UNIVERSITY OF TWENTE.

Faculty of Electrical Engineering,
Mathematics & Computer Science

Deep Sleep Stage Detection

Changqing Lu

Master in Computer Science
Specialization: Data Science and Technology

Master Thesis
26th July, 2020

Supervisors:
Dr. Christin Seifert
Email: c.seifert@utwente.nl
Dr. Ing. Gwenn Englebienne
Email: g.englebienne@utwente.nl
PhD Candidate Shreyasi Pathak
Email: s.pathak@utwente.nl

Data Management and Biometrics Group
Faculty of Electrical Engineering,
Mathematics and Computer Science
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

Declaration of Authorship

I, Changqing Lu, declare that this thesis titled, ”Deep Sleep Stage Detection” and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed: 陆长青

Date: 26-07-2020

Acknowledgements

In the past six months, I have been working on my master thesis, which is really an unforgettable and precious experience for me. During this period, with the help of my supervisors, I opened the door of my academic career and learnt many useful skills on conducting an academic research. Here, looking back on this experience, I have many thanks to express to those who were always there supporting and helping me, because without them, I would not have completed my thesis and grew from a layman to who I am now.

Firstly, I would like to thank my master assignment supervisors: Christin Seifert, Gwenn Englebienne and Shreyasi Pathak for always being patient, supportive and helpful throughout my master assignment. Your kind encouragements, helpful discussions and critical feedbacks helped and taught me a lot. Christin, thank you for providing me so many opportunities to improve myself with necessary research skills, for motivating me with brilliant ideas and useful methods when I was at a loss and for always being patient to give me detailed suggestions and guidance. Gwenn, thank you for your valuable suggestions on the implementation of my master assignment and for your critical feedbacks about the specific model description in my thesis. Shreyasi, thank you for always being very patient and helpful to help me solve the technical problems I met and for giving kind encouragements when I was down. It was my pleasure to work under your supervision and I really enjoyed this learning experience.

Secondly, I would also like to express my gratitude to Jeroen Geerdink, Loes Reichman and Mirjam Stappenbelt-Groot Kormelink from Ziekenhuis Groep Twente (ZGT). I am very grateful for your kind collaboration on clinical data collection, though the planned work was cancelled due to the lock-down situation.

Thirdly, I would like to thank my family, my girlfriend and my friends at the university for your constant understanding and support during my master study.

Finally, I would like to thank you, the reader, for your attention to read my thesis.

Abstract

Sleep quality is very important to human health. To detect sleep disorders, sleep scoring is performed by sleep experts on the polysomnograms that record the activities of different parts of the human body, like electroencephalogram (EEG), electrooculogram (EOG) and electromyogram (EMG). Current automatic sleep scoring approaches are mostly based on single-channel EEGs and the few multi-channel models that exist do not obtain a satisfying performance. In this master assignment, we firstly perform a module evaluation to test the performance of useful deep learning modules developed for optimizing single-channel models in multi-channel sleep scoring. Based on the results, we build a well-performing multi-channel automatic sleep scoring model, where temporal learning is applied to extract temporal features from sleep epochs, spatial learning is designed to capture correlation information among the channels of a modality, sequential learning is performed to extract transition rules from sleep sequences and the residual connection is used to consider temporal and sequential information together for sleep stage classification. We evaluate our model on two public datasets — the SleepEDF-13 and SHHS-1 datasets. Our model obtains an accuracy of 84.6%, macro F1 score of 78.3% and Cohen's kappa of 0.79 for the SleepEDF-13 dataset and an accuracy of 86.4%, macro F1 score of 77.7% and Cohen's kappa of 0.81 for the SHHS-1 dataset. Additionally, we employ two methods — the layer-wise relevance propagation (LRP) and an embedded channel attention network (Embedded CAN) to investigate the channel and feature importance in automatic sleep scoring. Results show that our multi-channel sleep scoring model performs well on different datasets compared to the state-of-the-art, and channel and feature importance obtained comply with the AASM rules and can be a guidance for further optimizing automatic sleep scoring models.

Contents

Chapter 1

Introduction

In this chapter, we give an overview on the research field of sleep scoring and the current scenario of automatic sleep scoring. Then, we point out the existing problems in the field and introduce possible interesting study directions accordingly. As a brief summary, we explain the associated research questions for this assignment. In the end, we introduce the organization of the thesis.

1.1 Sleep Scoring

Sleep quality is closely related to human health. Effective sleep quality detection can help sleep experts monitor and test sleep disorders and formulate corresponding treatments for the patients.

To detect the sleep quality scientifically, the polysomnography (PSG) (i.e. a sleep study) is carried out. Signals that record the activities of various parts of human body are analysed to diagnose sleep disorders. These collected signals mainly consist of electroencephalograms (EEGs), electrooculograms (EOGs), electromyograms (EMGs), electrocardiograms (ECGs) and some leg movements. In PSG, polysomnograms of usually 8 hours sleep are segmented into 30-second epochs, and the sleep epochs are then annotated into various sleep stages by technicians according to certain rules in sleep manuals. The classification procedure of sleep stages is called sleep scoring.

The unity of the rules described in sleep manuals is very significant for sleep scoring, as any slight difference might lead to different annotations. To keep the unity of the rules, standard manuals are published. The Rechtchaffen and Kales standard (the R&K manual) [1] and the American Academy of Sleep Medicine rules (the AASM manual) [2] are two most widely used manuals in sleep stage classification, where 5 (or 6) sleep stages are distinguished - Wake, Non-REM 1 (N1), Non-REM 2 (N2), Non-REM 3 (N3) and Rapid Eye Movement (REM) (i.e. the R&K manual [1] has a further classification from N3 to N3 and N4). Each stage is characterised by distinctive frequency-domain and time-domain patterns in the manuals. A summary of these scoring rules for particular sleep stages is presented in Table 1.1.

Originally, sleep scoring is manually performed by sleep experts, which is tedious and time-consuming. To improve that, automatic sleep scoring approaches are proposed. With feature analysis and extraction, sleep stages are classified automatically by applying machine learning classification algorithms to the extracted features.

Stages	EEG					EOG	EMG
	Delta (<4Hz)	Theta (4-7Hz)	Alpha (8-13Hz)	Beta (>13Hz)	Time-domain patterns		
Wake			x	x		0.5-2Hz	Variable amplitude but usually higher than during sleep stages
N1		x	x		Vertex waves	Slow eye Movement	Lower amplitude than in stage Wake
N2		x			K-complexes Sleep spindles	Usually no eye movement, but slow eye movements may persist	Lower amplitude than in stage Wake and may be as low as in stage REM
N3	x				Sleep Spindles may persist	Eye movements are not typically seen	Lower amplitude than in stage N2 and sometimes as low as in stage REM
REM		x	x		Sawtooth waves	Rapid eye movement	Lower chin EMG tone; usually the lowest level of entire recording

Table 1.1: Summary of EEG, EOG and EMG patterns for different sleep stages according to the AASM manual [2].

1.2 Current Scenario

Recently, many studies have been conducted for automatic sleep scoring with the help of the time-frequency analysis and machine learning algorithms. Generally, the automatic sleep scoring approaches can be divided into two categories according to their feature extraction methods. One is based on manual feature extraction, where the features that are used to identify the sleep stages are hand-engineered; the other is based on automatic feature extraction, where complex deep neural networks are utilized to capture underlying features from EEG, EOG and EMG signals automatically.

For manual feature extraction, time-frequency features of the signals are extracted by time-frequency analyses like Discrete Fourier Transform and Wavelet Transform [3]–[6]. These hand-engineered features are then passed to traditional machine learning models like the Support Vector Machine, Gaussian Mixture Models and the Random Forest [3], [4], [6], [7] for sleep stage classification. This kind of automatic sleep scoring can usually have a good performance on a small dataset, but it is hard to generalize to new datasets. The reason behind this is that manual feature extraction commonly requires prior knowledge and understanding of sleep scoring rules which vary among different sleep technicians. Additionally, the extracted time-frequency features in one dataset might differ from another. To solve these problems, sleep scoring approaches where features extraction is performed automatically are proposed.

It has been introduced in [8] that, complex deep neural networks can extract abstract feature representations from various data types including signals, images and time series, and end-to-end learning algorithms can combine the feature extraction and classification task together. For automatic feature extraction based sleep scoring, the deep learning architecture of Convolutional Neural Networks (CNNs) is most widely used to capture the time-invariant features of sleep epochs [9]–[14]. In addition, Recurrent Neural Networks (RNNs) are employed in some studies [11], [13], [15] to learn transition rules from the sleep sequences. These methods are mainly applied on single-channel EEG for sleep scoring. Compared to manual feature extraction based methods, the deep learning approaches like [11], [13] can obtain good performance on various datasets with the identical models, which proves their better capacity of generalization.

1.3 Existing Problems and Research Directions

Though the automatic feature extraction based approaches have shown good performance, there are still some existing problems deserving to be investigated and solved for an improvement of automatic sleep scoring.

Firstly, as far as we know, most existing works [10]–[14] were based on single-channel EEG, as EEG signals contain the most information. Some research [14] scored the sleep epochs based on single-channel EOG as well but achieved worse performance. Actually, other modalities (i.e. EOG and EMG) also contain useful information (see in Table 1.1) for sleep scoring according to the AASM manual [2] and incorporating them can help improve the performance. In an initial study [16], the optimal combination of polysomnographic channels was investigated and the best performance was obtained using 9 channels (6 EEGs, 2 EOGs and 1 EMG) for multi-class sleep staging, which shows the potential of sleep scoring based on multi-channel polysomnograms. To exploit the contributions of multiple modalities in automatic sleep scoring, several studies on multi-channel automatic sleep scoring were carried out afterwards. However, there were few well-performing multi-channel automatic sleep scoring approaches till now. Therefore, it is necessary to develop a model suitable for multi-channel sleep scoring.

Secondly, previous multi-channel work usually regarded their automatic sleep scoring as a new problem and developed novel spatial learning, temporal learning and sequential learning modules to capture time-invariant and sequential features from sleep epochs. Actually, the existing single-channel approaches have explored various effective deep learning modules with specific aims to improve sleep scoring, such as using CNNs with different filter sizes to capture time-domain patterns and frequency-domain patterns respectively [11] and applying the attention mechanism in sequential learning to learn relevant parts of sleep sequences [13], and have proved their benefits in model improvement. But until now, there was no multi-channel sleep scoring work testing their effectiveness and utilizing useful ones for multi-channel sleep scoring. Therefore, it is meaningful to evaluate the suitability of the existing 'good' modules for multi-channel sleep scoring and develop a model based on that.

Thirdly, according to sleep experts, information from different modalities and channels may have various influence in classifying different sleep stages, which can be illustrated by some studies as well. For example, the results of [11] show that using EEG Fpz-Cz channel can have an approximately 2% higher accuracy than using EEG Pz-Oz channel when classifying sleep stages with the same scoring model. According to the results of the study [14], EOG channel may have advantages in detecting stage N1 than EEG channel though only using EOG channel has a worse overall performance in sleep stage detection. Therefore, for multi-channel sleep scoring, it is interesting to investigate the channel importance to particular sleep stages, which can be utilized for further optimization of the sleep scoring model.

1.4 Research Questions

Based on the problems we discussed in Section 1.3, we propose two research questions in our research and list corresponding general solutions as follows:

1. (RQ1) *What is a well-performing model for multi-channel automatic sleep scoring?*

To build a well-performing model for multi-channel automatic sleep scoring, we first have a comprehensive review of the effective deep learning modules developed for single-channel models, test their suitability in our multi-channel model and employ the useful ones. Additionally, a spatial learning part will be designed to extract the correlation information among the channels within a modality.

2. (RQ2) *How much does the information of each channel contribute to sleep scoring?*

To infer the channel importance, two solutions are proposed. One is an intrinsic method, where we can add a channel attention identification module in training our multi-channel sleep scoring model. The channel attention weights will be calculated by a conditional neural network and reported as channel importance scores. The other is a post-hoc interpretation method inspired from the deep neural networks interpretation [17]. With a trained sleep scoring model, we can back-propagate the predictions to obtain the relevance of input channels to the predictions.

1.5 Thesis Organization

The rest of the thesis is organized as follows: Chapter 2 presents a review of the existing work on automatic sleep scoring and the current approaches that are helpful for channel importance investigation. Chapter 3 introduces the methodology proposed to solve the research questions. Chapter 4 describes the materials and experimental setup. Chapter 5 presents the experiment results and gives analyses and discussions accordingly. Chapter 6 provides a brief conclusion of our research and proposes the future work.

Chapter 2

Related Work

In this chapter, we present the existing works on automatic sleep scoring in two categories based on the number of channels they use. After that, an analysis is performed to summarize their performance, point out existing problems and start our research. In addition, the necessity and inspirations to find channel importance are introduced as well.

2.1 Automatic Sleep Scoring

As discussed in Section 1.2, there are currently two categories of automatic sleep scoring approaches. One is using hand-engineered features extracted from the time-frequency analysis for classification. The other relies on deep learning architectures to learn abstract pattern representations automatically. In our research, we focus on the latter ones, as it has been shown in [18] that automatic feature extraction based models can be better generalized to other datasets. More specifically, deep automatic sleep scoring methods can also be divided to two categories based on the number of channels they use — single channel and multiple channels. In this section, we will first review them separately and then summarize the possible improvements in building our well-performing multi-channel deep automatic sleep scoring model.

2.1.1 Single-channel Models

Most of the studies in this category were developed based on single-channel EEG. In a single-channel sleep scoring model, CNNs and RNNs are the most widely used deep learning architectures. Usually, CNNs are employed to extract time-invariant features from the current sleep epoch [11], [13], [14], and on top of that, RNNs are utilized to capture the transition rules by paying attention to neighbouring epochs as well [11], [13], [15]. A fully-connected layer is then used to classify the sleep stages based on the extracted features. There are also some studies [10], [12] extracting the time-invariant features directly from both the current sleep epoch and neighbouring epochs by CNNs to include transition information instead of employing extra sequential learning architecture for sleep scoring.

Architectures mentioned above are the basic components for almost every single-channel sleep scoring model. To improve the performance of a deep sleep scoring model, extra contributing modules were developed to extract target-specific features more comprehensively and precisely. For example, according to the AASM manual [2], EEG signals consist of two kinds of features: frequency-domain features throughout EEG signals and time-domain patterns usually appearing in an around 0.5-second period like K-complex and sleep spindles. Supratak et al. [11] employed two CNN pipelines with different filter sizes in temporal learning, where the motivation is to use smaller filters to extract the time-domain patterns and use larger filters to capture the frequency-domain information from EEG signals. Additionally, various mixtures of 0.5-second patterns may appear in identical sleep stages, which complicates the feature extraction. Since feature complexity can be increased by deeper layers in CNN [19], Yildirim et al. [14] and Sors et al. [12] employed CNN with 19 layers and 14 layers but very small filter size respectively to extract complex time-invariant patterns from sleep epochs. Considering the similarities between sleep scoring procedure and machine translation (i.e. sequence-to-sequence learning), Mousavi et al. [13] applied the attention mechanism to let sequential learning modules pay more attention to the important parts of sleep sequences. To avoid the final sleep stage classification focusing too much on the sequential information extracted by the sequential learning part, which might cause information loss of the time-invariant features, Supratak et al. [11] applied the residual connection that adds temporal information extracted by CNNs to sequential learning features from Bi-LSTM. Humanyun et al. [20] also implemented residual CNNs to resolve the vanishing gradient problem arising from the training of deeper CNN models. In addition, there was also some study [21] that represented raw EEG signals with their spectrograms and transformed sleep scoring into an image classification problem.

