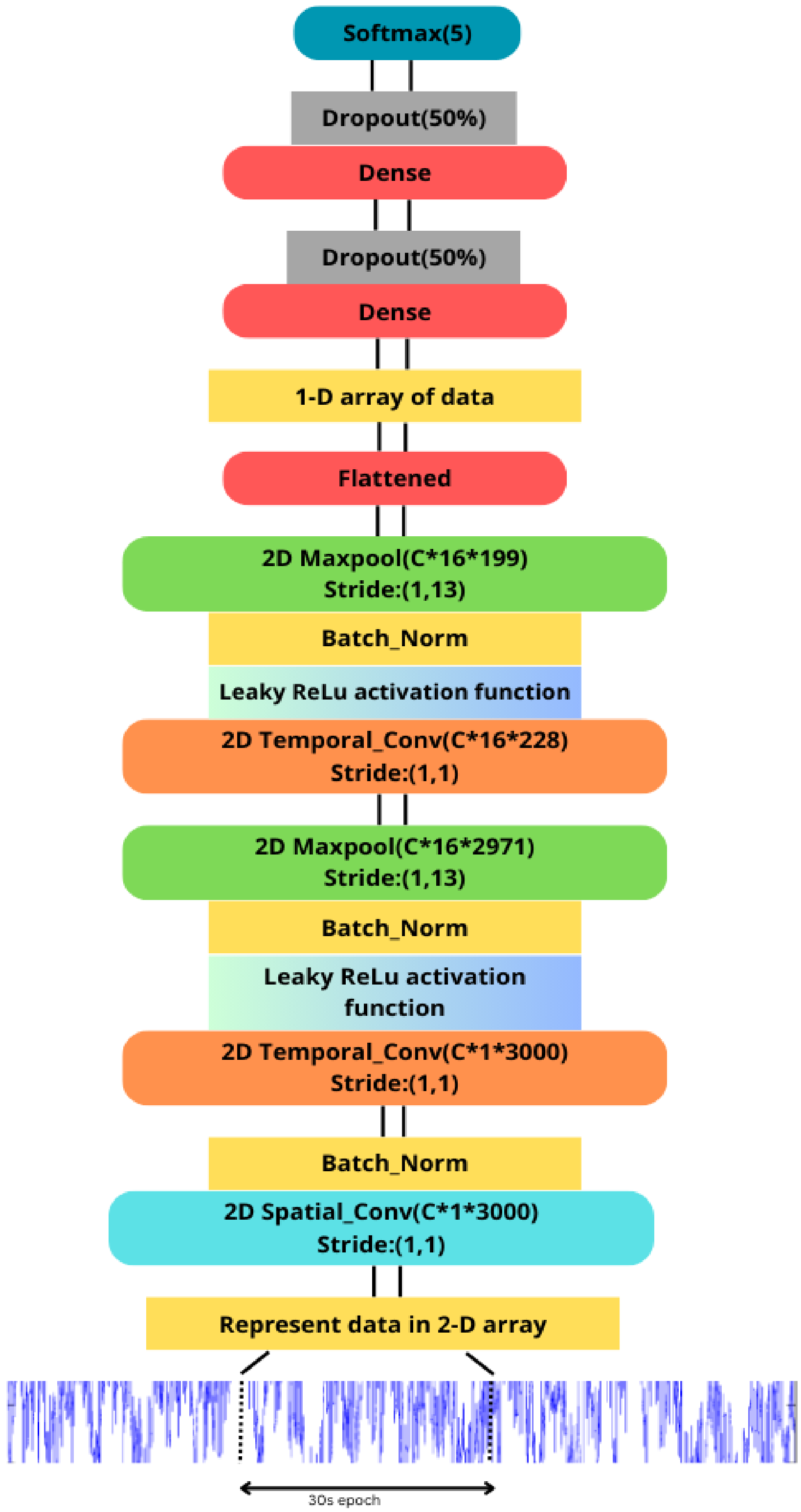


Figure 4.8: Demonstrate the working principle of the proposed CNN model

4.6 Summary

This chapter describes the methodology of the thesis work in detail. At first, I downloaded the dataset from the Physionet website. Then, remove the waking periods from the start and finish of the raw data. Then, to mitigate the impact of higher-frequency noise, we need to apply a low-pass filter with a cutoff frequency of 30 Hz to recordings. Then normalize the data to have a mean of 0 and a standard deviation of 1. Then extract a 30-second window from the continuous EEG recordings and concatenate all the datasets into a single dataset. Then split the dataset into training, testing, and validation datasets and compute the weight to balance the classes. Finally, the proposed CNN is created, and we apply this training, testing, and validation dataset to the proposed model for classifying sleep stages.

Chapter 5
Investigations, Results and Discussion

5.1 Introduction

Convolution Neural Network (CNN) is one of the effective method for classifying sleep stage. CNNs effectively classify sleep stages by automatically extracting spatial hierarchical features from EEG signals, robustness to noise, and scalability for processing large datasets in parallel.

In CNN, The 153 SC* files (SC = Sleep Cassette) which were obtained in a 1987-1991 study of age effects on sleep in healthy Caucasians aged 25-101, the subjects to use can be in the range of 0-82 (inclusive) are applied.

Many libraries such as mne, torch, matplotlib, scikit-learn, pandas are used to perform this work. To run CNN model high configuration computer is needed. Kaggle supplies this feature easily. It supplies 30 GB RAM and GPU T4 x2.

5.2 Classification Models

In this thesis work, four techniques are selected to classify the extracted features and the classification accuracy is obtained: support vector machine (SVM), convolution neural network (CNN), backpropagation neural network (BPNN), and decision tree (DT) algorithms.

Often used for supervised classification problems, SVM is a dependable classifier [33]. All characteristics are converted into 0–1 sequences using the z-score standardization method before SVM classification is applied. Grid search is utilized to modify the hyperparameters, and a linear function is selected as the kernel function. With three layers—an input layer, an implicit layer, and an output layer—the BP neural network algorithm is the most widely used neural network machine learning technique. For signaling purposes, neural nodes are used to connect one layer to the next [34]. All characteristics are standardized using the min–max normalization approach inside the range [0, 1] before being classified using a BP neural network. The number of neural nodes in the input layer had to be changed based on the quantity of feature values in various sample sets because this thesis study separated sleep into five periods. In the output layer, there are five nodes, there are twenty nodes in the implicit layer, and the learning efficiency is set at 0.1. The DT technique is a categorization rule that is based on examples and is produced by induction on a chaotic set of situations, is an example of an inductive learning algorithm [35]. The decision tree classification problem can be solved in two steps: first, a learning training set generates the decision tree classification model; second, the model is applied to categorize unknown categories of samples. In this study, the C4.5 decision tree technique is used, and the information gain rate is represented by the splitting index.

5.3Validation of Classification Models

Examining the model's prediction effect after training yields four main categories: true positive, which indicates a positive and positive prediction; fake positive, which indicates a positive but negative prediction; true negative, which indicates a negative but positive prediction; and fake negative, which indicates a negative but positive prediction. The evaluation measures employed for the classifier were f1-score, accuracy, precision, and recall [36].

(1) The most basic indicator is accuracy, which is determined by dividing the total number of observations by the number of observations that were correctly predicted:

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad \text{……………} \ (5.1)$$

(2) The percentage of real positives among the samples that were expected to be positive is known as precision.

$$\text{Precision} = \frac{TP}{TP+FP} \quad \text{……………} \ (5.2)$$

(3) Recall represents the proportion of all real positive samples that are expected to be positive:

$$\text{Recall} = \frac{TP}{TP+FN} \quad \text{…………….} \ (5.3)$$

(4) F1-score is a more balanced index between precision and recall:

$$\text{F1-score} = \frac{TP}{TP + \frac{FN+FP}{2}} \quad \text{………… (5.4)}$$

(5) The percentage of correctly categorized cases relative to all instances is known as accuracy.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad \text{…………… (5.5)}$$

(6) Specificity—also referred to as the true negative rate—means the percentage of real negative cases that the model properly detects.

$$\text{Specificity} = \frac{TN}{TN+FN} \quad \text{…………. (5.6)}$$

5.4 Visualizing the learning curve of proposed CNN model

A learning curve shows, in graphical form, how a model's performance increases with more data sets or with the training process. The quantity of training data or the number of training iterations (epochs) on the x-axis is usually plotted against some performance parameter (such as accuracy, error rate, loss, Cohen's kappa) on the y-axis.

