Automatic Sleep Stage Scoring based on Raw Single-Channel EEG using Transfer Learning

Nirali Parekh
Department of Computer Engineering
Dwarkadas J. Sanghvi College of Engineering
Mumbai, India
nirali25parekh@gmail.com

Bhavi Dave
Department of Computer Engineering
Dwarkadas J. Sanghvi College of Engineering
Mumbai, India
bhavidave5@gmail.com

Raj Shah
Department of Computer Engineering
Dwarkadas J. Sanghvi College of Engineering
Mumbai, India
rajshah2320@gmail.com

Kriti Srivastava
Department of Computer Engineering)
Dwarkadas J. Sanghvi College of Engineering
Mumbai, India
kriti.srivastava@djsce.ac.in

***Abstract*—Medicine has long reached an overwhelming consensus on the importance of sleep in maintaining mental and physiological homeostasis, and the link that sleep disruption has with both disease and mortality. With the advent of the domain of HealthTech, Deep Learning approaches have generated State Of The Art performance in solving several problems in the medicinal arena. The study of sleep- Polysomnography- uses Electroencephalogram (EEG) readings, among other parameters, to gain a clearer picture of a patient's sleep patterns since different brain activities correspond to different stages of sleep. Monitoring and interpreting EEG signals and the body's reactions to the changes in these cycles can help identify disruptions in sleep patterns. Successfully classified sleep patterns can in turn help medical professionals with the prognosis of several pervasive sleep related diseases like Sleep Apnea and the predilection for seizures. To address the pitfalls associated with the traditional manual review of EEG signals that help classify sleep stages, we have trained and analysed the performance of several Convolutional Neural Networks that classify the five sleep stages (Wake, N1, N2, N3, N4 and REM) using data from raw, single channel EEG signals. With the dataset being constant, a comparative analysis of the performance of popular convolutional neural network architectures can serve as a benchmark to the problem of sleep stage classification using EEG signals. We have implemented a CNN to extract time-invariant features and learn stage transition rules among sleep stages from EEG epochs. The models are promising, with some successfully classifying the five stages with a 97% accuracy as well as an f1 score of 0.98.**

***Index Terms*—Sleep stage scoring, EEG analysis, Deep Learning, Transfer Learning, Convolutional Neural Network**

I. Background and Motivation

Sleep plays a principal role in mental and physiological homeostasis. Sleep related disorders like insomnia and sleep apnea reduce the quality of life for the multitude of people who are affected. As connections between sleep disruption and both disease and mortality have become more firmly established in [1], accurate and efficient diagnosis and management of sleep disorders have become increasingly critical. Observing sleep cycles, along with the body's reactions to the changes in these cycles, can help identify disruptions in sleep patterns. Typically, all-night polysomnographic (PSG) recording consisting of an electrooculogram (EOG), electroencephalogram (EEG) and electromyogram (EMG) are analyzed for determining the quality of sleep. This PSG is divided into segments of 30-second recordings, known as epochs, which are then be classified into different sleep stages by the experts by following the guidelines such as the Rechtschaffen and Kales (R & K) [2] and the American Academy of Sleep Medicine (AASM) [3]. This process is known as sleep stage scoring or sleep stage classification. In the AASM classification, there is a wake stage, referred to as stage W, then three Non-REM sleep stages named as N1, N2, and N3, with N3 reflecting slow wave sleep and one REM sleep stage, referred to as stage R. This procedure of sleep scoring is, however, manual, hence labor-intensive and time-consuming [4]. Experts must visually inspect all epochs and label their sleep stages of entire PSG recordings. Thus, automatic sleep scoring for healthcare and well-being is in high demand. Artificial intelligence (AI) is already delivering on making aspects of health care more efficient, with the potential to support clinical and other applications that result in more insightful and effective care and operations. Deep Learning's efficiency in handling large amounts of data and the power to learn hidden features automatically is what makes it popular in numerous domains, and has hence received considerable attention from the sleep research community. Deep Learning can aid physicians at hospitals and health systems in sleep scoring by providing them with real-time, automatic classification that they can alter and monitor based on their personal expertise. Many sleep scoring methods using Deep Learning for automatic classification of PSG data have been proposed. Some works have been discussed in the section Lit review

TABLE I: Literature Review Analysis

Ref	Dataset Used	Architecture	Accuracy
[5]	MASS	SeqSleepNet - Hierarchical RNN	87.1%
[6]	Sleep-EDF	Transfer Learning Using CNN	84.3%
[7]	Sleep-EDF, MASS and SHHS	IITNet - transfer learning + bidirectional LSTM	Sleep-EDF: 83.9% MASS: 86.5% SHHS: 86.7%
[8]	Records from Massachusetts General Hospital Sleep Laboratory	SleepNet - RNN	85.76%
[9]	MASS	SVM	79.7%
[10]	Sleep-EDF and MASS	DeepSleepNet - CNN	MASS: 86.2%, Sleep-EDF: 82.0%
[11]	Sleep-EDF	Decision Tree	89.06%
[12]	Sleep-EDF	CNN + DNN with time-frequency image features	82.6%
[13]	Sleep-EDF	CNN + Temporal CNN + Conditional Random Field Layer	85.39%
[14]	Sleep-EDF	SleepEEGNet - CNN	84.26%
[15]	Records from Charite Clinic in Berlin	CNN + LSTM	40%
[16]	Sleep-EDF and MASS	CNN	Sleep-EDF: 82.3% MASS: 83.6 %
[17]	Sleep-EDF	bidirectional RNNs with attention + SVM	82.5%