2.1.2 Multi-channel Models

Most existing multi-channel studies simply combined the features extracted from all EEG, EOG and EMG channels together to classify sleep stages. As an initial study, Khalighi et al. [16] found the best combination of EEG, EOG and EMG channels for multi-channel sleep scoring through testing multiple combinations of their time-domain and frequency-domain features and applying the Support Vector Machine algorithm for classification. The model based on 9 channels gave the best results for multi-class sleep staging. For deep learning models, Cen et al. [9] utilized CNNs to extract time-invariant features of sleep epochs and applied the Hidden Markov Model for classification. Paisarnsrisomsuk et al. [22] developed a 17-layer CNN to learn the features from both the current sleep epoch and neighbouring epochs and tested it on two kinds of channel combinations: 1) channels from both EEG and EOG modalities and 2) channels from EEG only, where adding EOG channels increased the accuracy by 1%. Similar to [21], Phan et al. [23] generated spectrograms for the signals of EEG, EOG and EMG and used them to train a multi-task CNN model that created joint predictions from the current sleep epoch and neighbouring epochs. Their results showed an increase on accuracy by 4% when adding the EOG channel into input modalities and another increase on accuracy by 1% when adding the EMG channel. Chambon et al. [24] proposed a spatial-temporal deep learning architecture to extract the features from the current sleep epoch and neighbouring epochs as well, where the linear spatial filters can exploit the array of sensors to increase the signal-to-noise ratio. They also performed an experiment to find out the best combination from various EEG, EOG and EMG channels and achieved the conclusion that the best results came from using 6 EEGs with 2 EOGs and 3 EMGs while the inclusion of more EEG channels can not help increase the sleep staging performance. Biswal et al. [25] designed a recurrent and convolutional neural network for sleep scoring based on the spectrogram representations of EEGs. Yildirim et al. [14] employed a 19-layer CNN and tested it on two kinds of channel combinations: 1) one EEG channel and one EOG channel and 2) one EEG channel only as well, where adding the EOG channel could increase the accuracy by 1%. Pathak et al. [26], being with the Data Management and Biometrics Group at the University of Twente, developed a spatial-temporal-sequential model to respectively extract sptial-temporal features and sequential information from the sleep epochs of multiple modalities and interpreted their model using post-hoc interpretability methods.

2.1.3 Summary

According to [8], deep learning models usually require large and standardized data for training. In order to make automatic sleep scoring approaches comparable with each other, many classic databases established for PSG were used for evaluating a sleep staging model, such as the SleepEDF-13 and SleepEDF-18 databases [27], [28], the Montreal Archive of Sleep Studies (MASS) database [29] and the Sleep Heart Health Study (SHHS) visit 1 and visit 2 databases [30]. These databases have various channels of the modalities (i.e. EEG, EOG and EMG) and different main sampling rates for signal collection, but all of them follow the annotation rules in the R&K manual [1] or the AASM manual [2] resulting in identical sleep staging. An overview of the databases is shown in Table 2.1. To have a clear comparison and analysis of the automatic sleep scoring models discussed in Section 2.1.1 and 2.1.2, we summarize them in Table 2.2 with their datasets, channels, methods, evaluation methods and accuracy performance (Acc). We group these methods by the datasets they used. With the model comparison, we reach the conclusions as follows.

Firstly, as discussed in Section 2.1.2, it has been shown by many studies that, the inclusion of multiple modalities and channels can bring a performance improvement for automatic sleep scoring. However, according to the summary table, current multi-channel models didn't achieve a very satisfying performance so far. For example, training and testing on the SleepEDF-13 dataset, multi-channel sleep scoring approaches [22], [23] even showed a lower accuracy by approximately 2% compared to some single-channel approaches. Humayun et al. [20] and Yildirim et al. [14] obtained better results on heavily imbalanced datasets (i.e. biased to stage Wake), such that their claims need to be justified. Secondly, few of the multi-channel models considered the correlation information among the channels within EEGs and EOGs. Pathak et al. [26] developed the spatial-temporal-sequential model and used spatial learning to extract correlations within EEG channels and EOG channels, which achieved the accuracy of 85% on the SHHS visit 1 dataset. Thirdly, as discussed in Section 2.1.1, many single-EEG based approaches have proposed extra contributing modules (e.g. CNN with different filter sizes) and successfully improved automatic sleep scoring, which can be found from the summary table as well. However, to our knowledge, there was no research to test the effectiveness for multi-channel sleep scoring and utilize the useful modules with their benefits. Hence, to start our study, we propose to design corresponding experiments to verify whether these extra contributing modules in single-channel sleep scoring can also be helpful for multi-channel models, such as 'using CNNs with different filter sizes', 'increasing the depth of CNNs' and 'applying the attention mechanism in sequential learning'. Based on that, we develop a well-performing deep multi-channel automatic sleep scoring model by designing and adding a suitable spatial learning module to capture correlation information among the channels of a modality.

Database	Subjects	Channels	Main sampling rate	Sleep Stages
SleepEDF-13	61 PSGs	2EEGs, 1EOG and 1EMG	100Hz	Wake, N1, N2, N3, N4, REM
SleepEDF-18	197 PSGs	2EEGs, 1EOG and 1EMG	100Hz	Wake, N1, N2, N3, N4, REM
MASS	200 PSGs	4-20EEGs, 2EOGs and 3EMGs	256Hz	Wake, N1, N2, N3, REM
SHHS visit 1	6441 PSGs	2EEGs, 2EOGs and 1EMG	125Hz	Wake, N1, N2, N3, N4, REM
SHHS visit 2	3295 PSGs	2EEGs, 2EOGs and 1EMG	125Hz	Wake, N1, N2, N3, N4, REM

Table 2.1: Overview of the sleep study databases.

2.2 Channel Importance Investigation

So far, to our knowledge, there is currently no study for channel importance inference and visualization in automatic sleep scoring, but results from previous studies indicate that the scoring performance varies when different channels are used (see Section 1.3).

In similar studies of other medical fields, Bohle et al. [31] showed the potential of layer-wise relevance propagation (LRP) in assisting clinicians to explain neural network decisions for diagnosing Alzheimer's disease. They summed up the relevance of image inputs for different brain areas based on their classification model to demonstrate the area importance of the MRI. Obviously, it is a post-hoc interpretation method that works on a trained classification model. Additionally, attention mechanisms can be utilized to detect important parts and give them more attention accordingly, which has been exploited on channel-wise information fusion. Hu et al. [32] developed the Squeeze-and-Excitation (SE) block consisting of a conditional neural network to adaptively recalibrate channel-wise feature responses by explicitly modelling inter-dependencies between channels. Wang et al. [33] proposed the Efficient Channel Attention (ECA) module, where the difference with the SE block is that it employed an extra convolutional neural network for channel attention weight calculation. Bastidas et al. [34] implemented the channel attention network as well, which can allocate large attention weights to feature maps of important channels for final image prediction. The above approaches show the possibility that an embedded channel attention module in the sleep scoring models can help investigate channel importance through intrinsic interpretation.

In our study, we propose two approaches to investigate channel importance in automatic sleep scoring. Firstly, LRP [35] will be applied as a post-hoc interpretation method, where we can obtain the importance scores of a channel by adding up its relevance to predictions. This method is also set as the baseline method, as post-hoc interpretation methods have been successfully applied in many previous similar studies [36] and LRP has been found to have excellent benchmark performance [37].

Paper	Year	Dataset	PSGs	Channels	Approach	Evaluation	Acc
Tsinalis et al. [10]	2016	SleepEDF-13	20	1EEG	CNN	20-fold	74.8
Supratak et al. [11]	2017	SleepEDF-13	39	1EEG	CNN (2 filter sizes) -BiLSTM-Residual	20-fold	82.0
Mousavi et al. [13]	2019	SleepEDF-13	39	1EEG	CNN (2 filter sizes) -BiLSTM-Attention	20-fold	84.3
Wang et al. [21]	2019	SleepEDF-13	39	1EEG	Spectrogram-CNN	90-5-5	85.0
Humayun et al. [20]	2019	SleepEDF-13	39	1EEG	Residual CNN	70-30	91.4*
Paisarnsrisomsuk et al. [22]	2018	SleepEDF-13	39	2EEGs +1EOG	CNN	4-fold	81.0
Phan et al. [23]	2019	SleepEDF-13	39	1EEG +1EOG +1EMG	multi-task CNN	20-fold	82.3
Mousavi et al. [13]	2019	SleepEDF-18	61	1EEG	CNN (2 filter sizes) -BiLSTM-Attention	20-fold	80.0
Yildirim et al. [14]	2019	SleepEDF-18	61	1EEG	CNN (19 layers)	70-15-15	90.5*
Yildirim et al. [14]	2019	SleepEDF-18	61	1EEG +1EOG	CNN (19 layers)	70-15-15	91.0*
Supratak et al. [11]	2017	MASS	62	1EEG	CNN (2 filter sizes) -BiLSTM-Residual	31-fold	86.2
Chambon et al. [24]	2018	MASS	61	6EEGs +2EOGs +3EMGs	CNN	5-fold	83.0
Phan et al. [23]	2019	MASS	200	1EEG +1EOG +1EMG	multi-task CNN	20-fold	83.6
Sors et al. [12]	2018	SHHS visit 1	5728	1EEG	CNN (14 layers)	50-20-30	87.0
Biswal et al. [25]	2018	SHHS visit 1	5804	2EEGs	CNN-BiLSTM -Residual	90-10	77.9
Pathak et al. [26]	2019	SHHS visit 1	5793	2EEGs +2EOGs +1EMG	CNN-BiLSTM	81-9-10	85.0

Table 2.2: Summary of the state-of-the-art deep sleep scoring approaches. * denotes that Wake is the majority class in such datasets (see Table 2.3), and the predicting result has to be justified as Wake is easier to predict compared to the sleep stages.

Database	Sleep Stages						Total Samples
	Wake	N1	N2	N3	N4	REM	
SleepEDF-13 (Biased to Wake)	72,391 (68.0%)	2,804 (2.6%)	17,799 (16.7%)	3,370 (3.2%)	2,333 (2.2%)	7,717 (7.3%)	106,414
SleepEDF-13	8285 (19.6%)	2,804 (6.6%)	17,799 (42.1%)	3,370 (8.0%)	2,333 (5.5%)	7,717 (18.2%)	42,308
SleepEDF-18 (Biased to Wake)	285,937 (68.8%)	21,522 (5.2%)	69,132 (16.6%)	8,793 (2.1%)	4,246 (1.6%)	25,835 (6.2%)	415,465
SleepEDF-18	65,951 (33.7%)	21,522 (11.0%)	69,132 (35.4%)	8,793 (4.5%)	4,246 (2.2%)	25,835 (13.2%)	195,479

Table 2.3: Overview of the SleepEDF datasets biased or unbiased to Wake.

Inspired from the channel attention networks [32]–[34] discussed above, we also develop a novel channel attention module embedded in our deep sleep scoring model to calculate the channel importance in an intrinsic way. Additionally, we extend channel importance investigation to feature importance analysis of each channel in EEG, EOG and EMG, which could provide further suggestions for optimizing multi-channel automatic sleep scoring models.

Chapter 3

Methodology

In this chapter, we introduce the methodology used in our study. Section 3.1 talks about the effective modules evaluation we perform to test their usefulness for multi-channel automatic sleep scoring and the final architecture of our multi-channel deep sleep scoring model. Section 3.2 describes two approaches utilized to identify channel importance and a further analysis to find the significant features of EEG, EOG and EMG channels.

3.1 Multi-channel Automatic Sleep Scoring

To build a well-performing multi-channel automatic sleep scoring model, we take a two-step experiment. In the first step, we test the effectiveness of good deep learning modules used in single-channel models when applying them to multi-channel sleep scoring. In the second step, we combine and adapt the useful modules and additionally design a novel suitable spatial learning module, producing the final architecture of the multi-channel automatic sleep scoring model in our study.

3.1.1 Effective Modules Evaluation

We summarize four potential modules from the literature review that might be helpful in building a good multi-chanel model: 1) using CNNs with different filter sizes to capture time-domain patterns and frequency-domain patterns respectively [11], 2) increasing the depth of CNNs for complex feature extraction [12], [14], 3) applying the attention mechanism in sequential learning to pay more attention to relevant parts [13] and 4) adding the residual connection in the model to consider both the temporal and sequential information for final sleep stage classification [11]. To evaluate their effectiveness, we select the spatial-temporal-sequential sleep staging model proposed by Pathak et al. [26] as the baseline model, because it is a relatively successful multi-channel sleep scoring approach to our knowledge from the literature review. Compared to most existing work that simply combined the features of all channels together for sleep scoring, their work considered spatial relevance among the channels of a modality and obtained good results when tested on the SHHS-1 dataset. Their approach consists of three modules in the following order:1) the spatial filtering part that extracts correlation information within EEG and EOG signals, 2) the temporal filtering part that captures time-invariant features of EEG, EOG and EMG signals separately and 3) the sequential learning part that extracts transition rules from sleep sequences. To show the contribution of the first three testing modules (i.e. mentioned at the start of this section from 1) - 3)) precisely to multi-channel sleep scoring, we substitute the corresponding part of the baseline model with one module at a time as the testing architecture, and test their performance on a sample dataset generated by splitting the randomly shuffled SleepEDF-13 dataset into 81$\%$ for training, 9$\%$ for validation and 10$\%$ for testing. To evaluate the effectiveness of the residual connection module (i.e. mentioned at the start of this section as 4)), the baseline model we set is a model which have included all first three modules. Because, according to the study of Pathak et al. [26], the residual connection does not always work for any model architecture, and we intend to verify its usefulness in our final model. All evaluation experiments are explained separately. In this step, we only give an overview of the evaluation of these modules, as it mainly acts as an initial experiment for building our final multi-channel sleep scoring model, and the detailed information of each module that are finally employed in our model architecture will be introduced in Section 3.1.2.

Using CNNs with different filter sizes

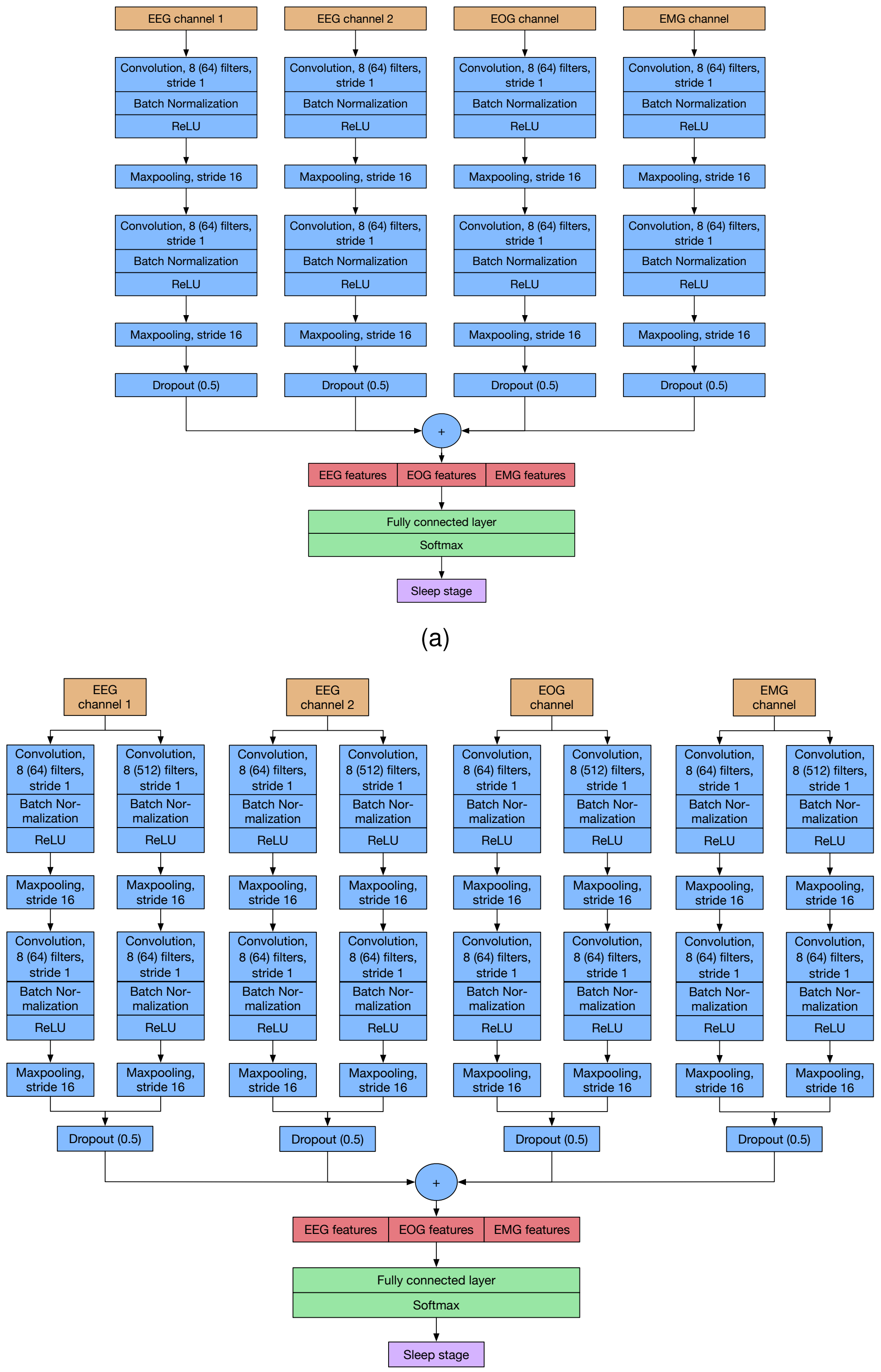
The module — using CNNs with different filter sizes, is inspired from [11]. According to the AASM manual [2], there are two types of features in polysomnograms: 1) time-domain information (e.g. distinctive 0.5-second patterns like K-complex and sleep spindles in EEG signals and amplitude information in EOG and EMG signals) and 2) frequency-domain information (e.g. dominant frequency components of the signals). In this case, using smaller filters in CNNs can capture time-domain information better and using larger filters can capture frequency-domain information better [11].