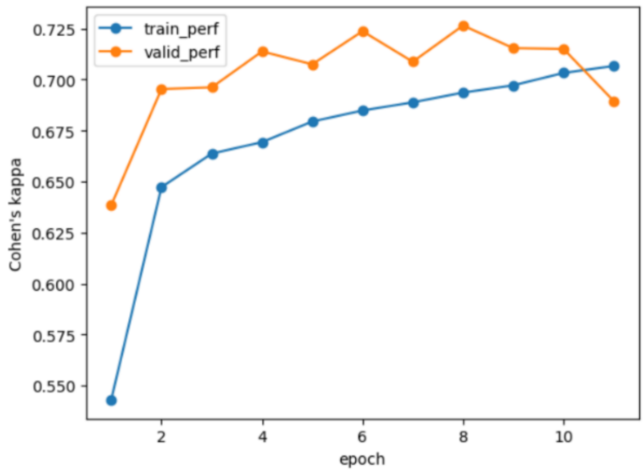


Figure 5.1: Training Accuracy Curve.

From Figure 5.1 it is seen that, as the epoch number is increasing, the cohen's kappa is increased. Similarly, from Figure 5.2 it is seen that, increasing epoch number Training loss is decreased.

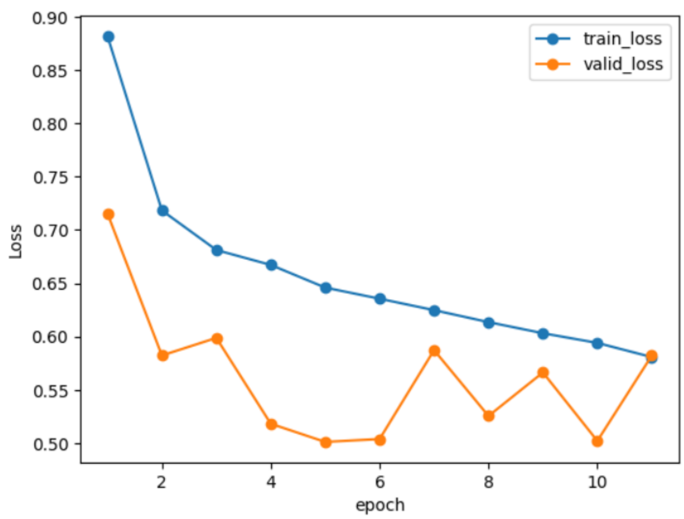


Figure 5.2: Training Loss Curve.

5.5 Calculate results for different model

5.5.1 SVM Model: Results and Evaluation

SVM model is used for automatic staging of EEG sleep data. Every dimensional feature is chosen. Table 5.1 displays the results of adjusting the model parameters, which indicate that phase W had the highest accuracy and phase N1 had the lowest accuracy, with an overall accuracy of 79.45%. Figure 5.3 displays the related confusion matrix. Figure 5.3 illustrates how muddled the REM and N1 stages were most likely. The N2 stage contains the majority of the incorrect N3 stage predictions, whereas the N2 stage's incorrect predictions are dispersed among the N3, N1, and REM stages. The W stage's incorrect predictions are primarily concentrated.

Table 5.1: Comparison of expert manual staging results with SVM model staging outcomes for all features[24]

Sleep stage	Training Samples	Test Samples	Correct Samples	Precision	Recall	f1-Score	Accuracy
N3(Deepest stage)	1326	345	321	0.8992	0.9304	0.9145	89.01%
N2(Deeper stage)	1601	428	336	0.8276	0.7850	0.8058	81.2%
N1(Lighter stage)	1639	390	255	0.7143	0.6538	0.6827	69.7%
REM(Rapid Eye Movement)	1547	391	316	0.7215	0.8082	0.7624	78.08%
W(Wake stage)	1643	386	360	0.9424	0.9326	0.9375	92.7%

Overall Accuracy: 79.45%

Figure 5.3: Confusion matrix of SVM model staging results.

Figure 5.4 shows the expert manual staging results with the SVM model staging results. The SVM model produces the prediction label, but the real label is the outcome of expert staging done by hand. The findings shown in the image indicate that the sleep stages that experts have labeled matched very well with the results of the proposed CNN model.

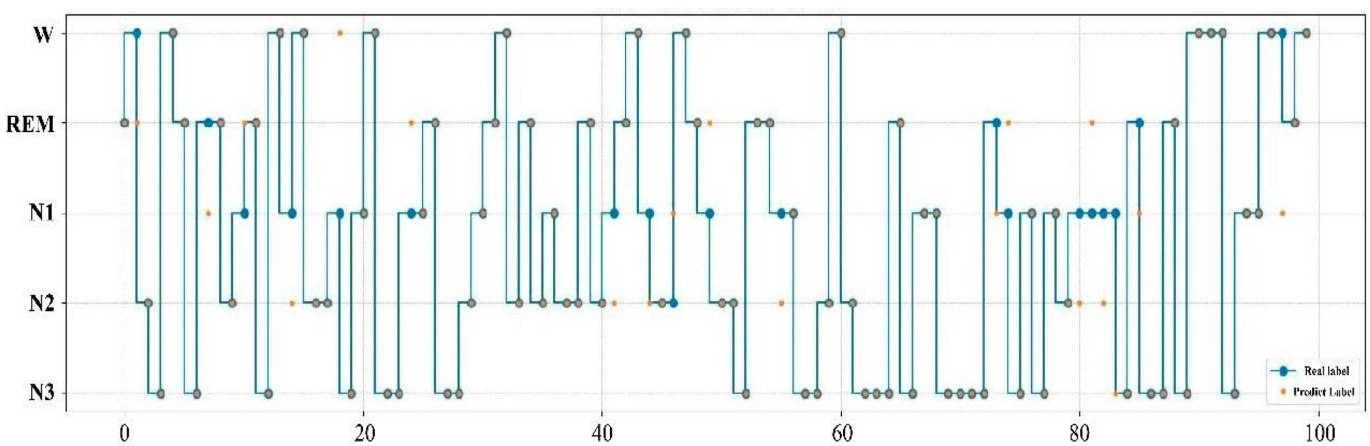


Figure 5.4: Demonstrate the comparison between expert manual staging compared to SVM model staging. The SVM model produces the predicted label, whereas real label is the outcome of expert manual staging.

5.5.2 BPNN Model: Results and Evaluation

The sleep EEG is automatically staged using a BPNN model. All dimensional characteristics is chosen. Table 5.2 illustrates that stage W had the greatest precision rate at 90%, followed by stage N2 at 84%. With an overall accuracy of 78.33%, the N1 stage have the lowest f1-score rate at 66%, while the N3 and REM phases have f1-score rates close to 75%. Figure 5.5 displays the related confusion matrix. It is evident that the REM and N1 periods are when people get confused between the two the most; The N3 period's incorrect prediction is mostly concentrated in the N2 period; the N2 period, N1 period, and REM period's incorrect predictions are more dispersed, suggesting that these three periods are easily mistaken for one another; and the W period's incorrect prediction is mostly concentrated in the N1 period.

Table 5.2: Comparison of expert manual staging results with BPNN model staging outcomes for all features[24]

Sleep stage	Training Samples	Test Samples	Correct Samples	Precision	Recall	f1-Score	Accuracy
N3(Deepest stage)	1310	331	332	0.769	0.963	0.855	88.7%
N2(Deeper stage)	1640	410	288	0.839	0.673	0.747	71.07%
N1(Lighter stage)	1601	350	255	0.656	0.654	0.655	68.9%
REM(Rapid Eye Movement)	1596	374	273	0.752	0.698	0.744	74.04%
W(Wake stage)	1690	397	372	0.901	0.964	0.931	93.2%

Overall Accuracy = 78.33%

Figure 5.5: Confusion matrix of BPNN model staging results.

Figure 5.6 illustrates how the BPNN model staging results have been combined with the expert manual staging findings. The findings shown in the image indicate that the sleep stages that experts have labeled matched very well with the results of the BPNN model.