II. Literature Review

A. Review of the Existing Work

Various review works have been focussed on the use of EEG waves in the problem of sleep scoring techniques. In [18], the authors have reviewed 154 papers that apply DL to EEG, published between January 2010 and July 2018, and spanning different application domains such as epilepsy, sleep, brain–computer interfacing, and cognitive and affective monitoring. The review also provides detailed methodological information on the various components of a DL-EEG pipeline to inform their own implementation. In another review [19], the authors has provided a comprehensive review of automated sleep stage scoring systems, since the year 2000. They analyse the system that were developed for Electrocardiogram (ECG), Electroencephalogram (EEG), Electrooculogram (EOG), and a combination of signals.

[9] proposes a Mixed Neural Network (MNN) that simultaneously aims to target the issues of population heterogeneity and temporal pattern recognition. Their novel architecture is composed of a rectifier neural network which is suitable for detecting naturally sparse patterns and a long short-term memory (LSTM) for the detection of temporally sequential patterns. Their model achieved a promising accuracy of 85.92%, outperforming SVM, RF and MLP in a comparative analysis.

[10] uses CNNs to extract time invariant features, and bidirectional-Long Short-Term Memory to learn transition rules among sleep stages to classify raw single-channel EEG using a two step training process. The training step one involves pre-training the model with an oversampled dataset to alleviate class-imbalance problems, proceed by step two that fine-tunes the model with EEG epochs that encode the temporal information into the model. Even without using any hard engineered features , DeepSleepNet archives an accuracy of 86.2% and 82.0% on the MASS and Sleep-EDF datasets respectively.

In [8], authors have proposed a deployed annotation tool for sleep staging. SleepNet uses a deep recurrent neural network trained on the largest sleep physiology database assembled to date, consisting of PSGs from over 10,000 patients from the Massachusetts General Hospital (MGH) Sleep Laboratory.The best performing instance of SleepNet uses expert-defined features to represent each 30-sec interval and learns to annotate EEG using a recurrent neural network (RNN).Their model analyses and compares various algorithms like Logistic Regression, Tree Boosting, Multilayer Perceptron, CNN, RNN and RCNN.

The authors of [14] have proposed an automatic sleep stage annotation method called SleepEEGNet using a single-channel EEG signal . The authors have applied sequence of deep learning models - CNN for feature extraction, BiRNNs for capturing temporal inf and an attention network to let the model learn the most relevant parts of the input sequence while training .The authors used the synthetic minority over-

sampling technique (SMOTE) to tackle class imbalance problems.

In [15], the authors have presented the implementation of deep learning methods for sleep stage detection by using three signals: heartbeat signal, respiratory signal, and movement signal They employ two different neural networks for this classification problem: the convolutional neural network (CNN) and the long-short term memory network (LSTM). However their model performs poorly giving only an accuracy of 40% and F1 score of 37% in classification of the 5 stages of sleep.

[5] has proposed a hierarchical recurrent neural network. The author treated automatic sleep staging as a sequence-to-sequence classification problem to jointly classify a sequence of multiple epochs at once. They achieve an overall accuracy and F1 score of 87.1% and 83.3% respectively.

[6] presented a deep transfer learning approach to address the problem of insufficient data in many sleep studies and to improve automatic sleep staging performance on small cohorts. They adopted the MASS dataset as the source domain and three different sleep databases are used as the target domains.

In their paper [7], the authors considered the inter- and intra-epoch temporal contexts using raw single-channel EEG. They model a deep CNN based on a modified ResNet-50 extracts the sleep-related features and the RNN via two-layered BiLSTM learns the transition rules.Their results show that the proposed temporal context learning at both the intra- and inter-epoch levels is effective to classify the time-series inputs.

The authors of [11] proposes the development of an Automatic Sleep Stage Classification (ASSC) system for detecting sleep stages using simple statistical statistical features (called EnergySis and Maximum-Minimum Distance) that are applied to 10 s epochs of single-channel EEG signals. Their solution that used Decision Tree achieved an average classification sensitivity, specificity and accuracy of 89.06%, 98.61% and 93.13% on the PhysioNet Sleep European Data Format (EDF) Database. The work is most notable for its focus on making the design ideal for being implemented in an embedded device in a real world setting in real time.

[12] proposes an efficient CNN for sleep stage classification. The simplified CNN is capable of learning features at a multitude of temporal resolutions while capturing time shift-invariance property of EEG signals because of its 1-max pooling layer. The work also creates a novel method to discriminatively learn a frequency-domain filter bank with a deep neural network (DNN) to preprocess the time-frequency image features. The model achieves an accuracy of 82.6% on the popular Sleep-EDF dataset.

The authors of [16] used a novel CNNs framework for sleep stage classification that simultaneously determined the classification label of the current epoch and the neighbouring epoch's prediction in the contextual output. The work archives an overall accuracy of 82.3% and 83.6%, on the Sleep-EDF Expanded (Sleep-EDF) and the Montreal Archive of Sleep Studies (MASS) dataset respectively.

In [13] a framework for sleep stage classification from single-channel EEG using a CNN to initially extract features that could serve as an input to a Temporal Convolutional Neural Network (TCNN) is proposed. The TCNN would help extract the temporal features from the extracted features vector of the CNN to achieve better results. The framework achieved an accuracy of 85.39% on the sleep EDF dataset.