The baseline model architecture (i.e. the adapted CNN part from Pathak et al. [26]) and the testing architecture to evaluate the 'using CNNs with different filter sizes' module are plotted in Fig. 3.1. In this experiment, we first exclude the spatial filtering part of the CNNs in [26], as it is applied on the raw data inputs before temporal filtering, which might destroy the time-domain information of signals (e.g. distinctive 0.5-second patterns and amplitude information) before they are recognized. The only filter size of CNNs in the baseline model is 64. In the testing architecture, we use two different sizes which are 64 (i.e. commonly the sampling rates of signals are 100-125 Hz and $0.5 \times (100 \text{ or } 125) \approx 64$) and 512 (i.e. a large window size to help detect dominant frequency components) for the smaller and larger filters respectively. We keep the remaining hyper-parameters same as the baseline model in order to eliminate their possible effects on the results. The tests are performed on CNNs only and we do not train the sequential learning part because we just want to compare the performances in capturing time-invariant features from the current sleep epoch. The performance metrics introduced in Section 4.4 are used for the comparison.

Increasing the depth of CNNs

The module — increasing the depth of CNNs, is inspired from [14] which applied a 19-layer CNN to extract the features from sleep epochs in classifying sleep stages. In the AASM manual [2], various mixtures of the 0.5-second patterns may appear in identical sleep stages. Therefore, increasing the depth of CNNs can help the sleep scoring model learn such complex features. However, in this experiment, we do not completely follow the identical model architecture in [14] that implements a CNN with 19 layers but just add more convolutional blocks to the baseline model as our testing model architecture, as the aim of our experiment here is only to test the potential of this module type.

The baseline model architecture (i.e. the adapted CNN part from Pathak et al. [26]) and the testing architecture to evaluate the 'increasing the depth of CNNs' module are plotted in Fig. 3.2. Similar to the experiments in Section 3.1.1, we still first exclude the spatial learning part of the CNNs in [26], as in our proposal the first convolutional layer with the size of 64 is used to capture distinctive time-domain features and applying the spatial filtering directly on the raw data will destroy these features. For the testing architecture, we add three more convolution layer blocks (i.e. each block consists of a convolutional layer, a batch normalization layer [38] and a rectified linear unit (ReLU) layer) and an extra dropout layer (i.e. to avoid the overfitting coming from the increasing model complexity), resulting in the 20-layer CNN for the feature extraction in each channel compared to the baseline 10-layer CNN model. Matching with increasing complexity of the network, we also add more filters in CNNs accordingly. The remaining hyper-parameters of the testing model architecture are kept the same as the baseline model, and the same performance



(a)

(b)

Figure 3.1: Baseline model architecture (a) and testing model architecture (b) for evaluating the module — using CNNs with different filter sizes.

metrics are used to evaluate this module as well.

Applying the attention mechanism in sequential learning

The module — applying the attention mechanism in sequential learning, is inspired by Mousavi et al. [13] who improved the work of Supratak et al. [11] through adding the attention mechanism to focus on the important parts of a sleep sequence when extracting transition rules. According to [13], similar to machine translation, sleep stage scoring can be regarded as a sequence-to-sequence learning task, where not all of the proceeding and following epochs have the same influence in predicting the current sleep stage. Thus, the attention mechanism can give more attention to significant epochs with higher attention weights.

The baseline model architecture (i.e. the whole spatial-temporal-sequential model from Pathak et al. [26]) and the testing architecture to evaluate the 'applying the attention mechanism in sequential learning' module are plotted in Fig. 3.3. The CNN part in the baseline model is represented simply by brief blocks, as they are not the main comparison object in this experiment and we only perform the substitution for the sequential learning part. An attention mechanism based sequential learning architecture similar to [13] is designed as the testing architecture. However, instead of transforming the sleep scoring problem simply into a machine translation problem like [13] where the outputs of their sequential learning part are sequences of sleep stages, our testing sequential learning module output new feature representations of the sleep epochs with sequential information added. The final sleep stage classification is performed based on these new feature representations. There are two motivations behind it: 1) we expect to give a final feature representation to each sleep epoch which would be useful for studying the characteristics of a particular stage in future work and 2) there might be the loss of the time-invariant information in sequential learning as the time-invariant features of the current epoch extracted by the CNN part are not focused on in sequential learning, so that in this architecture the necessity of residual connections can be tested. The same evaluation metrics are used for this experiment as well.

Adding the residual connection to final feature representations

The module — adding the residual connection from CNN features to final feature representations, is inspired from [11]. The residual connection can help avoid the information loss caused by the sequential learning part for two reasons. As we know, data imbalance is an important problem in deep automatic sleep scoring because minority classes are usually difficult to detect by deep neural networks. To deal

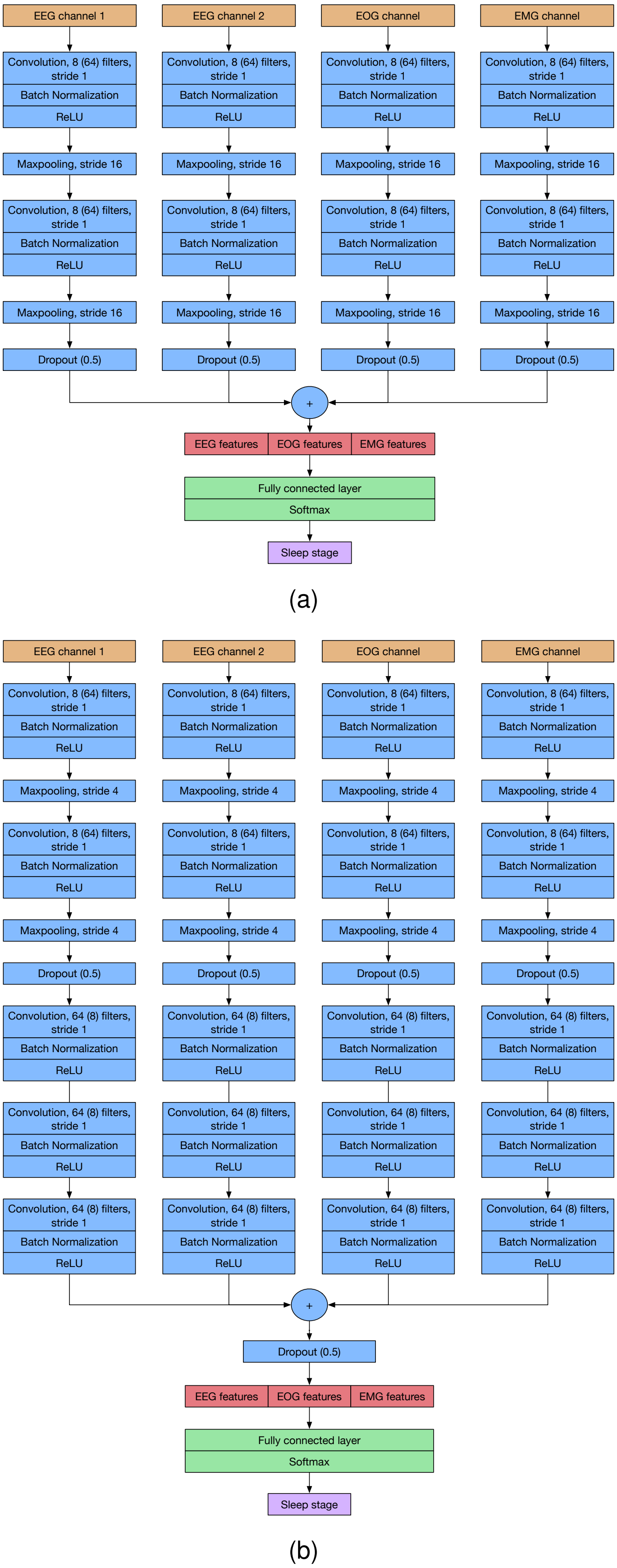
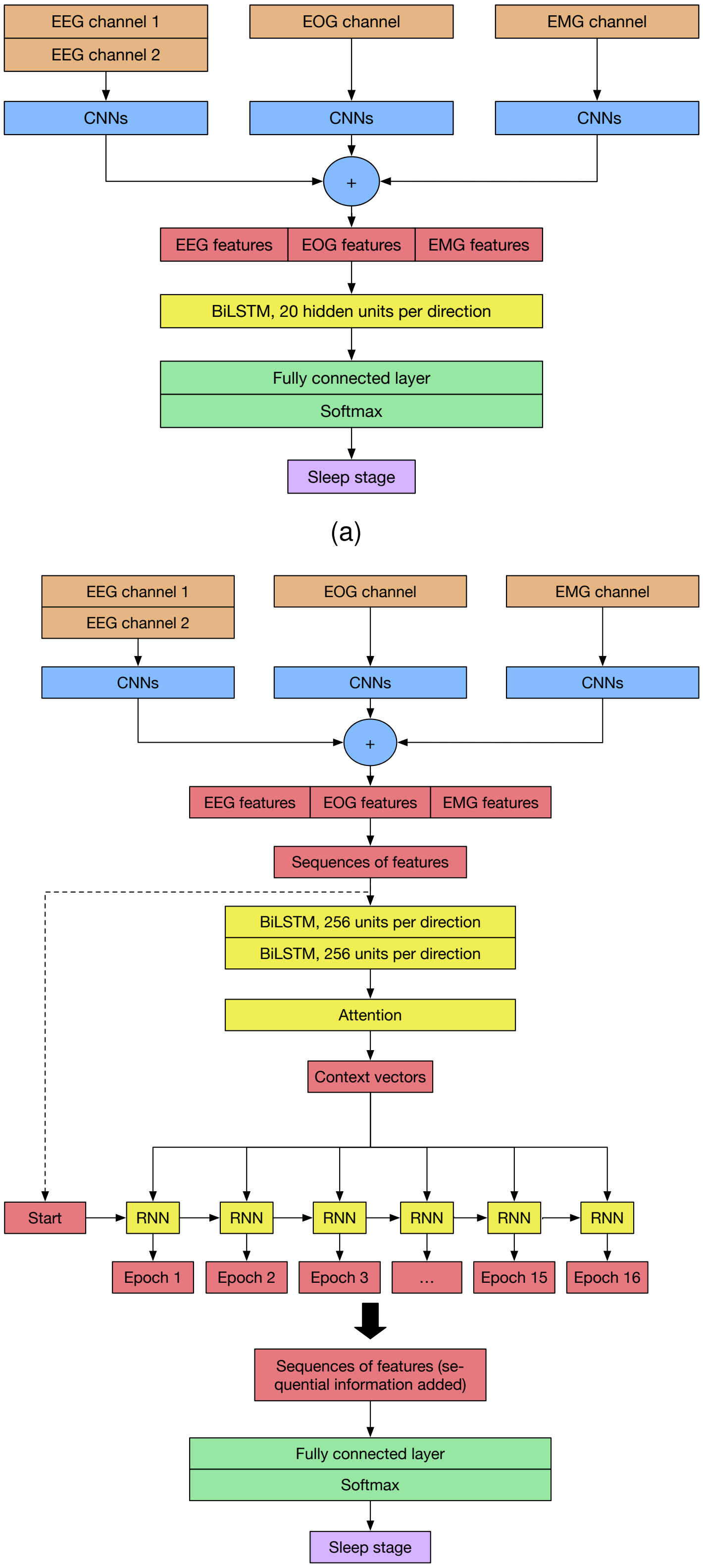


Figure 3.2: Baseline model architecture (a) and testing model architecture (b) for evaluating the module — increasing the depth of CNNs.



(a)

(b)

Figure 3.3: Baseline model architecture (a) and testing model architecture (b) for evaluating the module — applying the attention mechanism in sequential learning.

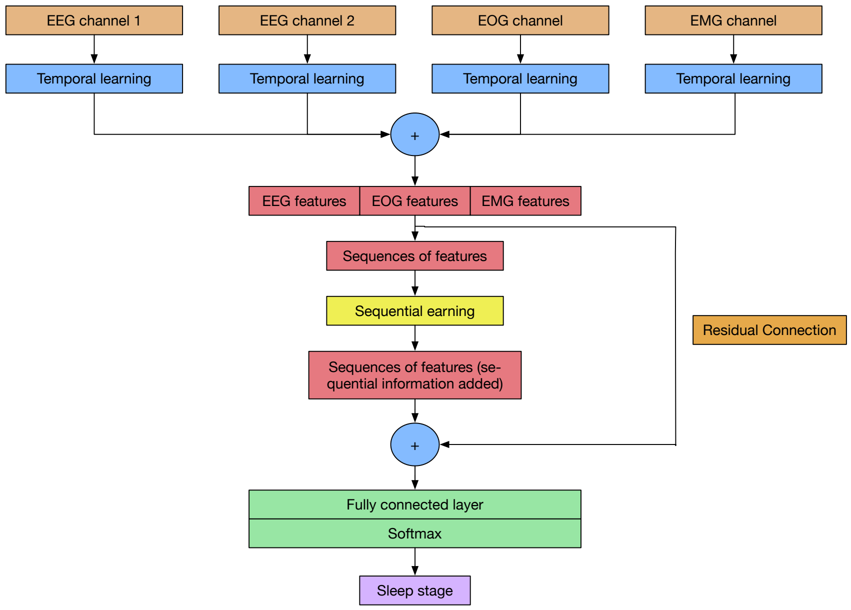


Figure 3.4: Testing model architecture for evaluating the module — adding the residual connection to final feature representations.

with that, data balancing techniques like oversampling data or applying weighted loss functions during training process were employed in previous studies like [11]. However, these data balancing techniques are used in the pre-training of the CNNs, as sequential learning requires sequential data where sleep stage instances cannot be arbitrarily duplicated. Therefore, the sequential learning process after temporal learning may again lead to the model focusing on learning majority classes. Additionally, sequential learning let the model understand transition rules from neighbouring sleep epochs, which may cause the loss of some time-invariant information of the current epoch. The residual connection can help with these problems through concatenating the time-invariant features of the current sleep epoch extracted by CNNs together with sequential information as the final feature representations.

The aim of this experiment is to investigate the necessity of applying the residual connection in our multi-channel sleep scoring model. According to the study performed by Pathak et al. [26], residual connections are not always required for sleep scoring models. Therefore, the test to evaluate the residual connection module is performed on our final model which combines all useful modules tested above. The testing model architecture of this experiment is plotted in Fig. 3.4. The temporal learning blocks refer to the CNNs applied with smaller and larger filters and deeper network depth, and the sequential learning block refers to the attention mechanism based sequential learning part. Performance metrics used for this comparison are kept the same as well.

3.1.2 Final Architecture of the Model

In this section, we introduce the final architecture of our multi-channel sleep scoring model and the data balancing techniques used in model training.

With the results obtained from the effective modules evaluation in Section 3.1.1, we design our multi-channel sleep scoring model mainly based on four parts: temporal learning, spatial learning, sequential learning and the residual connection. To exploit the benefits of the useful modules greatly, we adapt and optimize them in detail according to specific AASM rules. In addition, a new suitable spatial learning component is designed, and then a proper pipeline of all four parts is determined.