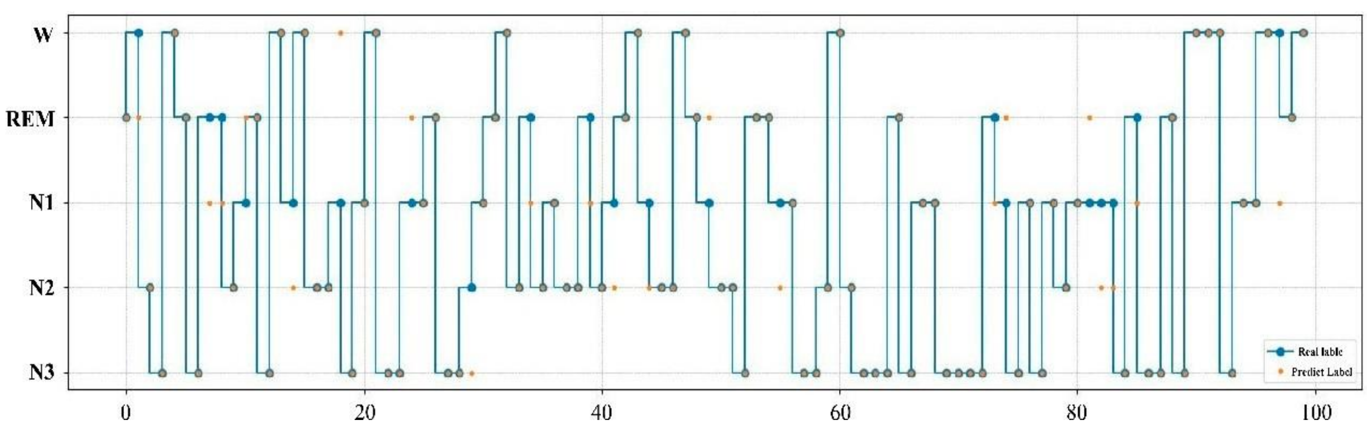


Figure 5.6: Demonstrate a comparison between the staging results from the BPNN model and expert manual staging. The prediction label is the output of the BPNN model, whereas the true label is the outcome of expert manual staging.

5.5.3 DT Model: Results and Evaluation

Using the DT model, all dimensional criteria have been selected for the automated staging of sleep EEG. The results indicate that the W period has the highest precision rate (88%), followed by the N3 period (86%), and the N1 period (61%), which has the lowest recognition rate, with an overall accuracy rate of 76.25%. The associated confusion matrix is shown in Figure 5.7. This model did a better job of distinguishing between light and deep sleep; the N2 phase was mostly responsible for the incorrect prediction of the N3 period; While the inaccurate predicted of the W period was mostly concentrated in the N1 period, the false prediction of the N2 period was dispersed among the other four periods.

Table 5.3: Comparison of expert manual staging results with DT model staging outcomes for all features[24]

Sleep stage	Training Samples	Test Samples	Correct Samples	Precision	Recall	f1-Score	Accuracy
N3(Deepest stage)	1320	311	297	0.877	0.861	0.879	89.5%
N2(Deeper stage)	1631	447	303	0.73	0.708	0.719	72.09%
N1(Lighter stage)	1655	382	238	0.625	0.611	0.618	68.4%
REM(Rapid Eye Movement)	1523	389	281	0.68	0.719	0.699	73.8%
W(Wake stage)	1661	397	352	0.88	0.912	0.896	81.01%

Overall Accuracy: 76.25%

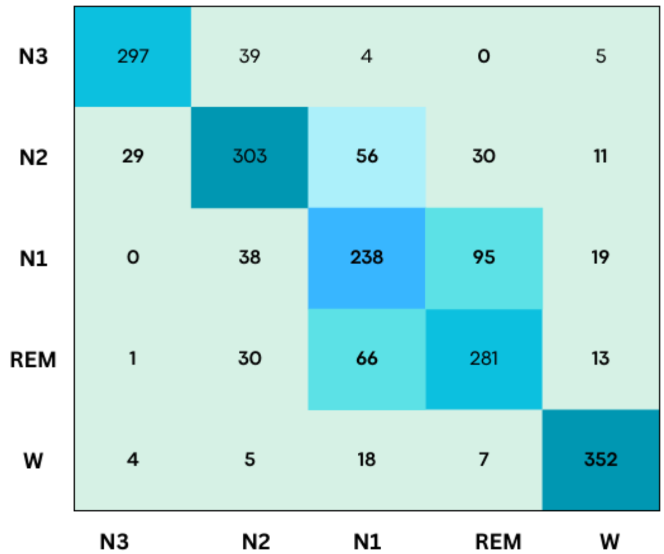


Figure 5.7: Confusion matrix of DT model staging results.

Figure 5.8 illustrates how the DT model staging results have been combined with the expert manual staging findings. The findings shown in the image indicate that the sleep stages that experts have labeled matched very well with the results of the DT model.

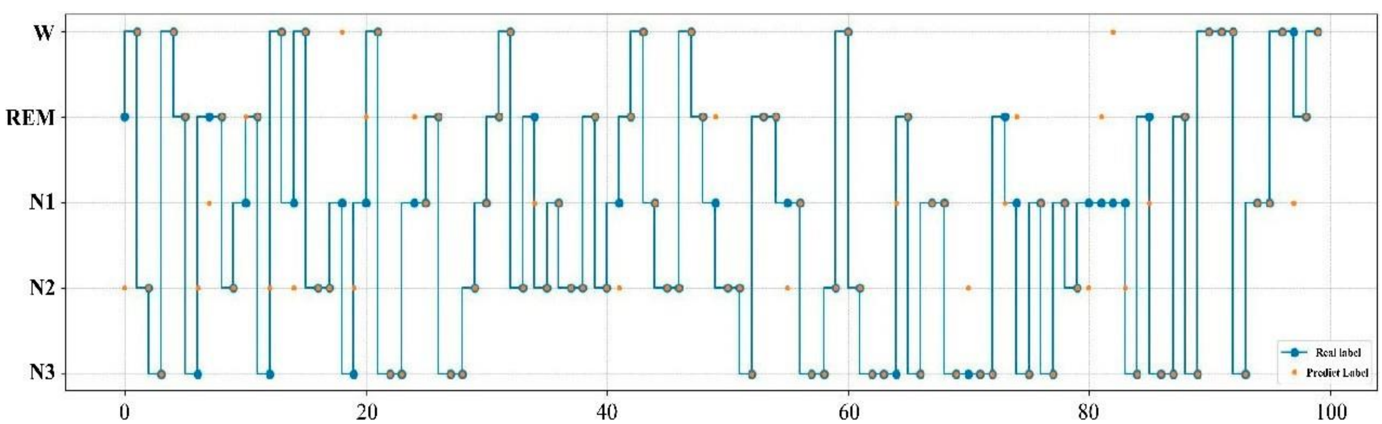


Figure 5.8: Demonstrate the comparison of expert manual staging results with DT model staging results. The real label represents the result of manual staging by experts, while the prediction label is the result of the DT model.

5.5.4 Proposed CNN model: Results and Evaluation

CNN model is used for automatic staging of sleep EEG data. Every dimensional feature is chosen. Following the alterations to the model's parameters, Table 5.4 displays a overall balancing accuracy of 80%. Figure 5.9 displays the related confusion matrix. Figure 5.9 illustrates how muddled the REM and N1 stages were most likely. The N3 stage's incorrect predictions are mostly concentrated in the N2 stage; the N2 stage's incorrect predictions are dispersed throughout the N3, N1, and REM phases; and the W stage's incorrect predictions are primarily concentrated in the N1 and N2 stages.

Table 5.4: Comparison of expert manual staging results with Proposed CNN model staging outcomes for all features

Sleep stage	Training Samples	Test Samples	Correct Samples	Precision	Recall	f1-Score	Accuracy
N3(Deepest stage)	5673	1483	1392	0.678	0.939	0.788	79.7%
N2(Deeper stage)	22163	4312	3225	0.917	0.748	0.824	83.02%
N1(Lighter stage)	8719	764	377	0.360	0.493	0.416	56.03%
REM(Rapid Eye Movement)	7125	1594	1332	0.673	0.836	0.745	81.5%
W(Wake stage)	58256	10995	10491	0.994	0.954	0.974	98.09%

Overall Balance Accuracy = 80%
Cohen's kappa = 0.805

Table 5.5: Specificity for all sleep stage for proposed CNN model

Sleep stages	Specificity
N3	0.9626
N2	0.9803
N1	0.9636
REM	0.9631
W	0.9925

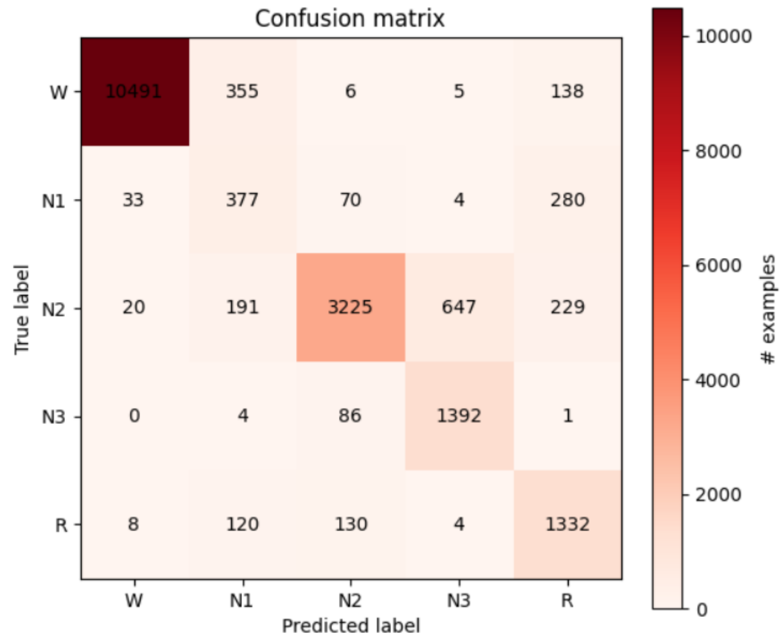


Figure 5.9: Confusion matrix of proposed CNN model staging results

Figure 5.10 illustrates how the proposed CNN model staging results have been combined with the expert manual staging findings. The findings shown in the image indicate that the sleep stages that experts have labeled matched very well with the results of the proposed CNN model.