In their paper [17], the authors used deep bidirectional RNNs with attention for single-channel sleep stage classification. The network works as a feature extractor and encodes information of an input sequence into a high-level feature vector that is then given to an SVM for the classification task. The work achieved an accuracy of 82.5% on the Sleep-EDF dataset.

Table I summarizes the existing work discussed in this section.

B. Research Gaps

- The review clearly demonstrates that most research uses only single-channel EEG signals to make classifications, and does not utilize all the available signal data including Fpz and Pz. The paper hypotheses that these signals contain valuable information about the temporal and spatial features of the waves and using all the signals can boost the accuracy.
- The existing architectures are also extremely bulky, requiring numerous layers and heavy preprocessing that inevitably increases the time required for training. Transfer learning can be invaluable in such a scenario.

III. Methodology

A. Dataset

In this study, two cohorts from the Physionet Sleep-EDF extended dataset contributed in 2018 are used for experimentation. The dataset contains PSG records and their corresponding sleep stages labeled by human sleep experts. These adopted cohorts have diverging health conditions, i.e. healthy (Sleep-EDF-SC) vs. mild sleep difficulty (Sleep-EDF-ST) [20], [21].

- Sleep-EDF-SC: This is the Sleep Cassette (SC) subset of the Sleep-EDF Expanded dataset. consisting of 20 subjects aged 25-34. Two subsequent day-night PSG recordings were collected for each subject.
- Sleep-EDF-ST: This is the Sleep Telemetry (ST) subset of the Sleep-EDF Expanded dataset which was collected for studying the temazepam effects on sleep. The subset consists of 22 subjects aged 18-79 with mild difficulty falling asleep. The PSG signals recorded after placebo intake are made available. Manual annotation was done similar to the Sleep-EDF-SC subset.

Table II summarizes the demographic information of datasets, including gender distributions and age characteristics. In each of the cohorts, each of the 30-second PSG epoch were manually labelled into one of eight categories W, N1, N2, N3, N4, REM, MOVEMENT, UNKNOWN by sleep experts according to the R &K standard [2]. Similar to previous works [10], [12], [13], [16], N3 and N4 stages were merged into a

TABLE II: Demographic information of the cohorts of Sleep EDF dataset

Dataset	Avg. epochs	Category	No. of subjects	Age		
				mean	min	max
Sleep Cassette	2650	male	77	59.3	26	97
		female	41	58.5	25	101
		total	36	58.9	25	101
Sleep Telemetry	2453	male	7	35.71	20	60
		female	15	50.85	18	79
		total	22	40.18	18	79

single stage N3 and MOVEMENT and UNKNOWN categories were excluded to make it compliant to AASM standards [3]. To conduct our experiments, we adopted the Fpz-Cz EEG and Pz-Oz EEG channels in this study.

B. Data Exploration

Using the Visbrain tool [22], the visualizations of the sample waveforms of different stages of sleep in an epoch of 30 seconds are shown in Figure 1.

C. Preprocessing

The data set is given as EDF [23] file for each subject. One EDF file represents one PSG record. For the scope of this work, only two signals were needed from the original files: Fpz-Cz EEG, and Pz-Oz EEG signals. By using pyEDFlib library [24], the mentioned signals were extracted and processed. Signals from each subject were divided into time sequences of 3000 timesteps, which, with the frequency of 100 Hz, corresponds to 30 seconds epochs. The tabular dataset obtained for each record is shaped like a three dimensional vector as following: (n epochs, y timesteps, z features) where n samples are the number of rows i.e. epochs inside the data set, n timesteps are the number of timesteps in each sample, 3000 in this case and z features is the number of signals, which is 2 for our study i.e. Fpz-Cz and Pz-Oz. Each data sample that represents one epoch has 480 timesteps, and each timestep has three values, one for every signal.

The obtained dataset was highly unbalanced with a high number of instances i.e. epochs of Wake stage. Due to the uneven distribution of data among the sleep stages, undersampling has been performed to address the data-skewing problem. Near equal number of data points were sampled for the stages Wake, N1, N2, N3 and REM.

Since great results have been achieved by processing data with deep learning techniques like convolutional neural networks (CNN) with images as input. CNN's great performance for reading, processing and extracting the most important features of two dimensional data have highly contributed to its popularity. In scenarios where input data isn't formatted as an image, like time series, transformation methods have been applied to apply CNN models for our data. In this study, such transformations have been used to convert the EEG signal data to images. So, the sampled epochs obtained in the previous steps are first converted into signal images of grayscale images like the one shown in Figure 2. This preprocessing step helps to detect hidden features and topologies like homogeneity, periodicity, driftnes and dispersion when dealing with time series like EEG signal data.

Thus, our image data is ready to be fed into the proposed Deep Learning architectures for training the models.