The model architecture designed for the SleepEDF-13 dataset (i.e. including 2 EEG channels, 1 EOG channel and 1 EMG channel) is plotted in Fig. 3.5 as the example to introduce. Generally, the temporal learning part consisting of two CNN pipelines for each channel is used to extract temporal features from the sleep epochs, and the small spatial learning part embedded in temporal learning is employed to extract correlation information among the channels of a modality. After that, the attention mechanism based sequential learning part is applied on concatenated features from all channels extracted by the temporal and spatial learning parts to add transition information from neighbouring epochs into final features. The residual connection is utilized to avoid the loss of time-invariant information and data balancing function by giving attention on both the sequential features from the neighbouring epochs and time-invariant features of the current epoch. Finally, a fully-connected layer followed by a softmax function is applied to classify sleep stages based on the final feature representations. Specific structures or techniques used with their functions and parameters are explained in the following paragraphs.

Temporal Learning

In our temporal learning, two convolutional layers with smaller and larger filter sizes are applied to extract time-domain patterns and frequency-domain features from 30-second sleep epochs of raw EEG, EOG and EMG signals, followed by deep convolutional layers to combine simpler features into complex features. Each filter in the first layer of the two CNN pipelines is employed to filter out one kind of the features, accordingly resulting in basic feature maps. The remaining layers are utilized to extract underlying information from the basic feature maps.

Specifically, each CNN pipeline in the model consists of four convolutional layers and two max-pooling layers, and each convolutional layer is followed by a batch nor-

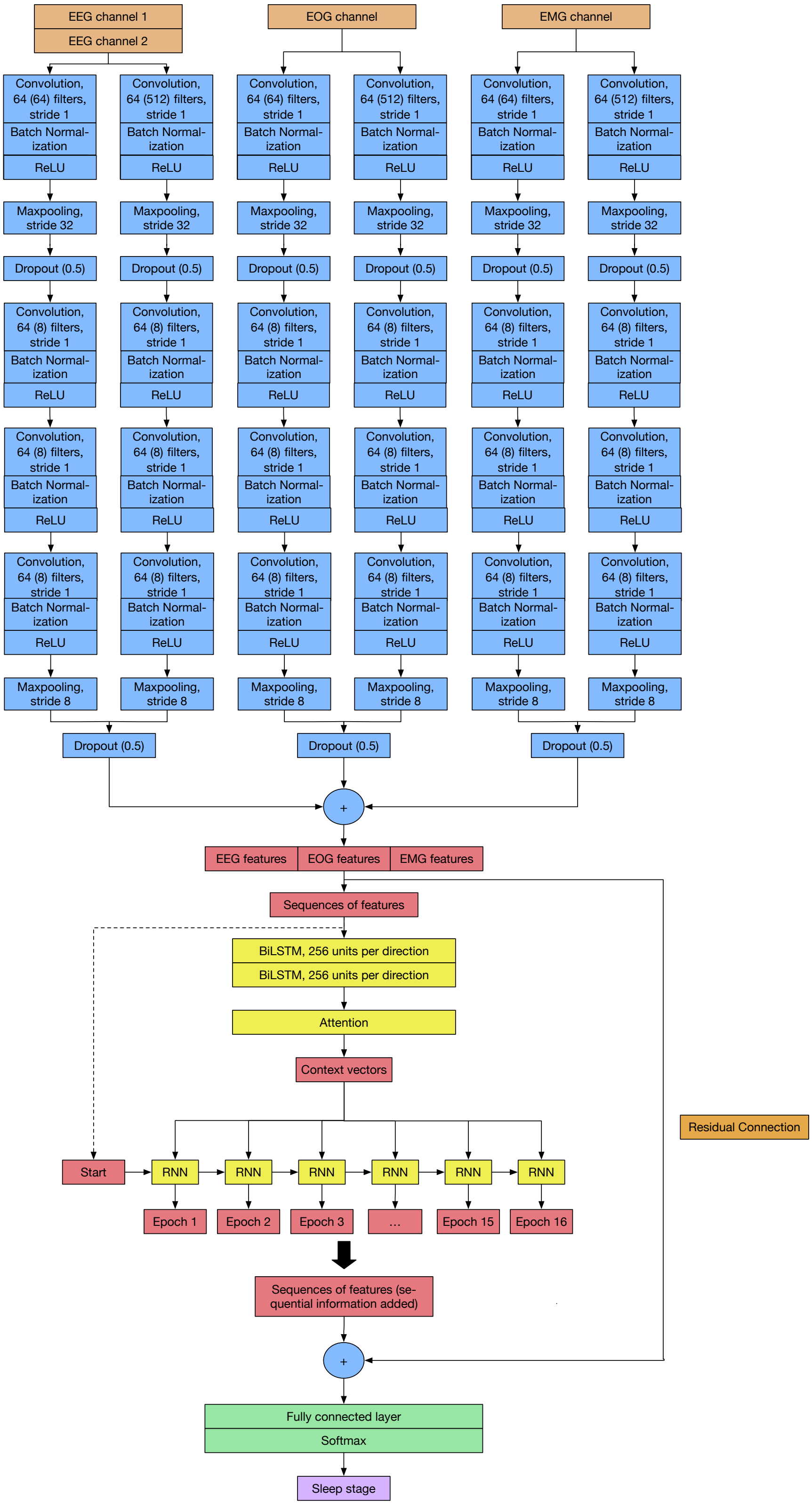


Figure 3.5: Final model architecture for multi-channel automatic sleep scoring

malization layer [38] and a rectified linear unit (ReLU) layer. The specifications of their number of filters, filter sizes, stride sizes and pooling sizes can be found in the model architecture in Fig. 3.5. The smaller filter size in the first layer is set to 64, as the distinctive time-domain patterns like K-complex and sleep spindles usually appear in a 0.5-second range of a sleep epoch while the sampling rates of signals are usually 100-125 Hz (i.e. 0.5 $\times$ (100 or 125) $\approx$ 64). The larger filter size is set to 512 in order to better detect frequency information of the signals. Different from [11], we set the stride size in the first convolutional layer to 1 instead of larger stride sizes to prevent the possible feature information loss. To avoid the overfitting it might bring, we set a larger pooling size in the max-pooling layer behind, which can filter out more representative and general features. At the end of the two CNN pipelines, the time-domain features and frequency-domain features are concatenated together as the time-invariant features of a sleep epoch. We also employ two dropout layers [39] as the regularization technique to prevent the overfitting in model training. The dropout probability is set to 0.5 and the dropout layers will expire during model evaluation.

Spatial Learning

We design a small spatial learning block embedded in the first convolutional layer of temporal learning to extract correlation information among the channels of a modality, as it has been shown in [40] that low temporal correlation can exist among EEG channels in Non-wake stages. This module is inspired from the work [41] that spatial learning can be implemented by a simple 1×1 convolutional layer with the filters (i.e. channels) dimensionality increase and reduction.

We first reshape the signals of a modality into an input of the size — [*Batch_size*, *No. channels in this modality*, *No. data points*], before they are passed into the first convolutional layer which is designed with *No. channels in this modality* input channels and 64 output channels. The spatial learning then can be implemented in the convolution transform by the channel dimensionality increase from *No. channels in this modality* to 64, as the calculation behind the convolution consists of two steps: 1) 64 smaller or larger convolutional filters are applied on each channel of this modality separately where corresponding basic feature maps of time-domain and frequency-domain patterns are obtained and 2) for each convolutional filter, the output feature maps of this modality is actually a weighted combination of the feature maps of its channels such that correlation information can be captured. The correlation information will be added into the extracted time-domain and frequency-domain feature maps with this spatial learning block. Compared to the study [26] where spatial learning is applied directly on raw signals, we apply our spatial learning module on the features maps extracted by smaller and larger filters in the first convolutional layer, which has the advantage that the distinctive time-domain patterns (e.g. K-complex and sawtooth waves) only existing in the raw signals will not be destroyed before they are recognized. Specifically, for the SleepEDF-13 dataset, only EEG modality has multiple channels. Therefore, we first divide the raw data into 3 modalities with their channels (i.e. 2 EEG channels, 1 EOG channel and 1 EMG channel). To extract the correlation from two EEG channels, the inputs of EEG modality are reshaped to the size — [*Batch_size*, *No. sub-channels: 2*, *No. data points: 3000*]. Then, they are passed into the first convolutional layer with 2 input filters and 64 output filters for smaller filter and lager filter pipelines each. The spatial learning on the EEG modality here is actually implemented by the channel dimensionality increase from 2 channels to 64 channels in the convolution transform.

Sequential Learning

We build the sequential learning framework, as shown in Fig. 3.5, to learn sequential information existing in a sleep sequence. According to the AASM rules, the current sleep stage can sometimes be determined by not only its time-invariant features but also some constraints from neighbouring epochs which is known as the transition rules. For example, stage N2 is usually scored for a sleep epoch where K-complex or sleep spindles occur. However, there is a circumstance where an epoch that has low amplitude should continue to be scored as N2 if its previous epoch is N2, even though particular patterns do not appear. Motivated by [13], we apply the attention mechanism into our sequential learning architecture to identify the most relevant parts of a sleep sequence and emphasize the sequential information belonging to the important parts.

Specifically, there are two phases in our sequential learning framework: the encoding phase which is used to understand the sequential information in sleep sequences and the decoding phase which is used to generate new feature representations for sleep sequences epoch by epoch. In the encoding phase, two bidirectional LSTM layers with 256 hidden units are employed to learn the original context dependencies from the concatenated CNN features of multiple modalities in both the forward and backward direction of sleep sequences. In the decoding phase, a RNN block with two LSTM layers and a Linear layer is used to generate target feature representations using the context dependency in a sequence-to-sequence way. With an attention module, the attention mechanism works, where the final context vector containing the context dependency for a particular sleep epoch in the sequence is created through allocating higher attention weights to more relevant parts. Concrete calculations to obtain the final context vector of a sleep epoch in the sequence and generate the target feature representation are expressed as follows:

$$e_i = mean(e_{i,for}, e_{i,back}) \tag{3.1}$$

$$f(d_t, e_i) = tanh(W_d d_t + W_e e_i) \tag{3.2}$$

$$a_i = softmax(f(d_t, e_i)) = \frac{exp(f(d_t, e_i))}{\sum_{j=1}^{n} exp(f(d_t, e_j))}, i \in (1, 2, ..., n) \tag{3.3}$$

$$c_t = \sum_{i=0}^{n} a_i e_i \tag{3.4}$$

$$FR_t = Linear(c_t || d_t) \tag{3.5}$$

where t denotes the current time step, $i \in (1, 2, ..., n)$ denotes n epochs of the sequence, e_i with $i \in (1, 2, ..., n)$ are the encoder's outputs (i.e. original context dependencies of the sequence), d_t is the output of decoder's RNN deriving from the feature representation of the last epoch, W_d and W_e are the weight matrices, $f(d_t, e_i)$ and a_i with $i \in (1, 2, ..., n)$ are the alignment scores and attention weights, c_t is the final context vector of the current sleep epoch, $||$ is a concatenate operation, FR_t is the feature representation of the current epoch, $Linear$ denotes the decoding function of the linear layer and $mean$, $tanh$ and $softmax$ are specific mathematical functions. The decoding iterations are repeated for each of the sleep epochs in the sequence so that sequences of features with sequential information added are obtained finally. There is also one thing to note that, in the training process of sequential learning, the start input of a sequence passed to the decoding phase is always set to the concatenated CNN features of the last epoch in the previous sequence, except for the first sleep sequence of a new subject. In that case, we just use the CNN features of the first epoch in that sequence instead as the start input.

Residual Connection

We design the residual connection that concatenates the CNN features of all modalities to the features output by the sequential learning part (see in Fig. 3.5). There are two motivations behind it as discussed in Section 3.1.1. Firstly, we consider the data imbalance problem in sleep scoring and apply data balancing techniques in the temporal and spatial learning parts. However, these techniques are not used in training the sequential learning part, as the training of the sequential learning part requires the sequential training set where sleep epochs can not be arbitrarily duplicated. Therefore, the sequential learning process will cause the model focusing on training majority classes again. The residual connection can help with this problem by reconsidering the CNN features which are learnt on a balanced dataset. Secondly, sequential learning let the model understand transition rules, which might cause some information loss of time-invariant features. The residual connection enables our model to consider both the temporal features of the current sleep epoch and the transition rules from neighbouring epochs simultaneously for sleep scoring.

In our model, the CNN features are directly side-by-side concatenated to the features deriving from the sequential learning part. After that, both of them are passed as the final feature representations to the fully-connected layer for sleep stage classification.

Data Balancing

Data imbalance is an important problem in sleep scoring, as stage N1 and N3 usually occur much less than other stages, which can be found from the instance information of the datasets in Table 4.1. According to [8], complex deep neural networks are usually biased to detecting majority classes better than minority classes. To solve this problem, we employ two data balancing techniques [42] — applying the weighted loss function (WLF) in training process and oversampling (OS) the instances of minority classes, to guarantee that all classes can be learnt equally in model training.

Specifically, for WLF, we calculate the cross entropy loss with the weighted function listed as follow:

$$W_c = 1 - \frac{N_c}{N} \tag{3.6}$$

where W_c is the weight for class c, N_c is the instances in class c and N is total number of instances in all classes. The weighted loss function can pay more attention to the loss from minority classes and optimize them better in model training. For OS, we perform two steps to let all classes contain an equal number of instances: 1) we duplicate all minority classes multiple times until their number of instances is very close to the largest class and 2) we then randomly duplicate single instances of the minority classes to guarantee all classes finally have the same number of instances in the training set. With the data balancing, all classes can receive equal attention in the loss back-propagation process of model training. We only apply the two data balancing techniques when training the temporal and sequential parts of our model (i.e. CNNs), as sequential arrangement in sequential learning will be destroyed if data balancing techniques are applied there.

3.2 Channel Importance Investigation

To investigate the channel importance in automatic sleep scoring, we employ two approaches — a post-hoc interpretability approach: the layer-wise relevance propagation (LRP) [35] and an intrinsic interpretability approach: the embedded channel attention network (Embedded CAN). Both of them are applied to the CNN part of our multi-channel sleep scoring model, not including the sequential learning part. Because we propose to focus solely on the time-invariant features of a sleep epoch that can be verified and explained by the patterns introduced in Table 1.1 from the AASM manual [2], while sequential information from neighbouring epochs will bring effects from neighbouring epochs. We also exclude the spatial learning part from our final model architecture such that the features of multiple channels can be learnt separately, as we are more interested in the single contribution of a channel. The corresponding experiments above are implemented on the SHHS-1 dataset, as it includes a broad range of subjects (5783 subjects) which can result in a general result of the channel importance. Channel importance scores obtained from both methods for the sleep epochs are summed up and averaged according to particular stages, resulting in an importance score per channel per stage. Specific importance score calculations for each method are described in Section 3.2.1 and 3.2.2.

We also extend the channel importance study to finding the most significant features (i.e. either the time-domain features like distinctive 0.5-second patterns and amplitude information or the frequency-domain features) for each EEG, EOG and EMG channel with the LRP method. Corresponding details are introduced in Section 3.2.3.

3.2.1 Layer-wise Relevance Propagation

Inspired by [31], the importance of a part in the data inputs to the prediction of a deep neural network can be inferred from the relevance of that part to the prediction. Therefore, we employ the LRP method to calculate the relevance scores of the inputs for the CNN part of our model. There are two steps in LRP: 1) a standard forward pass of sleep data is first implemented to collect the activation in each layer of our CNN parts and 2) predictions are then propagated backwards until the input layer using a specific propagation rule to calculate the relevance of the activation for each layer. The propagation rule in [17] is used to transfer the relevance from layer k (the following layer) to layer j (the preceeding layer):

$$R_j = \Sigma_k(\alpha \frac{a_j w_{jk}^+}{\Sigma_j a_j w_{jk}^+} - \beta \frac{a_j w_{jk}^-}{\Sigma_j a_j w_{jk}^-}) R_k \tag{3.7}$$

where R_j and R_k are the relevance at layer j and k, a_j is the activation at layer j and w_{jk}^{+} and w_{jk}^{-} denote the positive and negative connections between layer j and k respectively. We select the propagation rule version of $\alpha = 1$ and $\beta = 0$ here, as the channel importance we intend to find is defined as the positive contribution of a channel to detecting a particular sleep stage. There are two constraints for the propagation rule listed above: 1) the activation functions we use in our multi-channel sleep scoring model should be all non-negative and monotonically increasing and 2) the activation in every preceeding layer (including the inputs) that are represented as a_j in the rule should be non-negative as well. The ReLU function we use as the activation function in the CNN part meets the first requiremen. However, our inputs (i.e. EEG, EOG and EMG signals) are sometimes negative which violates the second constraint. To solve it, we adapt the propagation rule to the following:

$$R_j = \Sigma_k(\alpha\frac{(a_j w_{jk})^{+}}{\Sigma_j(a_j w_{jk})^{+}} - \beta\frac{(a_j w_{jk})^{-}}{\Sigma_j(a_j w_{jk})^{-}})R_k \tag{3.8}$$

by considering the $a_j w_{jk}$ as a whole, such that, if the multiplication of the input data and its weight is positive it actually plays a positive contribution to the output activation of that layer because we always use a non-negative and monotonically increasing activation function. The relevance score of a channel is calculated through summing up the relevance of all data points per channel per test sleep epoch, and the final channel importance are obtained by averaging the channel relevance scores per stage.