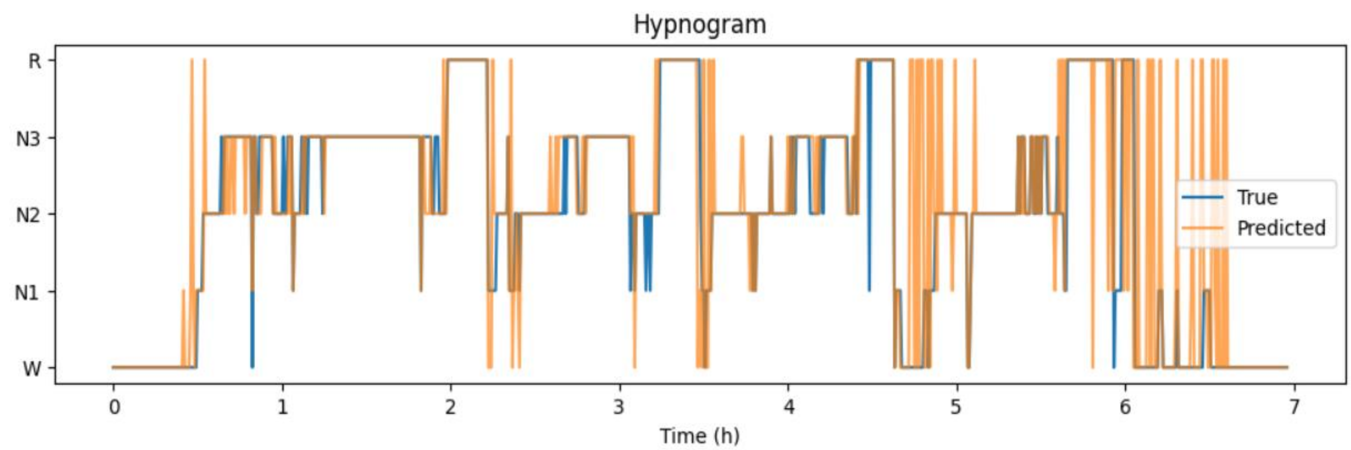


Figure 5.10: Demonstrate the comparison of expert manual staging results with proposed CNN model staging results. The real label represents the result of manual staging by experts, while the prediction label is the result of the proposed CNN model.

5.6 Comparison of proposed CNN model performance with other model

1. **Precision:** When assessing a classification model's performance, accuracy is an essential parameter to consider, especially when eliminating false positive predictions is the main goal. It aids in comprehending the model's positive forecasts' dependability, which is crucial in a variety of applications.

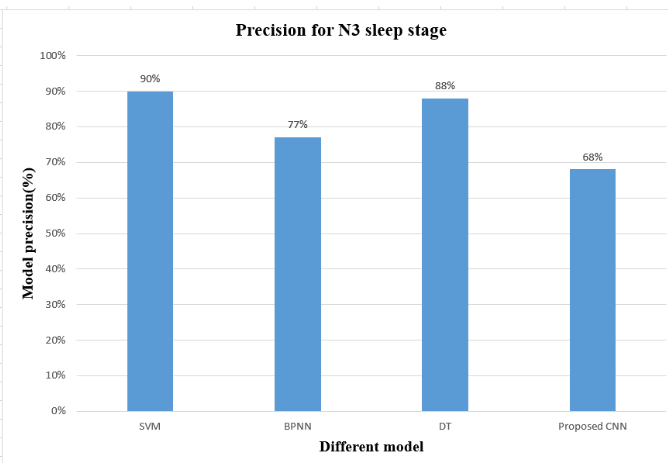


Figure 5.11: Comparison of precision between different model for N3 sleep stage

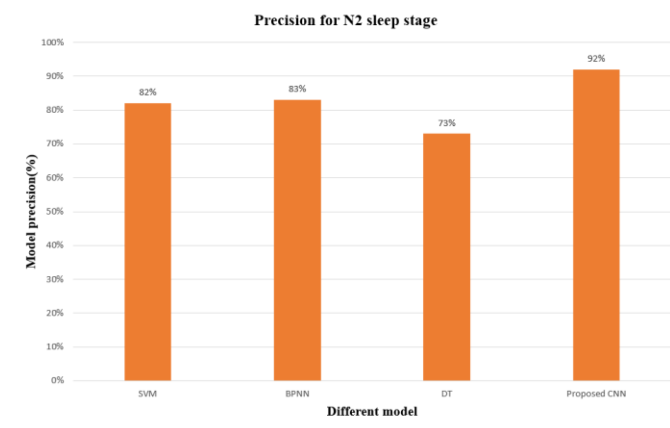


Figure 5.12: Comparison of precision between different model for N2 sleep stage

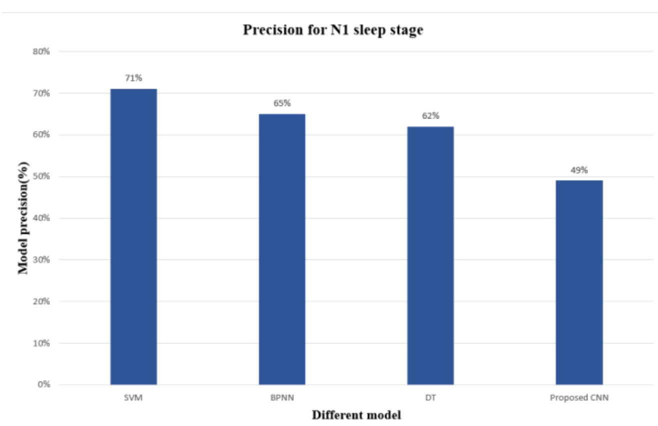


Figure 5.13: Comparison of precision between different model for N1 sleep stage

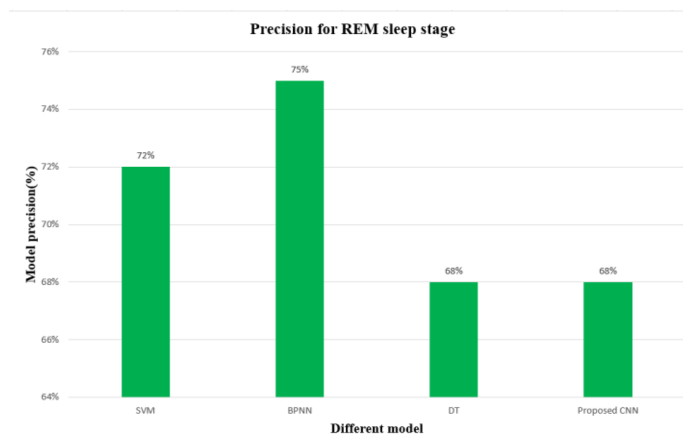


Figure 5.14: Comparison of precision between different model for REM sleep stage

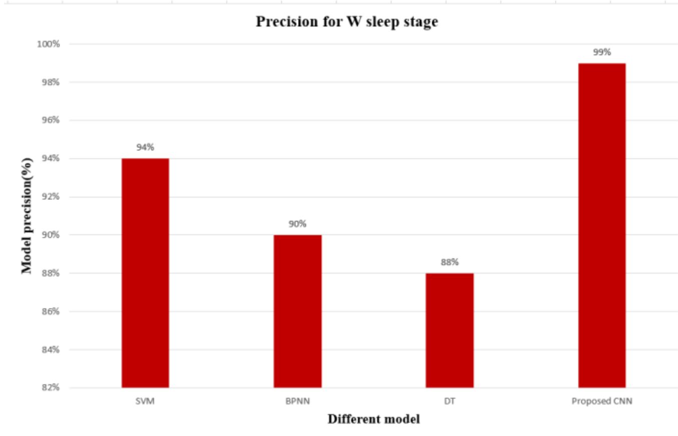


Figure 5.15: Comparison of precision between different model for N3 sleep stage

Table 5.6: Comparison table of different model for precision performance

	N3	N2	N1	REM	W
SVM model	0.8992	0.8276	0.7143	0.7215	0.9424
BPNN model	0.7685	0.8397	0.6555	0.7521	0.9007
DT model	0.8773	0.7301	0.6247	0.6804	0.88
Proposed CNN model	0.678	**0.917**	0.36	0.673	**0.994**

From table 5.6, it is seen that the proposed model has improved N2 and W sleep stage precision performance which are 0.917 and 0.994. In the table these value are identified with bold character.