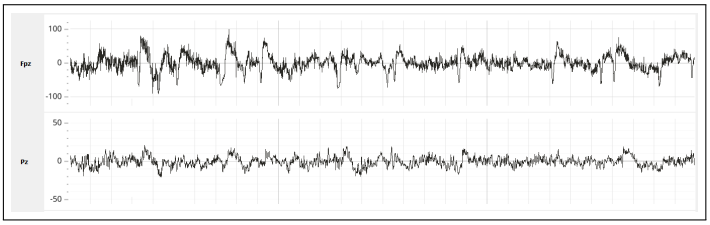
IV. Experimental Architectures

A convolutional neural network is a subtype of deep neural networks which uses alternating layers of convolutions and pooling with trainable filter banks (made up of kernels) per layer. Kernels scan over a receptive field to compute the output feature map with a dot product and bias computation. The result is passed through an activation function, and then subsampled using max pooling to reduce sensitivity to distortions of stimuli in the upper layers. CNNs generally employ the Rectified Linear Unit activation function that converts negative values to 0, extracting only the prominent features. The max pooling layer helps reduce complexity while allowing prominent features to pass through to upper layers of the network . This process is repeated to gain relevant, abstract and location invariant features that are accurate depictions for recognition problems. CNNs are location equivariant in the sense that they preserve the location of features, but extract the same features over the entire image. This manner of feature learning often produces state of the art results. Apart from dominating the domain of computer vision and 11 image analysis, CNNs are also apt for 1D problems like time series, and 3D image classification, because of their reliance on a method's ability to learn features. CNNs greatly reduce the number of parameters when compared to traditional feedforward networks because of their unique design consisting of weight sharing and pooling. Thus, because of sparse-connectivity and weight sharing, we are able to achieve a similar degree of complexity with a significantly smaller number of parameters with a Convolutional layer.

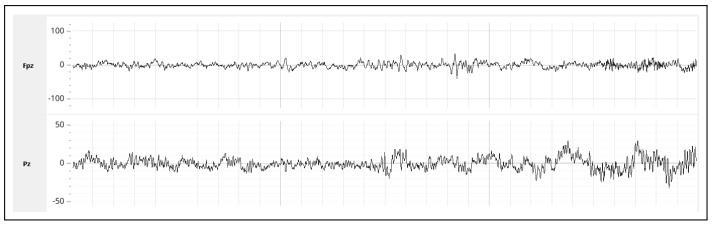
This paper performed experiments on several CNN architectures of varying complexity to analyse the effects of the different design features to the problem statement.The experiments were run with the pretrained models trained on the ImageNet dataset.The last layer to reflect the five sleep stages. With the dataset being constant, a comparative analysis of the performance of popular convolutional neural network architectures can serve as a benchmark to the problem of sleep stage classification using EEG signals. The architectures chosen vary in depth, parameters, design, complexity and emphasis. A comparative analysis of this performance can help give insight to the aspects that will be most relevant to an ideal solution. The following section provides an overview to the key features of the various CNNs trained on the EEG dataset.

A. AlexNet

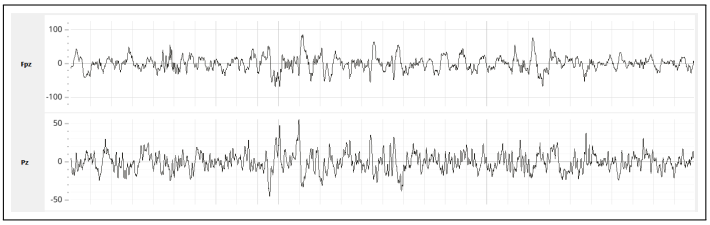
AlexNet [25] takes in colour images of dimensions 227x227 that are passed through 5 alternating convolution layers and 3 fully connected layers, with the last fully connected layer having neurons equivalent to the number of classes. The



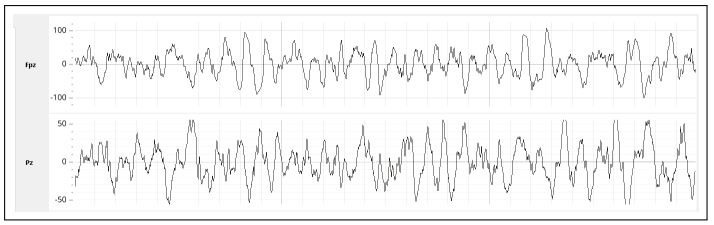
(a) Wake sleep stage



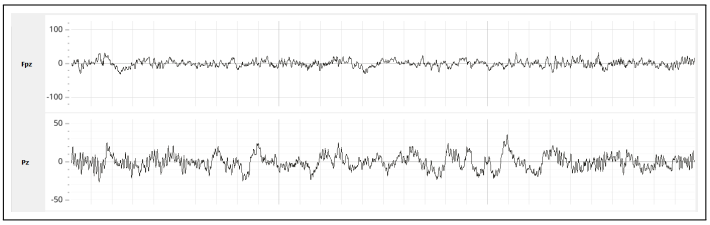
(b) N1 sleep stage



(c) N2 sleep stage

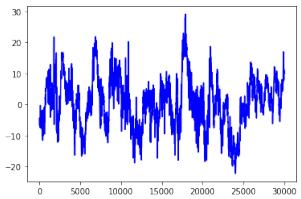


(d) N3 sleep stage

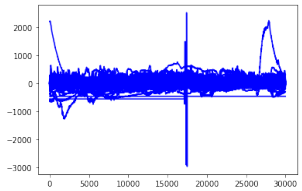


(e) REM sleep stage

Fig. 1: Raw EEG waveforms for each of the sleep stages



(a) Sleep Cassette sample



(b) Sleep Telemetry sample

Fig. 2: Samples of plotted signals of sleep stage 'REM' after raw data converted to graphs from Sleep Cassette and Sleep Telemetry cohorts

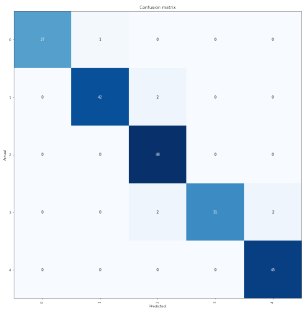
architecture employs max-pooling after convolution layer 1, convolution layer 2 and convolution layer 5. The count of the total number of layers in the architecture does not include the max-pooling layers as they do not carry weights. Thus, AlexNet has 8 layers. AlexNet was proposed in 2012 as a significant improvement on the CNN performance at the time and has been used as a benchmark since. Its key contribution was using the ReLU activation function instead of tanh or sigmoid to achieve faster training times.