3.2.2 Embedded Channel Attention Network

To investigate the channel importance in multi-channel sleep scoring in an intrinsic interpretation way, we design the novel method — a channel attention module embedded in our deep automatic sleep scoring model, for channel attention identification. The architecture is shown in Fig. 3.6. For each channel, two CNN pipelines as we discussed in Section 3.1.2 are employed to extract the time-invariant features from the signals. The features from smaller filters and larger filters are concatenated together for each channel. In channel attention identification, the embedded channel attention module is developed which takes the features of all channels as the input and outputs an attention weight vector for the channels. Specifically, in this module, the features of each channel is firstly passed into a global average pooling layer to generate one feature representation for each channel. The representations of the channel are then put into a 2-layer conditional neural network to calculate the attention weight vector. After that, the attention weights obtained are normalised by a softmax function and the features of each channel are self-normalised as well.

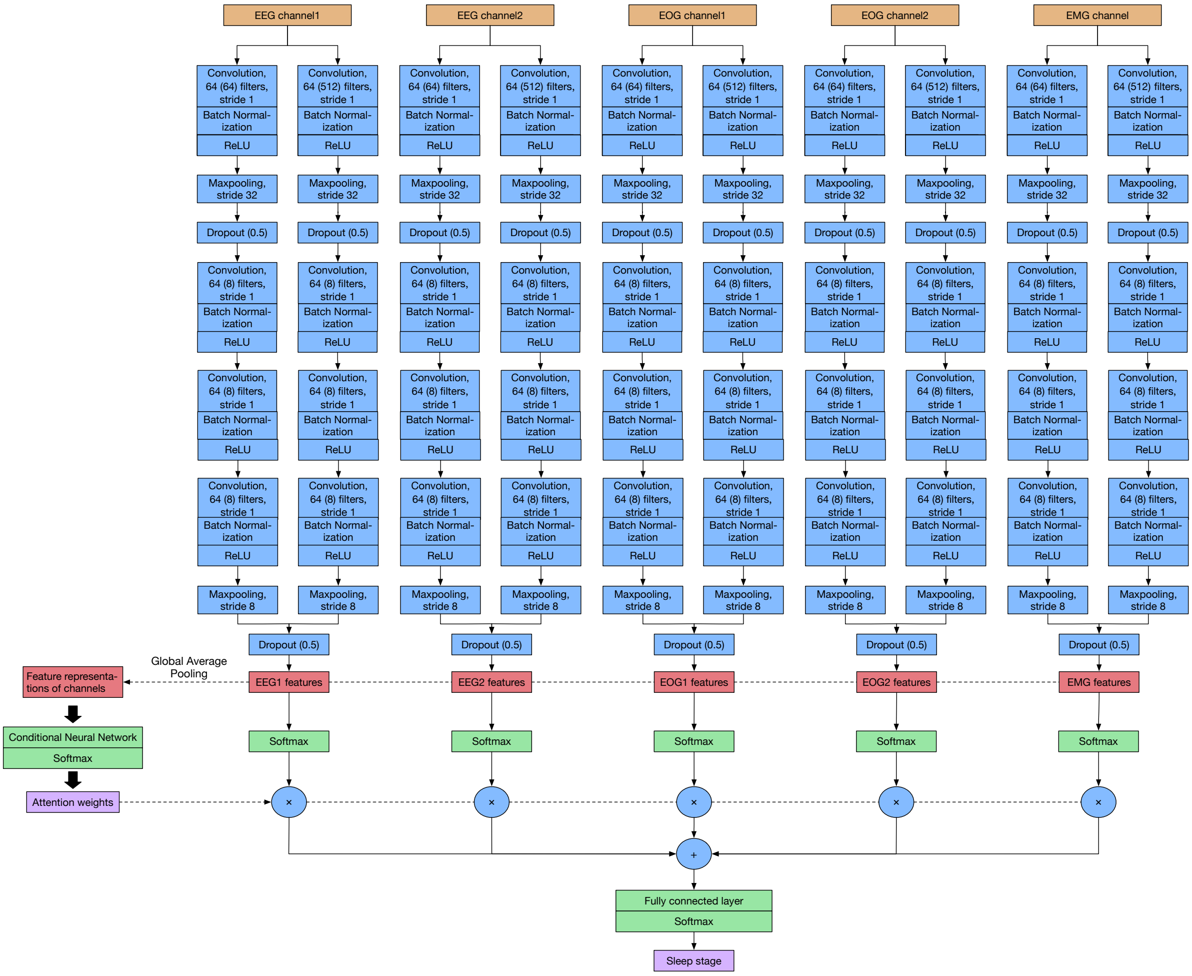


Figure 3.6: Architecture of embedded channel attention identification

Finally, the normalised attention weights are multiplied to the features of the corresponding channels, resulting in channel attention applied features which are passed to the fully-connected layer with a softmax function for sleep stage classification. The sleep scoring model with the Embedded CAN block is trained in the similar way to the pre-training of the CNN part in our multi-channel model with necessary data balancing techniques. With a trained model, the channel importance scores are obtained by summarizing the channel attention weights of all test sleep epochs grouped by particular stages and then averaging them per stage.

3.2.3 Feature Importance Analysis

According to the AASM rules, important features (i.e. either the time-domain features or the frequency-domain features) vary in EEG, EOG and EMG modalities. For example, EEG signals mainly include two kinds of patterns — distinctive 0.5-second pattern like K-complex and sleep spindles and dominant frequency components like alpha and beta waves. EOG signals mainly consist of amplitude features but also a few frequency features, and EMG signals consist of only amplitude features and no frequency features. Therefore, understanding their significant features in an experimental way is meaningful for the further improvement of sleep scoring by paying more attention to the important features in the model. To analyse the feature importance, we stick to the LRP method which is set as the baseline method in channel importance investigation and calculate the relevance scores for each kind of features per channel per stage. Discussed in Section 3.1.2, the smaller filters in our CNN part are utilized to extract time-domain features like the distinctive 0.5-second patterns for EEG signals and amplitude patterns for EOG and EMG signals, and larger filters are targeted at extracting frequency components from all signals. Therefore, we calculate the importance scores of these features by summarizing the relevance scores over the activation of the smaller and larger filters respectively.

Chapter 4

Experimental Setup

In this chapter, we introduce the datasets, the training algorithm, the experimental designs, the evaluation metrics and the training parameters and implementation for evaluating our multi-channel sleep scoring model. In addition, the experiments designed for channel importance investigation are described as well.

4.1 Datasets and Data Pre-processing

We evaluate our multi-channel automatic sleep scoring model with various signals of EEG, EOG and EMG modalities on two public datasets: the SleepEDF-13 dataset (39 PSGs $\approx$ 42 thousand epochs) and the SHHS-1 dataset (5783 PSGs $\approx$ 6 million epochs). Table 4.1 provides an overview of the datsets.

Dataset	Sleep Stages					Total Samples
	Wake	N1	N2	N3	REM	
SleepEDF-13	8,285 (19.6%)	2,804 (6.6%)	17,799 (42.1%)	5703 13.5%)	7,717 (18.2%)	42,308
SHHS-1	1,691,288 (28.8%)	217,583 (3.7%)	2,397,460 (40.9%)	739,403 (12.6%)	817,473 (13.9%)	5,863,207

Table 4.1: Overview of the evaluation datasets with number of 30-second epochs and class proportions.

4.1.1 SleepEDF-13

The SleepEDF-13 dataset [27], [28] is a small dataset, where two studies were performed: investigating age effect in healthy subjects (SC) and investigating Temazepam effects on sleep (ST). The subject set we use comes from the SC study. There are 20 subjects (age: 28.7 ± 2.9) in the dataset, and each subject contains 2 polysomnograms except for one subject (i.e. resulting in 39 polysomnograms totally) where 2

EEGs (channel Fpz-Cz and Pz-Cz), 1 EOG (horizontal) and 1 EMG were recorded. The EEG and EOG signals were sampled at 100 Hz while the EMG signals were sampled at 1 Hz. Sleep epochs in this dataset were manually annotated into one of the eight stages (Wake, N1, N2, N3, N4, REM, Movement, Unscored) according to the R&K manual [1]. We merge N3 and N4 into N3 to comply with the AASM manual [2] and remove the Movement and Unscored epochs which are meaningless to sleep scoring. In addition, following [11], we also exclude long wake periods that are located 30 minutes before and after sleep periods.

4.1.2 SHHS-1

The Sleep Heart Health Study (SHHS) dataset [30] is a large dataset established for sleep-disordered breathing researches launched by the National Heart Lung and Blood Institute, United States. There are 2 visits for data collection, and the subject set we use is the first visit (SHHS-1). Overall, 5783 subjects (age $\geq$ 40) from both genders participated, resulting in 5783 PSGs. Each PSG recorded 2 EEGs (channel C3-A2 and C4-A1), 2 EOGs (left and right) and 1 EMG. The EEG and EMG signals were sampled at 125 Hz while the EOG signals were sampled at 50 Hz. Similar to the SleepEDF-13 dataset, the R&K manual [1] is used for manual sleep scoring of the SHHS-1 dataset, resulting in eight sleep stages. We unify the annotations to comply with the AASM manual [2] by combining N3 and N4 as N3 and remove the epochs annotated as Movement and Unscored.

4.1.3 Data Pre-processing

We apply the same data pre-processing process on the SleepEDF-13 and SHHS-1 datsets. Firstly, we resample the signals with smaller sampling rates to the highest sampling rate among all modalities for each dataset (i.e. resampling EMG signals in the SleepEDF-13 dataset to 100 Hz and EOG signals in the SHHS-1 dataset to 125 Hz) such that all modalities can share an identical feature extraction mechanism in our sleep scoring model. However, we do not apply extra resampling on the two datasets to unify their signals with the same sampling rate, as their sampling rates are actually quite close with each other and will not affect the model performance (i.e. $0.5 \times (100 \text{ or } 125) \approx 64$; thus the filter of size 64 can detect time-domain patterns for both datasets). The only difference when applying our model on these two datsets will be the different feature map sizes. Secondly, following the study performed by Pathak et al. [26], we filter EEG and EOG signals of both two datasets to 0.16-30 Hz and EMG signals to 10-30 Hz as suggested by sleep experts, and standardize the signal of each channel to mean 0 and standard deviation 1.

4.2 Training Algorithm

We employ a two-step training algorithm to address the data imbalance problem and train our automatic sleep scoring model by loss back-propagation.

In the first step, we pre-train the temporal and spatial learning parts (i.e. CNN pipelines) of our model with one of the two data balancing techniques discussed in Section 3.1.2, WLF or OS. The extracted time-invariant features are directly passed into an extra fully connected layer with the softmax function for sleep stage classification. The extra fully-connected layer is only active in the pre-training and will be discarded after that. This step enables our pre-trained model to capture the time-invariant information of a sleep epoch precisely and learn minority classes equally compared to majority classes.

In the second step, we freeze the parameters of the temporal and spatial learning parts and only train the sequential learning and residual connection parts to add transition information and concatenate both time-invariant features and sequential features together as final feature representations. The final feature representations are used to classify the sleep stage by a fully-connected layer with the softmax function at the end of our model architecture. This step guarantees the features learnt in temporal and spatial learning will not be discarded after sequential learning and prevents the loss of the data balancing function caused by the sequential training set.

For both two training steps, the cross-entropy loss is used to measure the agreement between the predicted sleep stages determined by our model and the ground truth. We also prepare the validation set and employ the early stopping technique [43] to stop the training. The early stopping technique has a parameter — the early stopping patience. The patience set to k means that if the validation loss does not decrease for k training iterations, the training will stop.

4.3 Experimental Designs

This section introduces our experiments designed to test effective modules for our multi-channel model, evaluate our automatic sleep scoring model on the SleepEDF-13 and SHHS-1 datasets and investigate channel importance on the SHHS-1 dataset.

4.3.1 Effective Modules Evaluation

To prepare the dataset for effective modules evaluation, we randomly shuffle the data in the small SleepEDF-13 dataset and build a sample dataset based on this through splitting the shuffled dataset on subject level into 81$\%$ for training, 9$\%$ for validation and 10$\%$ for testing. With the sample dataset, we perform four experiments corresponding to four testing modules: 'using CNNs with different filter sizes', 'increasing the depth of CNNs', 'applying the attention mechanism in sequential learning' and 'adding the residual connection to final feature representations', to verify their usefulness for multi-channel sleep scoring respectively. There is no data balancing technique used in the first two experiments which only trains the CNN part, while in the latter two experiments, the data balancing technique — WLF is used in the pre-training of the model. To train a model, the early stopping on validation loss with a patience of 7 is used for the baseline models of the first three experiments as used by Pathak et al. in their study [26], while a patience of 16 is used for the remaining testing models. Because new modules complicate the models, so that more iterations have to be taken in training.

4.3.2 Final Model Evaluation

For final model evaluation, on top of the two-step training algorithm, we design distinctive evaluation experiments to train and test our multi-channel sleep scoring model on the SleepEDF-13 and SHHS-1 datasets separately, as they differ greatly in the dataset size.

For the **SleepEDF-13** dataset, we design an adapted nested cross validation scheme (see Fig. 4.1) to evaluate our model since the SleepEDF-13 dataset is quite small and only contains 20 subjects (39 PSGs). In the scheme, the outer loop is a 20-fold cross validation corresponding to 20 subjects, which is used to estimate the model performance on the SleepEDF-13 dataset globally. Specifically, in an outer fold k, we leave out one of the 20 subjects as the test set k at a time. After 20 iterations, we summarize the results of all 20 test sets together to obtain the results of the whole dataset. Every inner loop is a 10-fold cross validation to estimate the model performance on a test set k. We micro-average the results from the 10 models training on 10 random training-validation combinations and testing on the test set k to reduce the possible bias resulting from training on a fixed and small validation set. Finally, we combine the results of 200 (i.e. 20 outer loops $\times$ 10 inner loops) sets and calculate the performance metrics on the resulting confusion matrix to illustrate our model performance on the SleepEDF-13 dataset.

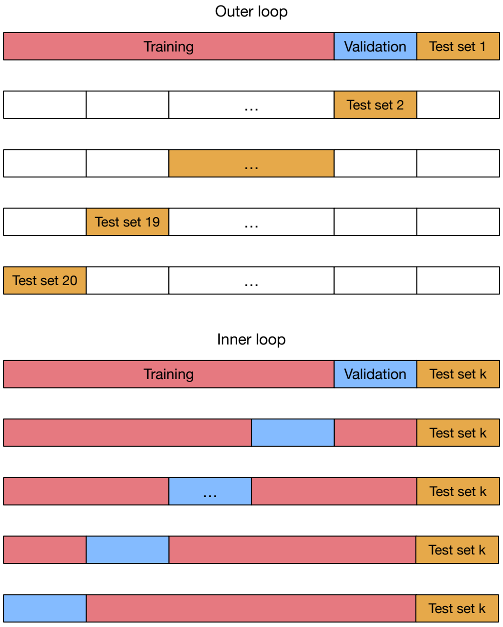


Figure 4.1: Scheme of the adapted nested cross validation used to evaluate our multi-channel automatic sleep scoring model on the small SleepEDF-13 dataset.

For the **SHHS-1** dataset, we randomly shuffle the subjects and split the dataset into 81% for training, 9% for validation and 10% for testing, as the SHHS-1 dataset is much larger and contains 5783 subjects (5783 PSGs). Based on that, we train our models on the training set and stop the training using the early stopping technique on the validation set. Finally, we report our model performance on the SHHS-1 dataset by testing the trained model on the test set and calculating performance metrics on the resulting confusion matrix.