2. **Recall:** Recall is a crucial parameter that is employed to assess the effectiveness of classification models. It is also referred to as sensitivity or true positive rate. It assesses a model's capacity to accurately distinguish every positive case from the dataset's total number of real positive instances.

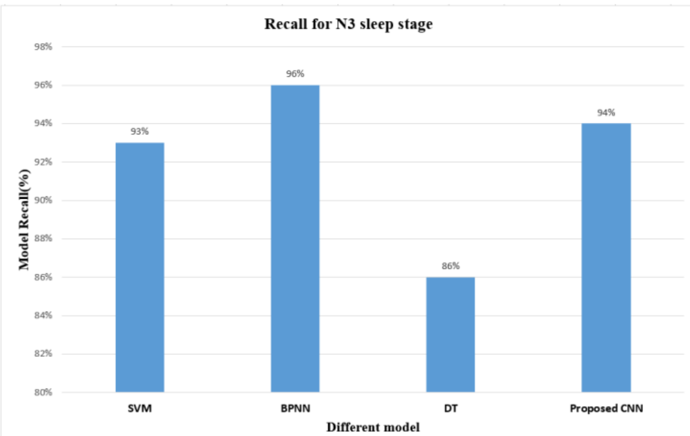


Figure 5.16 : Comparison of recall between different model for N3 sleep stage

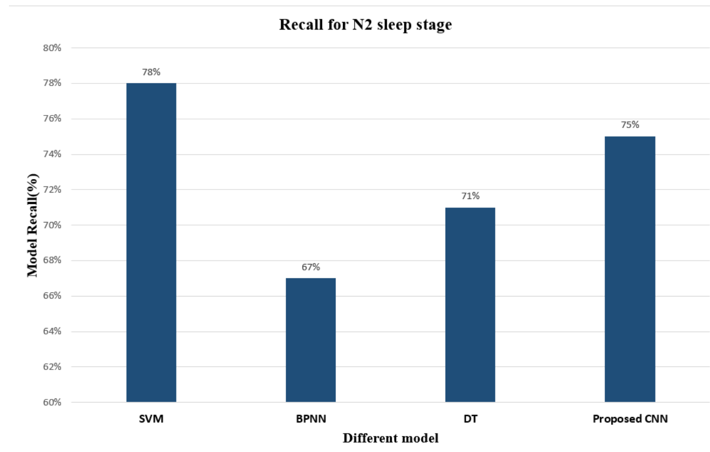


Figure 5.17: Comparison of recall between different model for N2 sleep stage

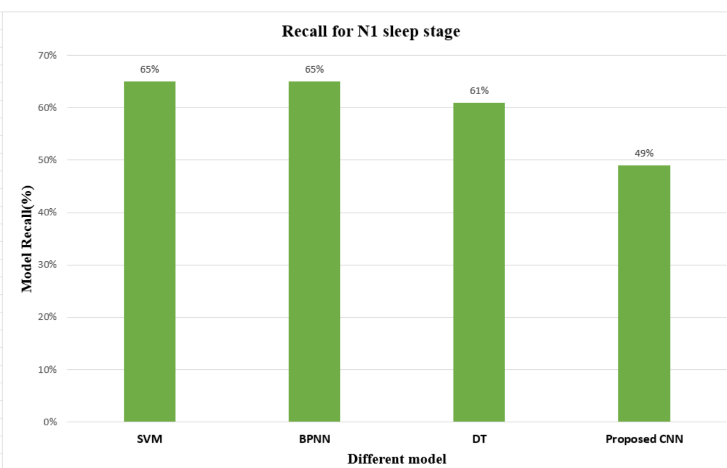


Figure 5.18: Comparison of recall between different model for N1 sleep stage

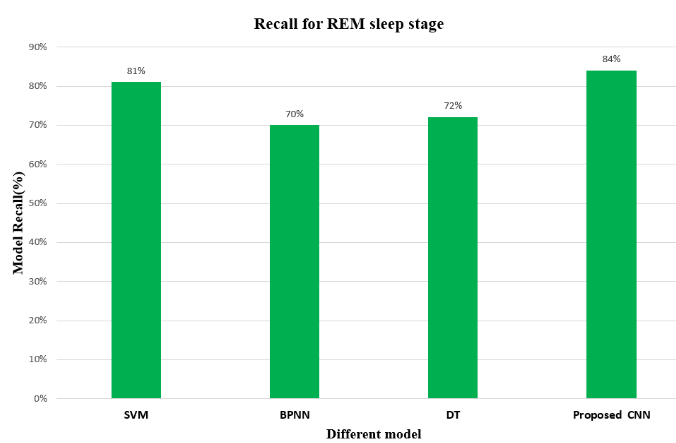


Figure 5.19: Comparison of recall between different model for REM sleep stage

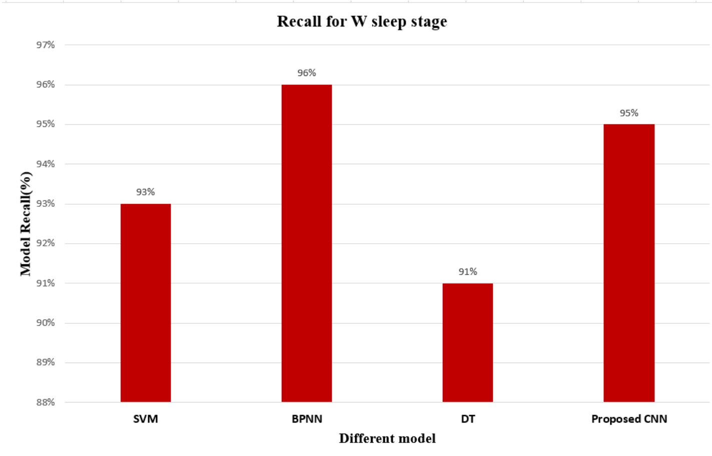


Figure 5.20: Comparison of recall between different model for W sleep stage

Table 5.7: Comparison table of different model for recall performance

	N3	N2	N1	REM	W
SVM model	0.9304	0.7850	0.6538	0.8082	0.9326
BPNN model	0.9623	0.6729	0.6538	0.6982	0.9637
DT model	0.8609	0.7079	0.6103	0.7187	0.9119
Proposed CNN model	0.939	0.748	0.493	**0.836**	0.954

From table 5.7, it is seen that the proposed model has improved REM sleep stage recall performance which is 0.836. In the table this value is identified with bold character.

3. **F1-score:** When there is an imbalance between the classes or when both recall and accuracy are significant, the F1-score is a statistic used to assess the effectiveness of classification algorithms. Recall and accuracy are harmonic means, and it yields a single score that strikes a balance between the two.