B. VGG

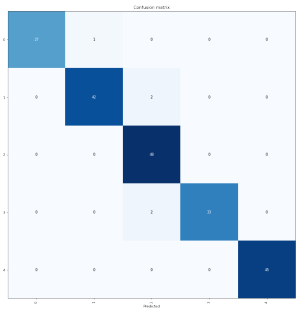
During the design of the VGGNet [26], it was found that alternating convolution & pooling layers were not required, and consequently VGG uses multiple Convolutional layers in sequence with pooling layers in between. Input of dimensions 224x224 is taken and a kernel size 3x3 is maintained throughout the network with only a change in depth between layers. Appropriate padding is provided to maintain the dimensions across the layers. The VGG architecture has several versions like the VGG-16 (a 16 layer network), VGG-19 (a 19 layer network) and so forth.VGG demonstrated a solution to the pertinent problem of increasing depth of a CNN - the small-size convolution filters in the architecture allowed VGG to have a large number of weight layers;that increased depth and improved performance.

C. ResNet

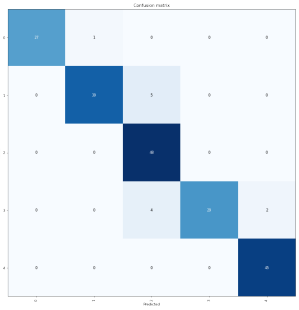
As the depth of the VGG network was increased, the model performance began to deteriorate. It was hypothesised that the gradients were not able to flow well through the deeper network. The scientists found that the information from the input was getting highly morphed as it reached the deeper output layers, and consequently proposed a solution of passing the input information repeatedly in stages creating a Residual Network or the ResNet [27]. In ResNet after every two layers the input given to the first layer along with the output obtained at the second layer. The information from the input that was getting highly morphed and by the time it reached the output layers is now passed as a residue of the input once again with the output. This helped the gradients to flow back better, improving training. ResNet's residual connections between



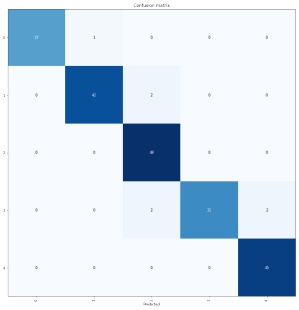
(a) ResNet50



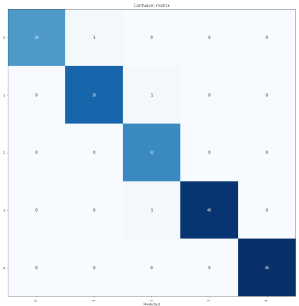
(b) MobileNetV2



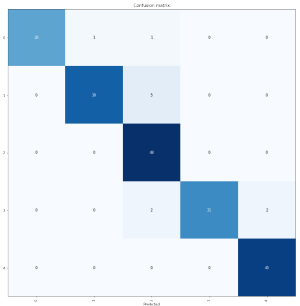
(c) AlexNet



(d) DenseNet121

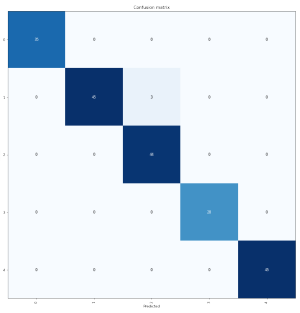


(e) VGG19

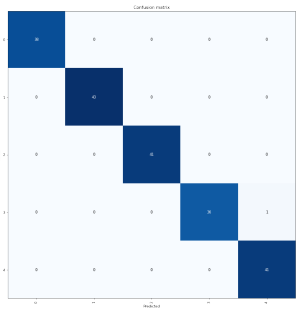


(f) SqueezeNet 1.1

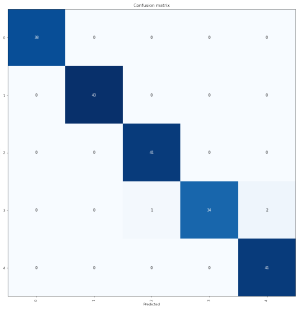
Fig. 3: Confusion Matrices for different models using Fpz-Cz channel of Sleep Cassette Cohort of Sleep-EDF dataset