4.3.3 Channel Importance Investigation

For channel importance investigation, we select the SHHS-1 dataset as the experimental object, as it is a large dataset and it is more reasonable to calculate the channel importance scores from various subjects. Additionally, it has one more channel than the SleepEDF-13 dataset so that we can obtain more channel information accordingly. In the training process of both two methods introduced in Section 3.2, we exclude the spatial learning part from our sleep scoring model. Between the two data-balancing techniques discussed in Section 3.1.2, the WLF is selected, as it gives better performance than OS, which is shown in the evaluation of our sleep scoring model. On top of that, the channel importance investigation based on the LRP and Embedded CAN methods are implemented separately, and accordingly the importance score matrices are calculated. It has to be noted that, the initial model training for both methods are performed on the whole SHHS-1 dataset, while the importance score generation and visualisation are finally performed on 20 patient data randomly selected from the dataset, due to a hardware limitation of our laptop's RAM.

4.4 Evaluation Metrics

Following most existing work like [11]–[13], we evaluate the performance of our model based on the following metrics: the overall accuracy (ACC), the macro F1-score (MF1), the Cohen's kappa (κ) [44] and the per-class F1-score (pF1). Among the metrics, ACC shows a general performance of our model. MF1 considers both the precision and recall, which can help reflect the detection performance on minority classes. κ measures the agreement between our sleep scoring model and manual sleep scoring implemented by sleep experts. pF1 shows the detection performance on a specific class. The formulas to calculate them are listed as follows:

$$ACC = \frac{\Sigma_{c=1}^{C} TP_c}{N} \tag{4.1}$$

$$MF1 = \frac{\Sigma_{c=1}^{C} pF1_c}{N} \tag{4.2}$$

$$\kappa = \frac{p_o - p_e}{1 - p_e} \tag{4.3}$$

$$p_e = \Sigma_{c=1}^{C} \frac{n_{cg}}{N} \frac{n_{cp}}{N} \tag{4.4}$$

$$pF1_c = \frac{2}{\frac{1}{Pr_c} + \frac{1}{Re_c}} \tag{4.5}$$

where c is a class of sleep stages, C is the number of the classes, TP_c is the true positive of class c, N is the total number of epochs, $pF1_c$ is the per-class F1-score of class c, p_o is the relative agreement between the ground truth and the predictions, p_e is the hypothetical probability of chance agreement, n_{cg} is the number of epochs in the ground truth for class c , n_{cp} is the number of epochs in the predictions for class c, Pr_c is the precision of class c and Re_c is the recall of class c.

4.5 Training Parameters and Implementation

We use the same training optimizer Adam [45] with the parameters — learning rate, beta1 and beta2, set to 0.0001, 0.9 and 0.999, in both the pre-training and final-training steps for our model. And the early stopping patience in both two training steps are set to 16[1]. Specifically, for the temporal and spatial learning parts, we pretrain them by mini-batch gradient descent with the two data balancing techniques, WLF and OS. Following [26], the mini-batch size is set to 192, as a sleep cycle usually lasts around 96 minutes (i.e. 30 seconds × 192). We hope that one mini-batch training can cover all sleep stages. When training the sequential learning and residual connection parts, we freeze the parameters in the temporal and spatial learning parts. The mini-batch gradient descent is also used, and the mini-batch size and sequence length are set to 32 and 16 respectively[2]. Additionally, due to that the number of sleep epochs in a PSG might not be multiple times of the sequence length: 16, we generate the sequential training set by padding the starting epochs of a PSG to the ending epochs to guarantee the ending epochs can be enrolled into a training or testing sequence of 16 sleep epochs as well.

Our models and corresponding evaluation experiments are implemented using PyTorch [46], and the training and testing processes are run on a high performance cluster (https://fmt.ewi.utwente.nl/redmine/projects/ctit_user/wiki/HPC) with multiple CPUs and GPUs.

[1] We test the patience ranging from 10 to 20, 16 gives the best results.

[2] We apply a grid search to find the best combination of them from various mini-batch sizes: (8,16,32) and sequence lengths: (8,16,32,64) (i.e. parameter ranges used by most existing work), and the mini-batch size: 32 and sequence length: 16 give the best results.

Chapter 5

Results and Discussion

In this chapter, we present the results of the experiments performed for testing effective modules, evaluating our multi-channel sleep scoring model and inferring the channel importance scores. A model analysis for our automatic sleep scoring, a comparison to state-of-the-art models and an analysis on the usefulness of channel and feature importance are discussed further.

5.1 Results

The results section is divided into three sub-sections. Section 5.1.1 shows the testing results of the effective modules we evaluate compared to corresponding baseline models. Section 5.1.2 presents the evaluation performance of our final multi-channel sleep scoring model on the SleepEDF-13 and SHHS-1 datasets. Section 5.1.3 shows the channel and feature importance scores we obtain in channel importance investigation.

5.1.1 Effective Modules Evaluation

The performance of all modules we test and their baseline models are summarized in Table 5.1. Four rows refer to the four evaluating experiments respectively: 'using CNNs with different filter sizes', 'increasing the depth of the CNNs', 'applying the attention mechanism in sequential learning' and 'adding the residual connection to final feature representations'. In each row, the first line refers to the result of the baseline model while the second line refers to the result of the testing model architecture. All details of the baseline models and the testing model architectures can be found in Section 3.1.1.

Obviously, all modules we test have an improvement for sleep scoring compared to corresponding baseline models. Specifically, the module of adding filters of larger size to the CNNs shows an increased performance in detecting all stages especially for stage N1 and N3. The reason may be that, the number of the N1 instances is very small and their time-domain patterns are not distinguishable, so that using larger filters to recognize frequency-domain features can help detect them. For stage N3, its frequency-domain pattern is the Delta wave which only appears in N3, therefore, adding larger filters to capture frequency features improves the performance of N3 a lot. The module of increasing the depth of CNNs mainly improves the detection of sleep stages (except for REM) with a sacrifice of stage Wake. It is because there is no complex pattern for stage Wake but some for sleep stages, by increasing the complexity of feature extraction, the underlying rules from the basic feature maps of sleep stages can be captured which is useful in detecting them. The module of applying the attention mechanism in sequential learning presents great improvements in detecting almost all stages especially for stage N1 and N3 as well. The reason is, in a sleep period, stage N1 and N3 occur much less frequently than other stages (see Table 4.1), which increasing the difficulty of the detection only based on the time-invariant features of them. The attention mechanism help this case through paying more attention to relevant epochs in sleep sequences, which emphasizes important transition rules, so that more instances of stage N1 and N3 can be correctly classified. The module of applying the residual connection to concatenate the CNN features to final feature representations outperforms the architecture without that, verifying the necessity to avoid the loss of the time-invariant information and the data balancing function, as discussed in Section 3.1.1. Overall, from the results above, it can be concluded that all four testing modules we find out and select from the literature can be exploited and transferred to multi-channel automatic sleep scoring, which contributes a lot in designing our final model architecture.

Modules	**Models**	**Acc**	**MF1**	κ	**pF1**				
					Wake	**N1**	**N2**	**N3**	**REM**
Using CNNs with different filter size	Baseline	82.6	70.6	0.74	88.3	22.4	87.4	73.5	81.4
	Testing	**83.5**	**73.6**	**0.76**	**88.6**	**31.5**	**87.9**	**77.6**	**82.3**
Increasing the depth of CNNs	Baseline	82.6	70.6	0.74	**88.3**	22.4	87.4	73.5	**81.4**
	Testing	**82.9**	**72.7**	**0.75**	84.6	**27.6**	**88.1**	**82.1**	81.0
Applying the attention mechanism in sequential learning	Baseline	85.4	73.3	0.79	**87.5**	21.2	89.1	82.6	86.0
	Testing	**85.8**	**76.6**	**0.79**	87.0	**32.0**	**89.5**	**87.7**	**86.5**
Adding the residual connection to final feature representations	Baseline	85.4	75.8	0.79	86.6	32.5	89.3	**82.4**	**88.0**
	Testing	**86.1**	**77.2**	**0.80**	**89.7**	**37.4**	**89.7**	81.5	87.6

Table 5.1: Results for effective modules evaluation across ACC, MF1, κ and pF1 on a sample dataset (discussed in Section 4.3.1) generated from the SleepEDF-13 dataset.

5.1.2 Automatic Sleep Scoring

The evaluation performance of our multi-channel automatic sleep scoring model on two public datasets: SleepEDF-13 and SHHS-1, are shown in Table 5.2. Here, 'Pre-trained' refers to the result of the CNN part trained in the first step of our two-step training algorithm to extract temporal and spatial features and 'Final-trained' refers to the final result of the whole model after the two-step training. Two data balancing techniques: weighted loss function (WLF) and oversampling (OS) are applied in the evaluation on both datasets, resulting in totally 8 results.

Generally speaking, our sleep scoring model can achieve the accuracy of around 85%, the macro F1 score of around 78% and the Cohen's kappa of around 0.80 when tested on both datasets. Summaries can be obtained that the 'Final-trained' models, with the sequential learning and residual connection parts, improves the scoring results a lot especially in the detection of some particular stages compared to the 'Pre-trained' model, and the model using the weighted loss function as the data balancing technique in training performs consistently better on both datasets. Specifically, for F1-scores of particular stages, the model evaluated on the SleepEDF-13 dataset have a better performance in stage N1 and N3 while the model evaluated on the SHHS-1 dataset have a better performance in stage Wake and REM, which may be because N1 and N3 have a lower proportion of instances in the SHHS-1 dataset (see Table 4.1) so that they are more difficult to recognize there. Additionally, if we compare the performance of the models using different data balancing techniques in model training, we can find that, for the 'Pre-trained' models, the weighted loss function performs much better than the oversampling technique, as the oversampling technique pays too much attention to the minority classes and misclassifies lots of majority class instances to them. However, for the 'Final-trained' models, the distinction of these two techniques almost disappears, which may be because the sequential dataset used in sequential learning is the original dataset without data balancing that brings some attention back to the majority classes when training the model. In order to keep the positive contribution of the data balancing techniques, the residual connection actually works by concatenating the CNN features which are learned equally for all classes to final feature representations.

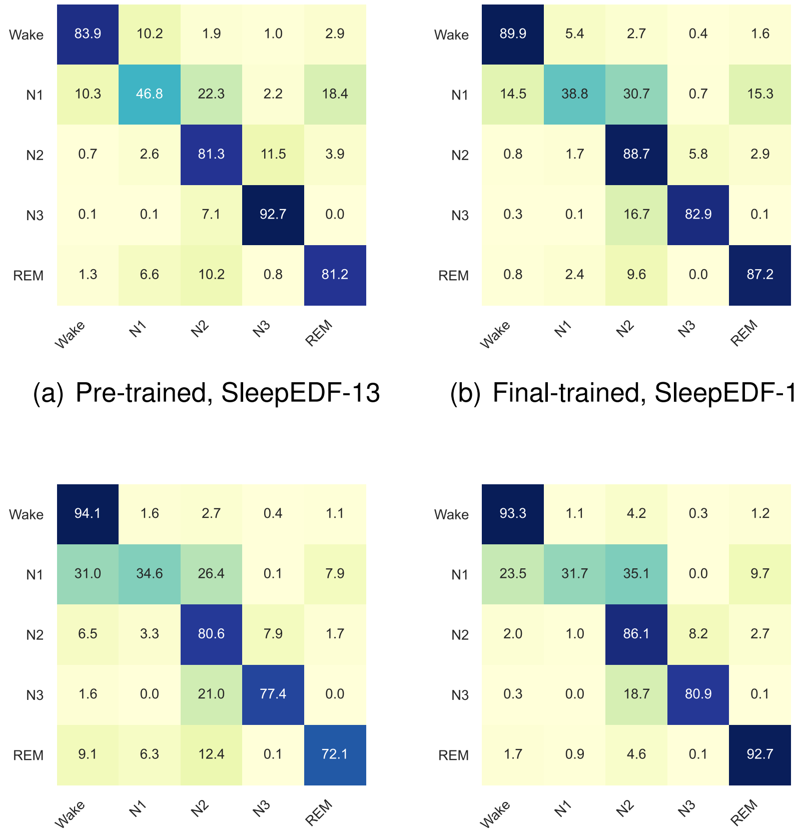
To have a deeper understanding of the functions of our sequential learning modules, we select the better models which apply the weighted loss function as the data balancing technique in model training and plot the row-wise normalised confusion matrices of their 'Pre-trained' models and 'Final-trained' models for both the SleepEDF-13 and SHHS-1 datasets in Fig. 5.1. The rows indicate the actual classes while the columns indicate the predicted classes. It can be found that, on both

Models	**Dataset**	**Acc**	**MF1**	κ	**pF1**				
					Wake	**N1**	**N2**	**N3**	**REM**
Pre-trained^{WLF}	SleepEDF-13	81.0	75.6	0.74	88.3	44.2	84.5	79.9	81.1
Pre-trained^{OS}	SleepEDF-13	74.9	71.2	0.67	84.1	40.7	80.0	74.5	76.9
Final-trained^{WLF}	SleepEDF-13	**84.6**	**78.3**	**0.79**	**91.1**	**45.0**	**86.8**	**82.1**	**86.7**
Final-trained^{OS}	SleepEDF-13	84.4	78.1	0.78	91.1	44.5	86.6	81.9	86.6
Pre-trained^{WLF}	SHHS-1	81.1	72.0	0.74	88.5	33.5	82.3	76.1	79.5
Pre-trained^{OS}	SHHS-1	77.7	68.8	0.69	88.6	33.5	81.0	65.0	75.7
Final-trained^{WLF}	SHHS-1	86.4	**77.7**	**0.81**	**93.2**	**41.0**	86.2	**77.9**	**90.3**
Final-trained^{OS}	SHHS-1	**86.5**	77.3	0.81	93.2	39.2	**86.5**	77.5	90.3

Table 5.2: Performance of our multi-channel sleep scoring model across ACC, MF1, κ and pF1 on two public datasets - SleepEDF-13 and SHHS-1.

datasets, there are several stage detection result transitions from the 'Pre-trained' model to the 'Final-trained' model with the application of sequential learning. Firstly, many N1 instances are misclassified to stage Wake and N2. Considering stage N1 is a minority class while Wake and N2 are majority classes, this transition illustrates that the retrieval of N1 relies more heavily on its time-invariant features learnt from a balanced dataset rather than on the transition rules. Secondly, the retrieval of stage Wake, N2 and REM recovers from the influence of data balancing technique where they are misclassified to the minority classes after sequential learning. It reflects the transition rules that, for these main stages (i.e. Wake, N2, REM) in sleep periods, the epochs following such stages will continue to be scored as the same stage if no distinctive criteria of other stages is met, which complies with the AASM rules. Thirdly, if we look at the detail information for stage REM, we can find that sequential learning has a high contribution to detecting REM, as it helps reduce the misclassification from REM to the other stages. For stage N3, the retrieval performance differs on two datasets that, many N3 instances are misclassified to N2 after sequential learning on the SleepEDF-13 dataset while it shows the opposite case on the SHHS-1 dataset. This may be due to the different sizes of two datasets that the SHHS-1 dataset has much more training examples to learn, therefore, the features and rules helping in detecting N3 can be captured better. Overall, if we compare the retrieval performance of the 'Pre-trained' and 'Final-trained' models to the class proportions of the original dataset (see Tab 4.1), conclusions can be reached that, the 'Pre-trained' models can retrieve minority classes better with data balancing techniques and the sequential learning brings transition information into sleep stage classification which leads to some detection result transitions from a stage to another.