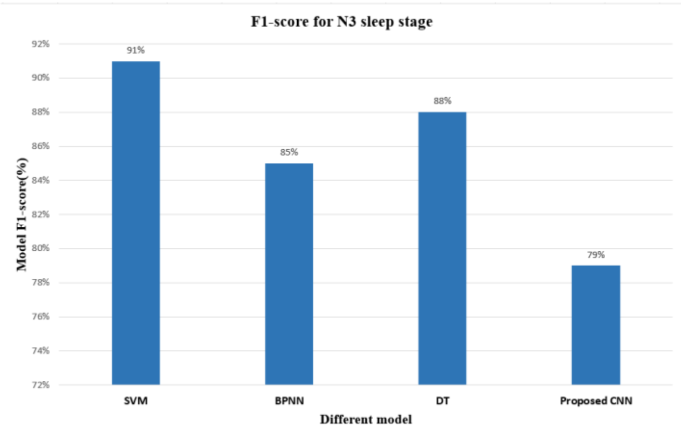


Figure 5.21: Comparison of F1-score between different model for N3 sleep stage

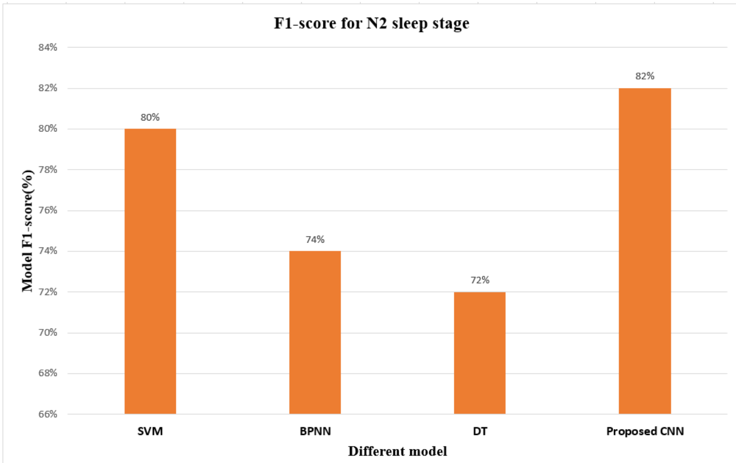


Figure 5.22: Comparison of F1-score between different model for N2 sleep stage

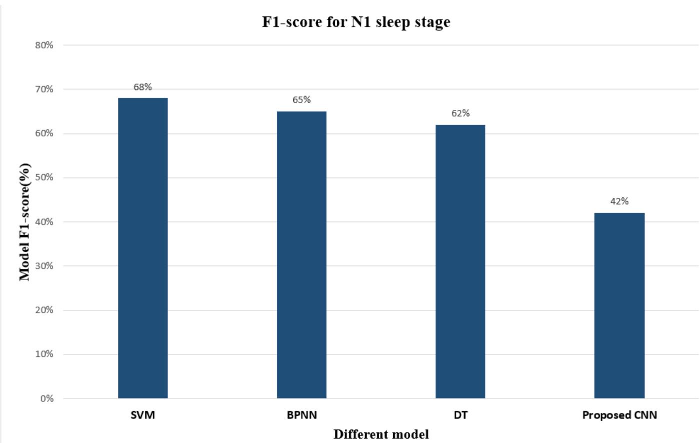


Figure 5.23: Comparison of F1-score between different model for N1 sleep stage

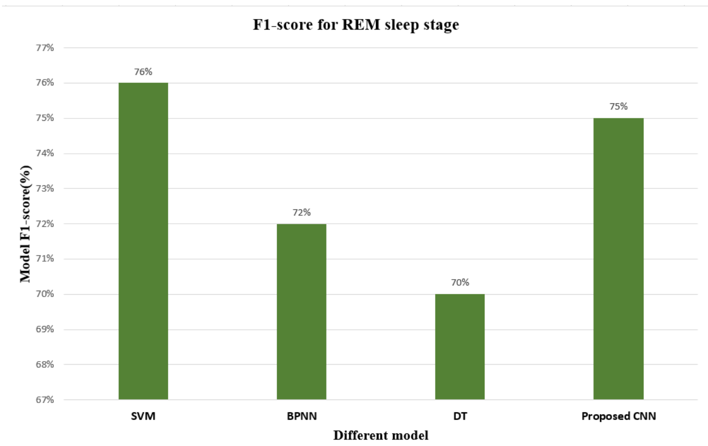


Figure 5.24: Comparison of F1-score between different model for REM sleep stage

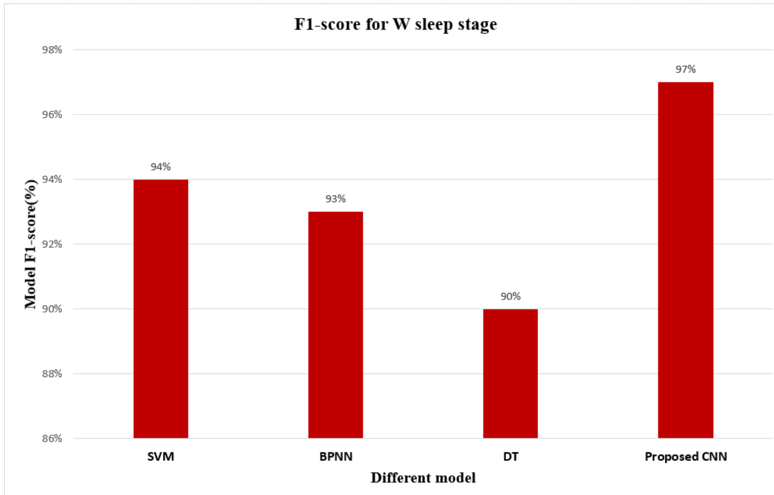


Figure 5.25: Comparison of F1-score between different model for W sleep stage

Table 5.8: Comparison table of different model for F1-score performance

	N3	N2	N1	REM	W
SVM model	0.9145	0.8058	0.6827	0.7624	0.9375
BPNN model	0.8546	0.7471	0.6547	0.7241	0.9312
DT model	0.8787	0.7189	0.6174	0.699	0.8957
Proposed CNN model	0.788	**0.824**	0.416	0.745	**0.974**

From table 5.8, it is seen that the proposed model has improved N2 and W sleep stage F1-score performance which are 0.824 and 0.974. In the table these value are identified with bold character.

5.7 Compare the proposed CNN model performance with other literature method

Method	Per-class F1-score W	N1	N2	N3	REM	Overall metrics Accuracy	Cohen's kappa
DeepSleepNet [DeepSleepNet: A model for automatic sleep stage scoring based on raw single-channel EEG]	90.9	45.0	79.2	72.7	71.1	77.8	0.70
SleepEEGNet [SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach]	89.8	42.1	75.2	70.4	70.6	74.2	0.66
ResnetLSTM [Deep convolutional network method for automatic sleep stage classification based on neurophysio logical signals]	90.7	34.7	83.6	80.9	67.0	78.9	0.71
MultitaskCNN [Joint classification and prediction CNN framework for automatic sleep stage classification]	90.9	39.7	83.2	76.6	73.5	79.6	0.72
My proposed CNN	**97.4**	41.6	82.3	78.8	**96.3**	**80**	**0.81**

Table 5.9: Demonstrate the comparision between different Literature method with the proposed CNN model

From the table, it is seen that my proposed model has better F1-score in W, REM sleep stage and has better accuracy than all the method in the table. My proposed model also has better cohen's kappa than all the method in the table.

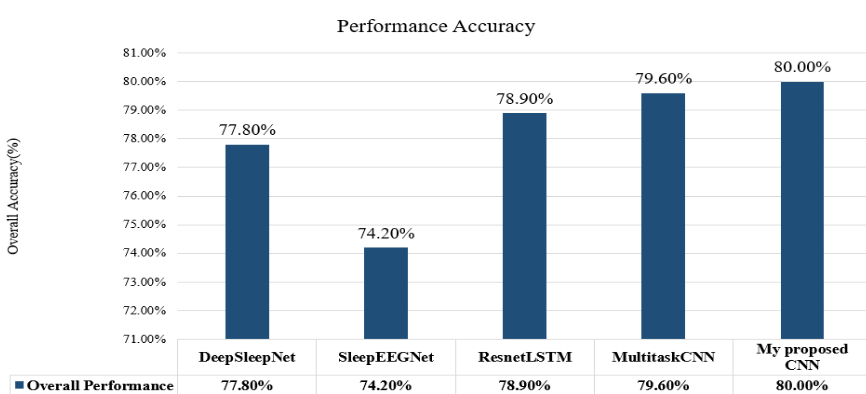


Figure 5.26: Demonstrate the comparision of Performance Accuracy of different Literature method with the proposed CNN method

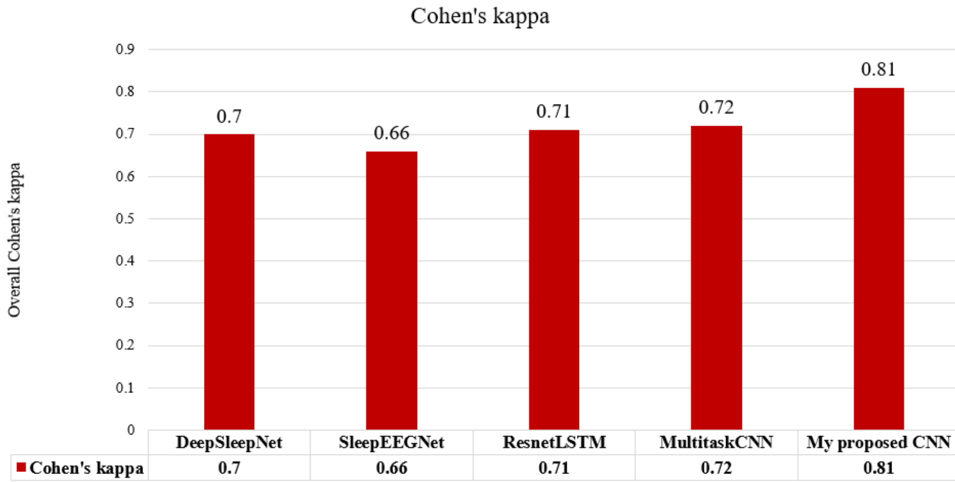


Figure 5.27: Demonstrate the comparision of Overall Cohen’s Kappa of different Literature method with the proposed CNN method.