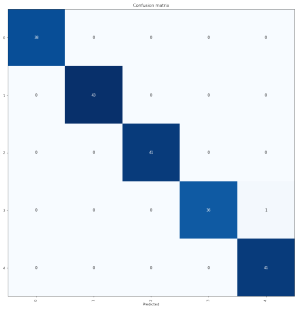
(a) ResNet50



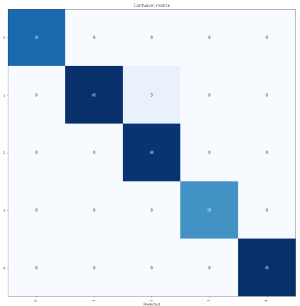
(b) MobileNetV2



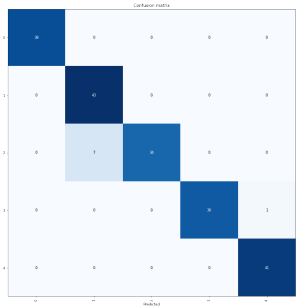
(c) AlexNet



(d) DenseNet121



(e) VGG19



(f) SqueezeNet 1.1

Fig. 4: Confusion Matrices for different models using Pz-Oz channel of Sleep Cassette Cohort of Sleep-EDF dataset

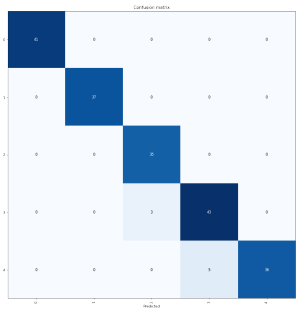
layers which were invaluable training models as deep as 151 layers.

D. DenseNet

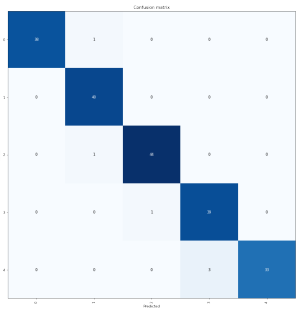
DenseNets [28] connect every layer to every other layer, implying that for L layers, there are L(L+1)/2 direct connections. Every layer uses the feature maps of all the preceding layers as inputs, and consequently its output feature maps are used as input for subsequent layers. The network is divided into densely connected blocks within which the feature map size remains the same. This facilitates both downsampling and feature concatenation. DenseNets require fewer parameters than a comparable size traditional CNN as the need to learn redundant feature maps is eliminated. It also facilitates the flow of information and gradient as each layer has direct access to the input and the gradient of the loss function. The architecture consists of four DenseBlocks with varying number of layers, with the convolution operations inside each of the architectures as the Bottle Neck layers. In other words, the 1×1 convolutions are introduced as a bottleneck layer before each 3×3 convolution to reduce the number of input feature-maps, consequently improving computational efficiency.

E. Squeezenet

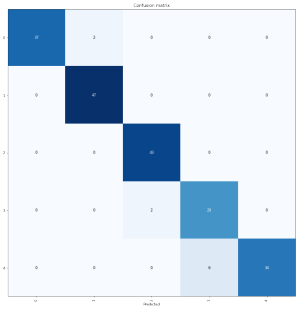
SqueezeNets [29] attempt to reduce the number of parameters in the network by using Fire modules which have a squeeze layer of 1x1 convolutions that can decrease parameters by restricting the number of input channels in every layer. SqueezeNets are therefore very low latency. They achieve AlexNet level accuracies while having 50 million fewer parameters. A Fire module consists of a squeeze convolution layer that feeds into an expand layer that has a mix of 1×1 and 3×3 convolution filters. SqueezeNet starts with a convolution layer, followed by 8 Fire modules, and concludes with a



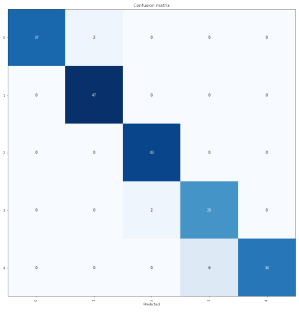
(a) ResNet50



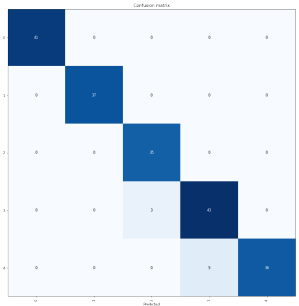
(b) MobileNetV2



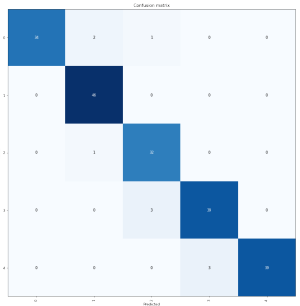
(c) AlexNet



(d) DenseNet121

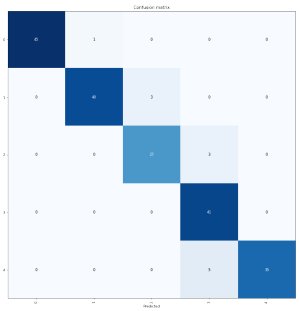


(e) VGG19

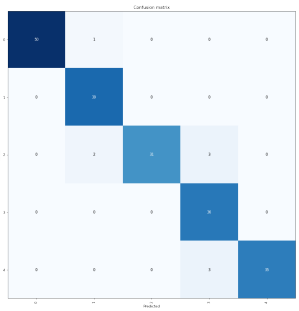


(f) SqueezeNet 1.1

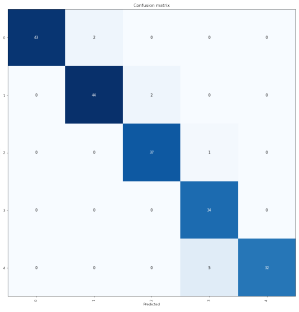
Fig. 5: Confusion Matrices for different models using Fpz-Cz channel of Sleep Telemetry Cohort of Sleep-EDF dataset