To visualise the performance of our multi-channel sleep scoring model intuitively, Fig. 5.2 demonstrates an example of the hypnograms that manually scored by the sleep expert and automatically scored by our model for a subject in the SHHS-1



	Wake	N1	N2	N3	REM
Wake	83.9	10.2	1.9	1.0	2.9
N1	10.3	46.8	22.3	2.2	18.4
N2	0.7	2.6	81.3	11.5	3.9
N3	0.1	0.1	7.1	92.7	0.0
REM	1.3	6.6	10.2	0.8	81.2

(a) Pre-trained, SleepEDF-13

	Wake	N1	N2	N3	REM
Wake	89.9	5.4	2.7	0.4	1.6
N1	14.5	38.8	30.7	0.7	15.3
N2	0.8	1.7	88.7	5.8	2.9
N3	0.3	0.1	16.7	82.9	0.1
REM	0.8	2.4	9.6	0.0	87.2

(b) Final-trained, SleepEDF-13

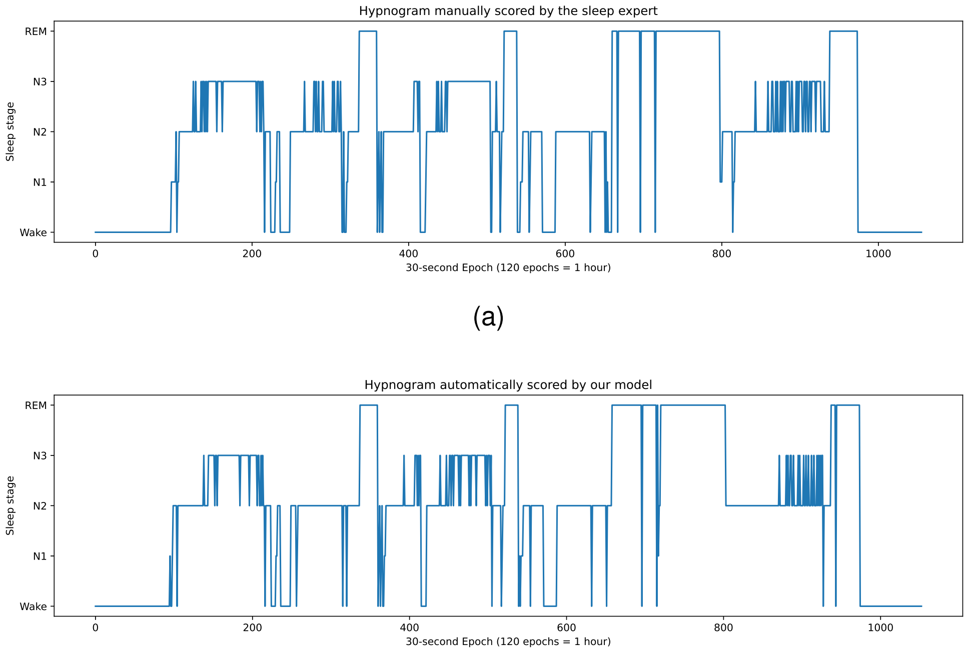
	Wake	N1	N2	N3	REM
Wake	94.1	1.6	2.7	0.4	1.1
N1	31.0	34.6	26.4	0.1	7.9
N2	6.5	3.3	80.6	7.9	1.7
N3	1.6	0.0	21.0	77.4	0.0
REM	9.1	6.3	12.4	0.1	72.1

(c) Pre-trained, SHHS-1

	Wake	N1	N2	N3	REM
Wake	93.3	1.1	4.2	0.3	1.2
N1	23.5	31.7	35.1	0.0	9.7
N2	2.0	1.0	86.1	8.2	2.7
N3	0.3	0.0	18.7	80.9	0.1
REM	1.7	0.9	4.6	0.1	92.7

(d) Final-trained, SHHS-1

Figure 5.1: Raw-wise normalised confusion matrices of our 'Pre-trained' and 'Final-trained' models on the SleepEDF-13 and SHHS-1 datasets.



(a)

(b)

Figure 5.2: Example of the hypnogram manually scored by the sleep expert (a) and the hypnogram automatically scored by our model (b) for a subject from the SHHS-1 dataset.

dataset. The example we choose reflects the average sleep scoring performance of our model (i.e. Acc $\approx 86\%$). With a detailed comparison, it can be found that the detection of many N3 epochs fails and the misclassification sometimes happens on Wake and N1 epochs as well.

5.1.3 Channel and Feature Importance Investigation

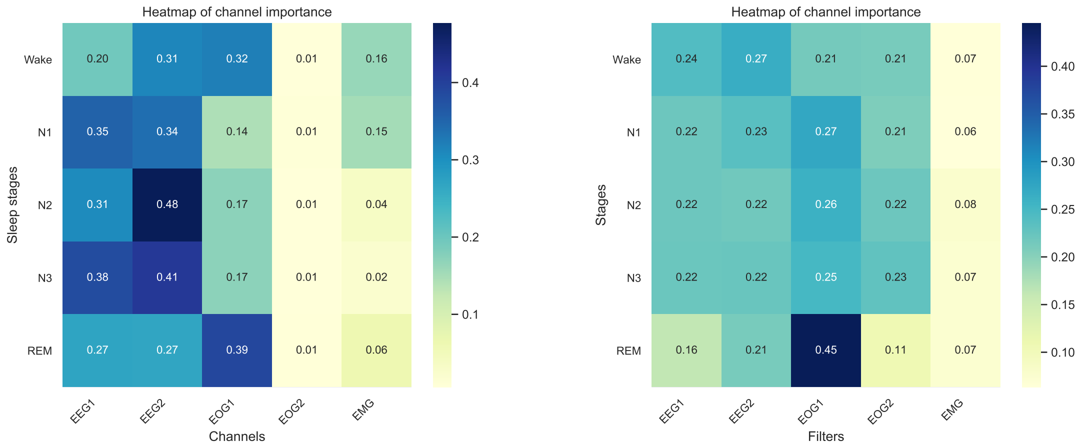
To visualize the channel importance scores calculated by two approaches discussed in Section 3.2, we plot two heatmaps that show the importance of the channels (5 columns) in the SHHS-1 dataset to particular sleep stages (5 rows) accordingly in Fig. 5.3. The importance scores are normalised by row so that how much a channel relevant to a stage can be found clearly.

According to the result from **LRP**, the main information in our automatic sleep scoring comes from EEG channels, which account for around 60-70 $\%$ for every stage. EOG channels are the second significant channels for sleep stage classification, and the EMG channel is the least important. Comparing the importance scores within EEG and EOG channels separately, we can find that both EEG channels have almost equal contribution but only one EOG channel is mainly used by the automatic sleep scoring model. The results illustrate that the sensors located at different positions on our brain (i.e. EEG sensors) record various sleep information that represents our sleep status. The inclusion of multiple EEG channels can help improve the performance of sleep scoring. Contrarily, most information in the two symmetric EOG channels (i.e. one is close to the left eye, the other is close to the right eye) are nearly identical and the utilization of one of them may be sufficient for sleep scoring. In addition, it can be obtained from the detailed importance scores that, EOG channel 1 has a very high importance score for stage Wake and REM, which is even higher than EEG channels. This matches the AASM rules, which state that eye movements are very typical features for Wake and REM.

According to the result from **Embedded CAN**, EEG channels and EOG channels almost have the same importance in sleep stage classification, and the EMG channel has nearly no contribution. When comparing the detailed importance scores within two EEG channels and two EOG channels, an illustration different from the results from LRP can be obtained, both EEG channels and both EOG channels have similar importance in detecting a particular stage. Additionally, EOG channel 1 has a very high importance score for stage REM perfectly complying with its name: Rapid Eye Movement, which indicates that eye movement is the most significant feature for stage REM.

As discussed in Section 2.2, the LRP method is set as the baseline method for channel importance investigation, as LRP has been found successfully applicable for this kind of problems [31] and can have excellent benchmark performance [37] while the intrinsic interpretability method may suffer from a trade-off between accuracy and interpretability [36]. In our experiments, the similar finding is obtained that the result from LRP complying with the AASM manual [2] better than Embedded CAN. Overall, the two methods we apply show similar results that EEG and EOG channels are more important than the EMG channel in sleep stage detction and EOG channel 1 is pretty important for identifying stage REM, while they show different result when inferring the importance of two EOG channels.

We also perform an analysis to find the **important features** (i.e. either time-domain features or frequency-domain features) for detecting a particular stage. We calculate the importance scores for the CNNs with smaller filters and larger filters respectively by LRP. The heatmap of the feature importance score matrix is plotted in Fig. 5.4. Here, the temporal features of a channel refer to the time-domain patterns extracted by the smaller filter in CNNs like the 0.5-second patterns: K-complex and sleep spindles in EEG signals and amplitude patterns in EOG and EMG signals, while the



(a) Importance score matrix from LRP

(b) Importance score matrix from Embedded CAN

Figure 5.3: Row-wise normalised channel importance score matrices calculated by LRP (a) and Embedded CAN (b) on 20 subjects' data randomly selected from the SHHS-1 dataset.

frequency features of a channel refer to the main frequency-domain components of the signals. It can be found from the matrix that temporal features, in all channels, have higher importance in sleep scoring than frequency features. Frequency components are usually similar among different channels of a modality, which means the frequency feature extraction on one channel of that modality might be sufficient. Additionally, EEG temporal features show higher importance in classifying the stages, while EOGs and EMG have smaller contributions. The reason is that the time-domain patterns in EEG signals like K-complex and sleep spindles vary in different stages, but only few stages have special amplitude patterns in EOG and EMG signals. Specifically, complying with the AASM rules, eye movements which can be detected by the amplitude changes mainly exist in stage Wake and REM, such that EOG temporal features of these two stages perform higher importance than the other stages. Similar result can be obtained for EMG signals where the chin movement mainly happens in stage Wake and at most in stage N1 in most cases, which explains why EMG temporal features have a high importance only in detecting stage Wake and N1. In addition, useful frequency features only exist in EEG and EOG signals, as frequency patterns discrimination appears there and EMG signals have no important frequency feature according to the AASM rules.

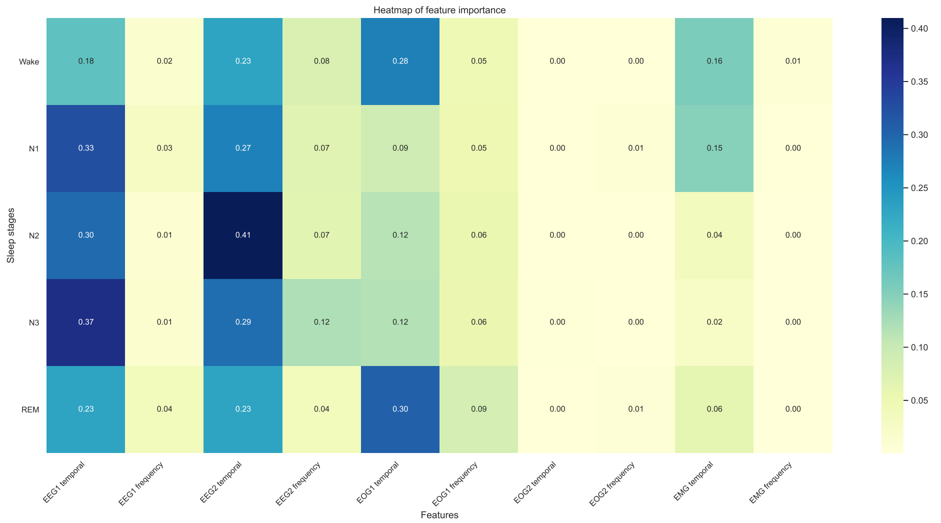


Figure 5.4: Row-wise normalised feature importance score matrix calculated by LRP on 20 subjects' data randomly selected from the SHHS-1 dataset.

5.2 Discussion

In this section, we analyse the advantages of our model and present our contributions, followed by a comparison to the state-of-the-art. In addition, we discuss the value of inferring channel importance scores by comparing it to the AASM rules and analyse its potential for further improving automatic sleep scoring.

5.2.1 Model Analysis

We develop a multi-channel automatic sleep scoring model in this research, which is inspired from the AASM rules that all three modalities (i.e. EEG, EOG and EMG) have useful information in classifying sleep stages. Our model is designed based on two phases. In the first phase, we review some contributing modules in single-channel sleep scoring from the literature and evaluate their effectiveness for our multi-channel study. In the second phase, we design a new model architecture consisting of temporal learning, spatial learning, sequential learning and the residual connection with the help of combining and adapting the effective modules. The model achieves good performance on two public datasets.

It can be concluded from our experiment results in the first phase (see Table 5.1) that, all four modules we test: 1) using CNNs with different filter sizes to extract time-domain and frequency-domain features respectively, 2) increasing the depth of CNNs to capture complex patterns, 3) apply the attention mechanism in sequential learning to emphasize important parts of sleep sequences when learning transition information and 4) adding the residual connection to avoid possible model degradation, can help improve the multi-channel sleep scoring model especially in detecting minority classes. In terms of the advantages, to our knowledge, our study is the first study to build a successful multi-channel automatic sleep scoring model utilizing the benefits of the modules from previous single-channel studies. The better evaluation results on two public datasets over the state-of-the-art in multi-channel sleep scoring (see Table 5.3) verify the correctness of the utilization of effective modules. However, due to the time limitation, an extended optimization of the multi-channel sleep scoring model has not been implemented based on the results of our channel and feature importance investigation.

In addition, our model is trained with a two-step training algorithm. We first pre-train the CNNs to capture temporal and spatial features with two data balancing methods: WLF and OS. After that, the sequential learning and residual connection parts are finally trained on a sequential dataset. The results of the two-step training algorithm in Fig. 5.1 show that the functions of the data balancing techniques are sometimes covered by the effects of transition information retrieved by the sequential learning part even though the residual connection has been applied. Comparing the performance of two data balancing techniques we use in Table 5.2, we can find that the oversampling technique can help retrieve the minority classes better but also misclassifies too many majority class instances to them. However, both of them obtain similar performance on final-trained models. According to the findings above, it is necessary and interesting to convey a further research to investigate and address the data imbalance problem in the whole training process of the model, which might help improve the sleep scoring performance.

5.2.2 Comparison to the State-of-the-art

Table 5.3 shows an overall comparison of the performance of our model on the SleepEDF-13 and SHHS-1 datasets with the state-of-the-art. Our multi-channel automatic sleep scoring model achieves the Acc of 84.6$\%$ and 86.4 $\%$, MF1 of 78.3$\%$ and 77.7$\%$ and κ of 0.79 and 0.81 for the SleepEDF-13 and SHHS-1 datasets respectively, which outperforms all relevant existing work in multi-channel sleep scoring[1], proving our work a better model in this field. Compared to the single-EEG based approaches, our model has the best Acc and κ on the SleepEDF-13 dataset

[1] It has to be noted that, we use early stopping to stop the model training so that we have to separate out the validation data in the evaluation, which means we always train our model on less data (i.e. excluding the validation data). Even though, our model still performs better.

but fails to outperform [13] in MF1 which shows our model does not outperform theirs in detecting some particular classes especially minority classes. The reason may be that, in [13], Mousavi et al. applied the synthetic minority oversampling techniques and an extra weighted loss function in sequential learning which can better detect minority classes, while we only employ one of two data balancing techniques in the pre-training of our model. For the SHHS-1 dataset, our model prevails in comparison to [25] but fails to outperform [12], as the latter used a different data processing method and excluded some subjects from the SHHS-1 dataset according to certain criteria when evaluating their model. Reviewing Table 5.2, the pF1s of our model are all above 85.0 except N1 and N3 which are lower than those in some previous studies based on single-EEG signal. The reason may be that, the inclusion of EOG and EMG signals can confuse the detection of N1 and N3 as their patterns in EOG and EMG signals are not that distinguishable, as can be seen in the feature importance matrix (see Fig. 5.4). Additionally, we also test the result of utilizing the two data balancing techniques — weighted loss function and oversampling, which are most widely used by most existing work. The weighted loss function gives a slight better result.