5.8 Summary

This Chapter describes the result of each step of this proposed method. The accuracy of the proposed CNN model is 80% and overall cohen’s kappa is 0.81. The precision, recall, F1-score of W stage of proposed model is better than all the model that is described above. The precision of N2, the recall of N3 and REM and the F1-score of N2 of proposed model show the better performance than SVM, BPNN and DT model.

Chapter 6
Socio-Economic Impact and Sustainability

6.1 Impact of the project on societal, health and cultural issues

1. Social Impact:

- **Education and Awareness:** People may develop healthier sleeping habits and become more aware of the quality of their sleep if they are given insights into sleep phases and patterns.

- **Accessibility:** The general public may find it easier to monitor and analyze sleep patterns with single-channel EEG-based categorization systems, especially when those systems are based on CNN models.

2. Health Impact:

- **Early Sleep Disorder Detection:** Sleep apnea, insomnia, and narcolepsy can all be detected earlier when sleep stages are accurately classified. Early detection and action can enhance the way these illnesses are managed and enhance general health results.

3. Culture Impact:

- **Work-Life Balance:** It might be common in societies where lengthy workdays are the norm to neglect sleep in order to be more productive. In order to effectively communicate with local culture norms and overcome cultural opposition, efforts to promote good sleep habits through categorization programs may need to modify their messaging.

6.2 Impact of project on the environment and sustainability

Compared to its effects on social or health aspects, the environmental and sustainable effects of sleep stage classifying employing a single-channel EEG (Electroencephalogram) with a CNN model might not be immediately obvious. However, these initiatives may have an indirect impact on environmental, sustainability in the following ways:

- **Impact on Environment:** Accurate sleep stage classification can reduce the need for labor-intensive and resource-heavy in-lab polysomnography, potentially saving energy and resources, designing devices with recyclability and sustainability can mitigate electronic waste.

- **Sustainability:** If the AI-driven sleep stage classification becomes a standard practice, long-term sustainability involves keeping the technology up to date, addressing biases that might arise over time, and continuously improving the accuracy and relevance of the model.

Chapter 7
Addressing Complex Engineering Problems and Activities

7.1 Complex engineering problems

Table 7.1: Mapping of P's:

Attributes		Addressing Complex Engineering Problems in the project/thesis
P1	Depth of Knowledge	Require knowledge of CNN (K1),Require knowledge about filtering, data processing (k2,k3), designing the neural network (k5), Literature review is required to study existing methods (k8).
P2	Range of Conflicting Requirements	Model complexity vs. computational efficiency, Feature Extraction vs. End-to-End learning, Data quantity vs. Data quality
P3	Depth of Analysis	Have no obvious model of deep learning based sleep staging classification using single channel EEG with the desire accuracy, precision, re-call, f1-score of W sleep stage.
P4	Familiarity of Issues	Extracting relevant features from this data can be challenging, noise and artifact can affect the quality of the data and subsequently impact the performance of the classification algorithm.
P5	Extent of Applicable Codes	Pre-processing code, Feature extraction code, model architecture code, training code , Evaluation Code, Visualization Code, Code for Figures and Plots

7.2Addressing Complex Engineering Activities

Table 7.2: Mapping of A’s:

Attributes		Addressing Complex Engineering Activities in the project/thesis
A1	Range of resources	Need Data Resources, Computing resources, such as GPUs or TPUs, Software Resources(PyTorch , Numpy, Pandas, MNE-Python).
A3	Innovation	A degree of innovation of model Architecture Innovation, Feature Representation Innovation, Noise Handling Innovation
A4	Consequences for society and the environment	This model can Improve Healthcare Diagnostics, Enhance Sleep Monitoring of patient and reduce wearable device manufacturing
A5	Familiarity	A comprehensive understanding requires a thorough grasp of various components, including Confusion metrics, Leaky ReLU activation function, CNN structure, optimization algorithms like Adam,Sgd etc.

Chapter 8
Conclusions, Limitations and Future Works

8.1 Conclusion

The goal of this research is to create a single EEG channel-based automatic sleep stage classification system. Initially, the dataset was obtained via the Physionet website. after which the awake times are subtracted from the raw data's beginning and end. Then, recordings must be subjected to a lowpass filter with a 30 Hz cutoff frequency in order to lessen the effects of higher frequency noise. Next, normalize the data so that the standard deviation is one and the mean is zero. After that, a 30-second window was extracted from the continuous EEG recordings, and each dataset was combined into a single dataset. After dividing the dataset into training, testing, and validation subsets, the weights for each class were calculated. Lastly, the suggested CNN is developed, applying the training, testing, and validation datasets to the suggested model for sleep stage classification. Next, the model's performance is monitored, and the suggested CNN model's performance is contrasted with that of the pretent SVM, BPNN, and DT models. The suggested CNN model has an 80% accuracy rate and a 0.81 overall Cohen's kappa. The suggested model outperforms all of the previously mentioned models in terms of precision, recall, and F1-score in the W stage. The suggested model outperforms the SVM, BPNN, and DT models as evidenced by the precision of N2, recall of N3 and REM, and F1-score of N2.

8.2 Limitation

When compared to multi-channel EEG setups, single-channel EEG collects electrical activity from only one area of the brain, offering less spatial information. This lack of spatial resolution could make it more difficult for CNN models to identify intricate spatial patterns connected to various phases of sleep. Some sleep stages, such N1 and N2, might have identical EEG patterns, making it difficult to tell them apart using a single-channel EEG. CNN algorithms could find it difficult to discern between these minute variations, which could result in incorrect classifications. EEG signals are prone to a range of noise sources and artifacts, such as ambient interference, electrode movement, and muscle aberrations. These aberrations can negatively impact CNN model performance and deteriorate data quality in single-channel EEG settings.

If these limitations can be removed, then the accuracy of the proposed model will be improved. The performance of precision, recall, F1-score of each stage of proposed model will also improve.

8.3 Future works

All noise and artifacts which are associated with the EEG signal should be removed completely from the EEG signal when the signal is processing. The EEG signal produce imbalance class. These imbalance classes should be balanced very accurately. CNN model should focus on advancing methodological techniques and the structure of CNN model should be improved more. Also, future work will focus on multiple PSG signals, such as EOG and EMG, to improve the performance of the approach.

References

[1] G. Vanini and P. Torterolo, “Sleep-Wake Neurobiology,” 2021, pp. 65–82. doi: 10.1007/978-3-030-61663-2_5.

[2] L. Matricciani *et al.*, “Rethinking the sleep-health link,” *Sleep Health*, vol. 4, no. 4, pp. 339–348, Aug. 2018, doi: 10.1016/j.sleh.2018.05.004.

[3] H. Emsellem, “The reimagining of sleep and health,” *Sleep Health*, vol. 5, no. 1, p. 2, Feb. 2019, doi: 10.1016/j.sleh.2018.12.001.

[4] S. Ackermann and B. Rasch, “Differential Effects of Non-REM and REM Sleep on Memory Consolidation?,” *Curr Neurol Neurosci Rep*, vol. 14, no. 2, p. 430, Feb. 2014, doi: 10.1007/s11910-013-0430-8.

[5] J. P. Munoz and L. A. Rivera, “Towards Improving Sleep Quality Using Automatic Sleep Stage Classification and Binaural Beats,” in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, IEEE, Jul. 2020, pp. 4982–4985. doi: 10.1109/EMBC44109.2020.9176385.

[6] E. Aserinsky and N. Kleitman, “Regularly Occurring Periods of Eye Motility, and Concomitant Phenomena, During Sleep,” *Science (1979)*, vol. 118, no. 3062, pp. 273–274, Sep. 1953, doi: 10.1126/science.118.3062.273.

[7] T. Hori *et al.*, “Proposed supplements and amendments to ‘ *A Manual of Standardized Terminology, Techniques and Scoring System for Sleep Stages of Human Subjects* ’, the Rechtschaffen & Kales (1968) standard,” *Psychiatry Clin Neurosci*, vol. 55, no. 3, pp. 305–310, Jun. 2001, doi: 10.1046/j.1440-1819.2001.00810.x.

[8] A. Zarei and B. M. Asl, “Automatic Detection of Obstructive Sleep Apnea Using Wavelet Transform and Entropy-Based Features From Single-Lead ECG Signal,” *IEEE J Biomed Health Inform*, vol. 23, no. 3, pp. 1011–1021, May 2019, doi: 10.1109/JBHI.2018.2842919.