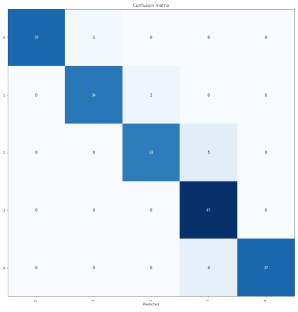
(a) ResNet50



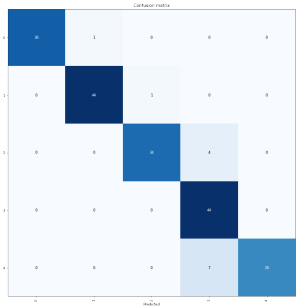
(b) MobileNetV2



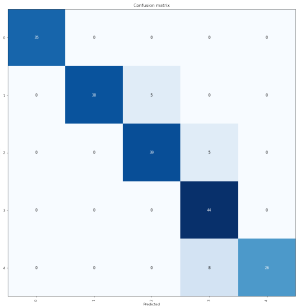
(c) AlexNet



(d) DenseNet121



(e) VGG19



(f) SqueezeNet 1.1

Fig. 6: Confusion Matrices for different models using Pz-Oz channel of Sleep Telemetry Cohort of Sleep-EDF dataset

final convolution layer. The number of filters per fire module decreases with depth. Inspired by ResNet, the architecture also employs bypass.

F. MobileNet

MobileNets [30] utilise Depth convolutions and point convolutions, reducing the comparison and recognition time. They are designed to increase accuracy while simultaneously factoring the resource constrained environments of embedded devices or mobile phones, therefore getting their name. They also reduce the number of parameters and hence latency. To construct an even smaller and computationally cheap model, MobileNets also have useful model-shrinking parameters that help make the network thinner in a uniform manner at every layer.The architecture helps explore a reasonable accuracy, latency and size trade off. MobileNet has 28 layers if the depthwise and pointwise convolutions are considered to be distinct. All layers in mobilenet are built on depth wise separable convolutions followed by batch normalization and ReLU non-linearity, except for the first layer that is a fully convolutional layer.

V. Results and Discussions

A. Evaluation measures

After training, the model has been evaluated on the test set that it has never seen before. The model was trained on 80% of the data and tested on 20% of the complete data set.

1. Accuracy: It is the fraction of classifications the model got correct.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

The accuracy is a widespread measurement for evaluation but is not always reliable. In cases of imbalanced classes where

TABLE III: Result metrics of various experimental architectures

Dataset	Channel	Metric	Model					
			ResNet50	MobileNetV2	AlexNet	DenseNet121	VGG19	SqueezeNet 1.1
Sleep-EDF Sleep Cassette (SC)	Fpz-Cz	Accuracy	0.96	0.97	0.94	0.96	0.98	0.94
		Precision	0.97	0.98	0.95	0.97	0.98	0.96
		Recall	0.96	0.97	0.94	0.96	0.98	0.94
		F1-score	0.96	0.98	0.94	0.96	0.98	0.95
	Pz-Oz	Accuracy	0.98	0.99	0.98	0.97	0.98	0.96
		Precision	0.99	1.00	0.99	0.97	$0.9^{\cdots}$	0.97
		Recall	0.98	0.99	0.98	0.98	0.99	0.96
		F1-score	0.99	0.99	0.98	0.97	0.99	0.96
Sleep-EDF Sleep Telemetry (ST)	Fpz-Cz	Accuracy	0.96	0.97	0.95	0.95	0.96	0.95
		Precision	0.96	0.98	0.95	0.95	0.95	0.95
		Recall	0.95	0.97	0.95	0.96	0.96	0.95
		F1-score	0.95	0.97	0.95	0.95	0.95	0.95
	Pz-Oz	Accuracy	0.94	0.95	0.95	0.94	0.94	0.91
		Precision	0.94	0.96	0.95	0.95	0.95	0.93
		Recall	0.94	0.95	0.95	0.94	0.93	0.91
		F1-score	0.94	0.95	0.95	0.94	0.94	0.91

one class is significantly more represented, the model will give high accuracy by classifying every sample into that class. That is why precision and recall are introduced because they are giving a better insight into classification performance.

2. Precision: Proportion of the positive identifications which are correct.

$$\text{Precision } = \frac{TP}{TP + FP} \tag{2}$$

3. Recall: Proportion of actual positives which were identified correctly.

$$\text{Recall } = \frac{\tau P}{TP + FN} \tag{3}$$

F1 score should be taken as a reliable metric as it is not influenced by bias in the number of samples per class.

4. F1-score: It is a weighted average of Precision and Recall. It is calculated as the harmonic mean of precision and recall.

$$F1 - \text{ score } = \frac{2 \times \text{ precision } \times \text{ recall}}{\text{precision } + \text{ recall}} \tag{4}$$

B. Results

The models were used to classify the data into 5 classes - the Wake stage, N1 stage, N2 stage, N3 stage and REM stage. The results are shown in Table III, which shows previously described measures. All measurements were calculated on test data. For every model and for each classification, learning accuracy reached around 94-95%. Comparing the models, for the Sleep EDF -SC cohort, the best performing model for Fpz-Cz is VGG-19 and for Pz-Oz, it is MobileNet-v2. Similarly, for the Sleep EDF- ST cohort, MobileNet-v2 shows the best performance for Fpz-Cz and Pz-Oz channels both.