Methods	**Dataset**	**Channels**	**Evaluation**	**Acc**	**MF1**	κ
Tsinalis et al. [10]	SleepEDF-13	1 EEG	20-fold CV	74.8	70.0	-
Supratak et al. [11]	SleepEDF-13	1 EEG	20-fold CV	82.0	77.0	0.76
Mousavi et al. [13]	SleepEDF-13	1 EEG	20-fold CV	84.3	**80.0**	0.79
Paisarnsrisomsuk et al. [22]	SleepEDF-13	2EEGs+1EOG	4-fold CV	81.0	-	-
Phan et al. [23]	SleepEDF-13	1EEG+1EOG+1EMG	20-fold CV	82.3	-	-
Our model^{WLF}	SleepEDF-13	2EEGs+1EOG+1EMG	20-fold NestedCV	**84.6**	78.3	**0.79**
Our model^{OS}	SleepEDF-13	2EEGs+1EOG+1EMG	20-fold NestedCV	84.4	78.1	0.78
Biswal et al. [25]	SHHS-1	1 EEG	90-10	77.9	-	0.73
Sors et al. [12]	SHHS-1	1 EEG	50-20-30	**87.0**	**78.0**	0.81
Pathak et al. [26]	SHHS-1	2EEGs+2EOGs+1EMG	81-9-10	85.0	76.6	0.79
Our model^{WLF}	SHHS-1	2EEGs+2EOGs+1EMG	81-9-10	86.4	77.7	**0.81**
Our model^{OS}	SHHS-1	2EEGs+2EOGs+1EMG	81-9-10	86.5	77.3	0.81

Table 5.3: Performance comparison between our model and the state-of-the-art across ACC, MF1, κ and pF1 on two public datasets: SleepEDF-13 and SHHS-1. - in the table indicates that the value is not available in the respective publication.

5.2.3 Channel and Feature Importance

The channel and feature importance plotted in Fig. 5.3 and 5.4 show that the inclusion of all channels is not necessary for automatic sleep scoring; the inclusion of frequency features from some channels as well. For example, from the channel importance information calculated by LRP, the EOG channel 2 in the SHHS-1 dataset obtains only around 1$\%$ importance for detecting all 5 stages, which means the symmetric eye movement sensors contain almost identical sleep information and EOG channel 2 is actually not needed in sleep staging. From the channel importance obtained by Embedded CAN, the EMG channel shows small importance for sleep scoring meaning that the chin movement may not required. Though the two methods (i.e. LRP and Embedded CAN) give different importance results for some channels which has to be further studied and verified, we can reach the finding that some channel information is not necessary to consider so that the feature extraction may not be required on these channels. Similarly, for frequency features captured from the EMG channel, it shows no significance to all stages other than very little to stage Wake, indicating that the frequency information extraction in EMG channel is not required in sleep scoring as well, which complies with the AASM rules. EEG channels behave contrarily, various EEG channels are very effective in sleep stage detection as they contain distinctive valuable temporal features. However, the frequency features of both EEG channels seem identical, which means the detection from only one of two EEG channels is sufficient for sleep scoring.

The findings and discussion above can provide a guidance for improving the multi-channel sleep scoring model in the future, as we can preserve the raw data from important channels as model inputs and keep useful filters for feature extraction as well while deleting the useless ones. Additionally, according to [24], the additional modalities of EOG and EMG can improve performance over only using EEG, but the performance is not improved further when adding too many EEG channels, as 6 EEG channels gave the best results in their experiment. Due to the variety of EEG channels, it is difficult to figure out the best combination of the channels only through constantly testing the possible combinations. Therefore, channel and feature importance analysis can be performed as an initial experiment to find the effective channels and features, which is meaningful for optimizing the data collection and data utilization in future automatic sleep scoring.

Chapter 6

Conclusions and Future Work

In this chapter, we answer the research question in Chapter 1. Following that, we briefly conclude our contributions and findings. In the end, the future work is discussed.

6.1 Research Questions

The answers to our two research questions are as follows:

1. (RQ1) *What can be a well-performing model for multi-channel automatic sleep scoring?*

For RQ1, the final well-performing multi-channel automatic sleep scoring model we develop consists of the temporal learning, spatial learning, sequential learning and the residual connection parts that can extract time-invariant features of one sleep epoch and transition rules of sleep sequences. The architecture benefits from effective modules inspired by single-channel sleep scoring models, and a spatial learning block is embedded in it which can capture the correlation information among the channels of a modality. Good results have been achieved for our model, in that the overall accuracy, the macro F1-score and the Cohen's kappa can reach around 85$\%$, 78$\%$ and 0.8 on both public datasets. To our knowledge, our model outperforms most of the state-of-the-art as well.

2. (RQ2) *How much does the information of each channel contribute to sleep scoring?*

For RQ2, we apply two methods: LRP and Embedded CAN to calculate the channel importance score showing the contribution of a channel to a particular sleep stage. If we group the importance scores by the modalities, the results of both methods (see Fig. 5.3) illustrate similarly that EEG and EOG channels are very important in detecting all stages while the EMG channel is less used. Specifically, according to LRP, one EOG channel is sufficient for sleep scoring and the other EOG channel is almost not used, while Embedded CAN shows that both EOG channels are supposed to be employed with almost equal attention. Further, the feature importance of each channel calculated by LRP (see Fig. 5.4) demonstrates, the significant features of all channels are time-domain patterns and only EEG channels and EOG channels contain important frequency-domain features. Specific importance scores normalised over all stages can be inquired from the matrices.

6.2 Conclusions

In conclusion, we perform two sub-studies in our master assignment.

Firstly, we propose a multi-channel automatic sleep scoring model based on the utilization and adaptation of effective modules inspired from previous single-channel studies. We also design a new spatial learning part to extract the correlation information among the channels of a modality without destroying the time-domain information (e.g. K-complex and sleep spindles) of raw signals. We apply a two-step training algorithm to train our model and employ two data balancing techniques in the pre-training part to address the data imbalance problem. The evaluation of our model is performed on two public datasets — SleepEDF-13 and SHHS-1 — where better performance is obtained compared to the state-of-the-art. Specifically, for the SleepEDF-13 dataset, our model can predict sleep stages with an overall accuracy of 84.6% and stage Wake, N1, N2, N3 and REM with F1 scores of 91.1%, 45.0%, 86.8%, 82.1% and 86.7% respectively. And for the SHHS-1 dataset, our model can predict sleep stages with an overall accuracy of 86.4% and stage Wake, N1, N2, N3 and REM with F1 scores of 93.2%, 41.0%, 86.2%, 77.9% and 90.3%.

Secondly, the channel importance is investigated through a post-hoc interpretation method — layer-wise relevance propagation and an intrinsic interpretation method — embedded channel attention network, where the importance score per channel per sleep stage is obtained. Further, feature importance of each channel is analysed with corresponding scores as well. The results mainly show two findings: 1) EEG and EOG channels have relatively higher importance in sleep scoring than the EMG channel and not all channels are necessarily required to extract representative features for a stage and 2) the time-domain features are more significant than frequency-domain features and the frequency-domain features of the stages mainly exist in EEG and EOG signals. To summarize, the channel and feature importance illustrate the contribution of a channel and a feature for sleep stage classification, which is meaningful to understand how our multi-channel sleep scoring model works and explore the potential of channel and feature information for optimizing automatic sleep scoring.

6.3 Future Work

Due to the time limitation and current lock-down situation, some of the planned work is cancelled, such as collecting data from clinical cases, evaluating our model based on clinical data and discussing and verifying the channel importance conclusions with sleep experts other than simply comparing them to the AASM manual [2]. Therefore, there are some possibilities that our study can be extended to in future work.

Firstly, it is necessary to test our model on clinical data for more comprehensive performance evaluation, as the aim of automatic sleep scoring is to develop better and more generalizable sleep stage classification approaches, which can be extended to different datasets and used to help sleep technicians analyse and detect sleep disorders efficiently in clinical cases.

Secondly, it is necessary to deeply understand and verify the channel and feature importance we obtain, as the results from the two methods — LRP and Embedded CAN — show both similarities and differences. An extended research by discussing the results with clinical sleep experts and designing further experiments to verify the importance of a channel or a feature, can finally help build a standard mechanism for analysing channel and feature importance in sleep scoring.

Thirdly, if we take the channel and feature importance information into consideration when developing the multi-channel sleep scoring model, we can design a model which is more efficient to extract useful features from the raw signals by paying more attention to important channels and features. Therefore, it is interesting to convey a study to optimize the construction of multi-channel sleep scoring models based on the knowledge of channel and feature importance.

Bibliography

[1] E. A. Wolpert, “A manual of standardized terminology, techniques and scoring system for sleep stages of human subjects,” *Archives of General Psychiatry*, vol. 20, no. 2, pp. 246–247, 1969.

[2] C. Iber, S. Ancoli-Israel, A. L. Chesson, S. F. Quan *et al.*, *The AASM manual for the scoring of sleep and associated events: rules, terminology and technical specifications*. American Academy of Sleep Medicine Westchester, IL, 2007, vol. 1.

[3] T. L. T. da Silveira, A. J. Kozakevicius, and C. R. Rodrigues, “Single-channel eeg sleep stage classification based on a streamlined set of statistical features in wavelet domain,” *Medical & biological engineering & computing*, vol. 55, no. 2, p. 343—352, 2017.

[4] A. R. Hassan and M. Bhuiyan, “A decision support system for automatic sleep staging from eeg signals using tunable q-factor wavelet transform and spectral features,” *Journal of Neuroscience Methods*, vol. 271, pp. 107–118, 2016.

[5] O. Tsinalis, P. Matthews, and Y. Guo, “Automatic sleep stage scoring using time-frequency analysis and stacked sparse autoencoders,” *Annals of biomedical engineering*, vol. 44, pp. 1587–1597, 2015.

[6] M. Sharma, D. Goyal, A. PV, and U. R. Acharya, “An accurate sleep stages classification system using a new class of optimally time-frequency localized three-band wavelet filter bank,” *Computers in Biology and Medicine*, vol. 98, pp. 58–75, 2018.

[7] K. Pillay, A. Dereymaeker, K. Jansen, G. Naulaers, S. Huffel, and M. de Vos, “Automated eeg sleep staging in the term-age baby using a generative modelling approach,” *Journal of Neural Engineering*, vol. 15, no. 3, p. 036004, 2018.

[8] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[9] L. Cen, Z. L. Yu, Y. Tang, W. Shi, T. Kluge, and W. Ser, “Deep learning method for sleep stage classification,” in *Neural Information Processing*. Springer International Publishing, 2017, pp. 796–802.

[10] O. Tsinalis, P. M. Matthews, Y. Guo, and S. Zafeiriou, “Automatic sleep stage scoring with single-channel eeg using convolutional neural networks,” *ArXiv*, vol. abs/1610.01683, 2016.

[11] 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, 2017.

[12] A. Sors, S. Bonnet, S. Mirek, L. Vercueil, and J.-F. Payen, “A convolutional neural network for sleep stage scoring from raw single-channel eeg,” *Biomedical Signal Processing and Control*, vol. 42, pp. 107–114, 2018.

[13] 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, 2019.

[14] O. Yildirim, U. B. Baloglu, and U. R. Acharya, “A deep learning model for automated sleep stages classification using psg signals,” *International journal of environmental research and public health*, vol. 16, no. 4, p. 599, 2019.

[15] Y. Wang and D. Wu, “Deep learning for sleep stage classification,” *2018 Chinese Automation Congress (CAC)*, pp. 3833–3838, 2018.

[16] S. Khalighi, T. Sousa, G. Pires, and U. Nunes, “Automatic sleep staging: A computer assisted approach for optimal combination of features and polysomnographic channels,” *Expert Systems with Applications*, vol. 40, p. 7046–7059, 2013.

[17] G. Montavon, W. Samek, and K.-R. Müller, “Methods for interpreting and understanding deep neural networks,” *Digital Signal Processing*, vol. 73, pp. 1–15, 2018.

[18] A. Malafeev, D. Laptev, S. Bauer, X. Omlin, A. Wierzbicka, A. Wichniak, W. Jernajczyk, R. Riener, J. Buhmann, and P. Achermann, “Automatic human sleep stage scoring using deep neural networks,” *Frontiers in Neuroscience*, vol. 12, p. 781, 2018.

[19] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, “Understanding neural networks through deep visualization,” *arXiv preprint arXiv:1506.06579*, 2015.

[20] A. I. Humayun, A. S. Sushmit, T. Hasan, and M. I. H. Bhuiyan, “End-to-end sleep staging with raw single channel eeg using deep residual convnets,” *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, pp. 1–5, 2019.

[21] J. Wang, Y. Zhang, Q. Ma, H. Huang, and X. Hong, “Deep learning for single-channel eeg signals sleep stage scoring based on frequency domain representation,” in *HIS*, 2019.

[22] S. Paisarnsrisomsuk, M. Sokolovsky, F. Guerrero, C. Ruiz, and S. A. Alvarez, “Deep sleep: convolutional neural networks for predictive modeling of human sleep time-signals,” *KDD Deep Learning Day*, 2018.

[23] H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. De Vos, “Joint classification and prediction cnn framework for automatic sleep stage classification,” *IEEE Transactions on Biomedical Engineering*, vol. 66, no. 5, pp. 1285–1296, 2018.

[24] 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, 2018.

[25] S. Biswal, H. Sun, B. Goparaju, M. B. Westover, J. Sun, and M. T. Bianchi, “Expert-level sleep scoring with deep neural networks,” *Journal of the American Medical Informatics Association*, vol. 25, no. 12, pp. 1643–1650, 2018.

[26] S. Pathak, C. Seifert, and M. van Putten, “Deepsleep: Deep learning for automatic sleep scoring,” Dutch-Belgian Database Day, 2019. [Online]. Available: http://www.cslab.cc/dbdbd2019/abstracts/dbdbd2019-pathak.pdf

[27] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. Ivanov, R. Mark, J. Mietus, G. Moody, C. Peng, and H. Stanley, “Physiobank, physiotoolkit, and physionet : Components of a new research resource for complex physiologic signals,” *Circulation*, vol. 101, no. 23, pp. E215–E220, 2000.

[28] B. Kemp, A. H. Zwinderman, B. Tuk, H. A. C. Kamphuisen, and J. J. L. Oberye, “Analysis of a sleep-dependent neuronal feedback loop: the slow-wave microcontinuity of the eeg,” *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 9, pp. 1185–1194, 2000.

[29] C. O'Reilly, N. Gosselin, J. Carrier, and T. Nielsen, "Montreal archive of sleep studies: an open-access resource for instrument benchmarking and exploratory research." *Journal of sleep research*, vol. 23, no. 6, pp. 628–635, 2014.

[30] S. F. Quan, B. V. Howard, C. Iber, J. P. Kiley, F. J. Nieto, G. T. O'Connor, D. M. Rapoport, S. Redline, J. Robbins, J. M. Samet *et al.*, "The sleep heart health study: design, rationale, and methods," *Sleep*, vol. 20, no. 12, pp. 1077–1085, 1997.

[31] M. D. Boehle, F. Eitel, M. Weygandt, and K. Ritter, "Layer-wise relevance propagation for explaining deep neural network decisions in mri-based alzheimer's disease classification," *Frontiers in Aging Neuroscience*, vol. 11, 2019.

[32] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7132–7141, 2018.

[33] Q. Wang, B. Wu, P. Zhu, P. Li, W. Zuo, and Q. Hu, "Eca-net: Efficient channel attention for deep convolutional neural networks," *ArXiv*, vol. abs/1910.03151, 2019.

[34] A. Bastidas and H. Tang, "Channel attention networks," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 881–888, 2019.

[35] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PLoS ONE*, vol. 10, 2015.

[36] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (xai)," *IEEE Access*, vol. 6, pp. 52 138–52 160, 2018.

[37] W. Samek, A. Binder, G. Montavon, S. Lapuschkin, and K.-R. Müller, "Evaluating the visualization of what a deep neural network has learned," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, pp. 2660–2673, 2017.

[38] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[39] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[40] Y. Li, X. Tang, Z. Xu, W. Liu, and J. Li, “Temporal correlation between two channels eeg of bipolar lead in the head midline is associated with sleep-wake stages,” *Australasian Physical & Engineering Sciences in Medicine*, vol. 39, pp. 147–155, 2015.

[41] M. Lin, Q. Chen, and S. Yan, “Network in network,” *CoRR*, vol. abs/1312.4400, 2014.

[42] J. L. Leevy, T. M. Khoshgoftaar, R. A. Bauder, and N. Seliya, “A survey on addressing high-class imbalance in big data,” *Journal of Big Data*, vol. 5, no. 1, p. 42, 2018.

[43] R. Caruana, S. Lawrence, and C. Giles, “Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping,” *Advances in Neural Information Processing Systems*, vol. 13, pp. 402–408, 2000.

[44] J. W. Cohen, “A coefficient of agreement for nominal scales,” *Educational and Psychological Measurement*, vol. 20, pp. 37–46, 1960.

[45] D. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *International Conference on Learning Representations*, 2014.

[46] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, “Pytorch: An imperative style, high-performance deep learning library,” *ArXiv*, vol. abs/1912.01703, 2019.