[9] A. Roebuck *et al.*, “A review of signals used in sleep analysis,” *Physiol Meas*, vol. 35, no. 1, pp. R1–R57, Jan. 2014, doi: 10.1088/0967-3334/35/1/R1.

[10] S. Mousavi, F. Afghah, and U. R. Acharya, “SleepEEGNet: Automated sleep stage scoring with sequence to sequence deep learning approach,” *PLoS One*, vol. 14, no. 5, p. e0216456, May 2019, doi: 10.1371/journal.pone.0216456.

[11] A. Supratak, H. Dong, C. Wu, and Y. Guo, “DeepSleepNet: a Model for Automatic Sleep Stage Scoring based on Raw Single-Channel EEG,” Mar. 2017, doi: 10.1109/TNSRE.2017.2721116.

[12] Y.-L. Hsu, Y.-T. Yang, J.-S. Wang, and C.-Y. Hsu, “Automatic sleep stage recurrent neural classifier using energy features of EEG signals,” *Neurocomputing*, vol. 104, pp. 105–114, Mar. 2013, doi: 10.1016/j.neucom.2012.11.003.

[13] S.-F. Liang, C.-E. Kuo, Y.-H. Hu, Y.-H. Pan, and Y.-H. Wang, “Automatic Stage Scoring of Single-Channel Sleep EEG by Using Multiscale Entropy and Autoregressive Models,” *IEEE Trans Instrum Meas*, vol. 61, no. 6, pp. 1649–1657, Jun. 2012, doi: 10.1109/TIM.2012.2187242.

[14] S. Chambon, M. N. Galtier, P. J. Arnal, G. Wainrib, and A. Gramfort, “A Deep Learning Architecture for Temporal Sleep Stage Classification Using Multivariate and Multimodal

Time Series," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 4, pp. 758–769, Apr. 2018, doi: 10.1109/TNSRE.2018.2813138.

[15] X. Li, L. Cui, S. Tao, J. Chen, X. Zhang, and G.-Q. Zhang, "HyCLASSS: A Hybrid Classifier for Automatic Sleep Stage Scoring," *IEEE J Biomed Health Inform*, vol. 22, no. 2, pp. 375–385, Mar. 2018, doi: 10.1109/JBHI.2017.2668993.

[16] H. Phan, F. Andreotti, N. Cooray, O. Y. Chen, and M. De Vos, "Automatic Sleep Stage Classification Using Single-Channel EEG: Learning Sequential Features with Attention-Based Recurrent Neural Networks," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, IEEE, Jul. 2018, pp. 1452–1455. doi: 10.1109/EMBC.2018.8512480.

[17] H. Phan, F. Andreotti, N. Cooray, O. Y. Chen, and M. De Vos, "Joint Classification and Prediction CNN Framework for Automatic Sleep Stage Classification," *IEEE Trans Biomed Eng*, vol. 66, no. 5, pp. 1285–1296, May 2019, doi: 10.1109/TBME.2018.2872652.

[18] M. Sokolovsky, F. Guerrero, S. Paisarnsrisomsuk, C. Ruiz, and S. A. Alvarez, "Deep Learning for Automated Feature Discovery and Classification of Sleep Stages," *IEEE/ACM Trans Comput Biol Bioinform*, vol. 17, no. 6, pp. 1835–1845, Nov. 2020, doi: 10.1109/TCBB.2019.2912955.

[19] Y. Yuan *et al.*, "A hybrid self-attention deep learning framework for multivariate sleep stage classification," *BMC Bioinformatics*, vol. 20, no. S16, p. 586, Dec. 2019, doi: 10.1186/s12859-019-3075-z.

[20] D. JIANG, Y. LU, Y. MA, and Y. WANG, "Robust sleep stage classification with single-channel EEG signals using multimodal decomposition and HMM-based refinement," *Expert Syst Appl*, vol. 121, pp. 188–203, May 2019, doi: 10.1016/j.eswa.2018.12.023.

[21] G. Zhu, Y. Li, and P. Wen, "Analysis and Classification of Sleep Stages Based on Difference Visibility Graphs From a Single-Channel EEG Signal," *IEEE J Biomed Health Inform*, vol. 18, no. 6, pp. 1813–1821, Nov. 2014, doi: 10.1109/JBHI.2014.2303991.

[22] N. Michielli, U. R. Acharya, and F. Molinari, "Cascaded LSTM recurrent neural network for automated sleep stage classification using single-channel EEG signals," *Comput Biol Med*, vol. 106, pp. 71–81, Mar. 2019, doi: 10.1016/j.compbiomed.2019.01.013.

[23] J. Zhang, R. Yao, W. Ge, and J. Gao, "Orthogonal convolutional neural networks for automatic sleep stage classification based on single-channel EEG," *Comput Methods Programs Biomed*, vol. 183, p. 105089, Jan. 2020, doi: 10.1016/j.cmpb.2019.105089.

[24] S. Zhao, F. Long, X. Wei, X. Ni, H. Wang, and B. Wei, "Evaluation of a Single-Channel EEG-Based Sleep Staging Algorithm," *Int J Environ Res Public Health*, vol. 19, no. 5, p. 2845, Mar. 2022, doi: 10.3390/ijerph19052845.

[25] F. Li *et al.*, "End-to-end sleep staging using convolutional neural network in raw single-channel EEG," *Biomed Signal Process Control*, vol. 63, p. 102203, Jan. 2021, doi: 10.1016/j.bspc.2020.102203.

[26] S. K. Satapathy, H. K. Kondaveeti, S. R. Sreeja, H. Madhani, N. Rajput, and D. Swain, "A Deep Learning Approach to Automated Sleep Stages Classification Using Multi-Modal Signals," *Procedia Comput Sci*, vol. 218, pp. 867–876, 2023, doi: 10.1016/j.procs.2023.01.067.

[27] A. Supratak, H. Dong, C. Wu, and Y. Guo, "DeepSleepNet: A Model for Automatic Sleep Stage Scoring Based on Raw Single-Channel EEG," *IEEE Transactions on Neural*

Systems and Rehabilitation Engineering, vol. 25, no. 11, pp. 1998–2008, Nov. 2017, doi: 10.1109/TNSRE.2017.2721116.

[28] X. Shao and C. Soo Kim, "A Hybrid Deep Learning Scheme for Multi-Channel Sleep Stage Classification," *Computers, Materials & Continua*, vol. 71, no. 1, pp. 889–905, 2022, doi: 10.32604/cmc.2022.021830.

[29] E. Eldele *et al.*, "An Attention-Based Deep Learning Approach for Sleep Stage Classification With Single-Channel EEG," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 29, pp. 809–818, 2021, doi: 10.1109/TNSRE.2021.3076234.

[30] S. Santaji and V. Desai, "Analysis of EEG Signal to Classify Sleep Stages Using Machine Learning," *Sleep Vigil*, vol. 4, no. 2, pp. 145–152, Dec. 2020, doi: 10.1007/s41782-020-00101-9.

[31] O. Yildirim, U. Baloglu, and U. Acharya, "A Deep Learning Model for Automated Sleep Stages Classification Using PSG Signals," *Int J Environ Res Public Health*, vol. 16, no. 4, p. 599, Feb. 2019, doi: 10.3390/ijerph16040599.

[32] M. S. Mourtazaev, B. Kemp, A. H. Zwinderman, and H. A. C. Kamphuisen, "Age and Gender Affect Different Characteristics of Slow Waves in the Sleep EEG," *Sleep*, vol. 18, no. 7, pp. 557–564, Sep. 1995, doi: 10.1093/sleep/18.7.557.

[33] "Applications of Support Vector Machine (SVM) Learning in Cancer Genomics," *Cancer Genomics Proteomics*, vol. 15, no. 1, Jan. 2018, doi: 10.21873/cgp.20063.

[34] S. Tian, N. Dai, L. Li, W. Li, Y. Sun, and X. Cheng, "Three-dimensional mandibular motion trajectory-tracking system based on BP neural network," *Mathematical Biosciences and Engineering*, vol. 17, no. 5, pp. 5709–5726, 2020, doi: 10.3934/mbe.2020307.

[35] D. Che, Q. Liu, K. Rasheed, and X. Tao, "Decision Tree and Ensemble Learning Algorithms with Their Applications in Bioinformatics," 2011, pp. 191–199. doi: 10.1007/978-1-4419-7046-6_19.

[36] W. Mumtaz, S. S. A. Ali, M. A. M. Yasin, and A. S. Malik, "A machine learning framework involving EEG-based functional connectivity to diagnose major depressive disorder (MDD)," *Med Biol Eng Comput*, vol. 56, no. 2, pp. 233–246, Feb. 2018, doi: 10.1007/s11517-017-1685-z.