Figure 3 and Figure 4 presents the confusion matrices achieved by each model using the Fpz-Cz and Pz-Oz channels respectively of the Sleep-EDF- Sleep Cassette Cohort. Similarly,Figure 5 and Figure 6 shows the confusion matrices using the Fpz and Pz-Oz channels respectively of the Sleep EDF Sleep Telemetry cohort. The main diagonals in each confusion matrix denote the true positive (TP) values which indicate the number of stages scored correctly. It can be seen from the tables (the confusion matrices' parts) that TP values are significantly higher than other values in the same rows and columns, indicating notable performance of the models.

VI. Conclusion

The work has implemented and automated the sleep staging pipeline- when given raw, single channel EEG data signals of a patient from the Fpz-Cz of the Sleep-EDF data sets, we classify the five sleep stages (Wake, N1, N2, N3, N4 and REM) using several different Convolutional Neural Network architectures. The preprocessing and sampling implemented helps avoid model bias, ensuring that the models evaluate sleep scoring according to the new AASM standards. All models achieve an accuracy ¿95% and an f1 score of ¿0.93 without utilizing any hand-engineered features.¡Best Model¿

The pipeline utilizes CNNs to extract time-invariant features and learn stage transition rules among sleep stages from EEG epochs. The results also showed that the temporal information learned from the sequence residual learning part helped improve the classification performance. The experiments clearly demonstrated that the models could learn features for sleep stage scoring from different raw single-channel EEGs. The paper also provides an exhaustive overview of the existing literature in the domain, allowing researchers to review the

possible approaches of solving the problem. The work also benchmarked the performance of several popular CNN architectures on a consistent sleep stage classification dataset. ¡¿ The success of EEGTransNet can serve as a step on tangible progress not just in the domain of sleep stage classification in particular but also aid the process of integrating AI in medicine.

REFERENCES

[1] S. L. Worley, "The extraordinary importance of sleep: the detrimental effects of inadequate sleep on health and public safety drive an explosion of sleep research," *Pharmacy and Therapeutics*, vol. 43, no. 12, p. 758, 2018.

[2] A. Rechtschaffen, "A manual for standardized terminology, techniques and scoring system for sleep stages in human subjects," *Brain information service*, 1968.

[3] C. Iber, "The aasm manual for the scoring of sleep and associated events: Rules," *Terminology and Technical Specification*, 2007.

[4] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[5] H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. De Vos, "Seqsleepnet: end-to-end hierarchical recurrent neural network for sequence-to-sequence automatic sleep staging," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 3, pp. 400–410, 2019.

[6] H. Phan, O. Y. Chén, P. Koch, Z. Lu, I. McLoughlin, A. Mertins, and M. De Vos, "Towards more accurate automatic sleep staging via deep transfer learning," *IEEE Transactions on Biomedical Engineering*, vol. 68, no. 6, pp. 1787–1798, 2020.

[7] H. Seo, S. Back, S. Lee, D. Park, T. Kim, and K. Lee, "Intra-and inter-epoch temporal context network (iitnet) using sub-epoch features for automatic sleep scoring on raw single-channel eeg," *Biomedical Signal Processing and Control*, vol. 61, p. 102037, 2020.

[8] S. Biswal, J. Kulas, H. Sun, B. Goparaju, M. B. Westover, M. T. Bianchi, and J. Sun, "Sleepnet: automated sleep staging system via deep learning," *arXiv preprint arXiv:1707.08262*, 2017.

[9] H. Dong, A. Supratak, W. Pan, C. Wu, P. M. Matthews, and Y. Guo, "Mixed neural network approach for temporal sleep stage classification," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 2, pp. 324–333, 2017.

[10] 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.

[11] K. A. Aboalayon and M. Faezipour, "Real time sleep detection system using new statistical features of the single eeg channel," 2017.

[12] H. Phan, F. Andreotti, N. Cooray, O. Y. Chén, and M. De Vos, "Dnn filter bank improves 1-max pooling cnn for single-channel eeg automatic sleep stage classification," in *2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*. IEEE, 2018, pp. 453–456.

[13] E. Khalili and B. M. Asl, "Automatic sleep stage classification using temporal convolutional neural network and new data augmentation technique from raw single-channel eeg," *Computer Methods and Programs in Biomedicine*, vol. 204, p. 106063, 2021.

[14] S. Mousavi, F. Afghah, and U. R. Acharya, "Sleepeegnet: Automated sleep stage scoring with sequence to sequence deep learning approach," *PloS one*, vol. 14, no. 5, p. e0216456, 2019.

[15] K. Stuburić, M. Gaiduk, and R. Seepold, "A deep learning approach to detect sleep stages," *Procedia Computer Science*, vol. 176, pp. 2764–2772, 2020.

[16] 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.

[17] H. Phan, F. Andreotti, N. Cooray, O. Y. Chen, and M. De Vos, "Automatic sleep stage classification using single-channel eeg: Learning sequential features with attention-based recurrent neural networks," in *2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*. IEEE, 2018, pp. 1452–1455.

[18] Y. Roy, H. Banville, I. Albuquerque, A. Gramfort, T. H. Falk, and J. Faubert, "Deep learning-based electroencephalography analysis: a systematic review," *Journal of Neural Engineering*, vol. 16, no. 5, p. 051001, aug 2019. [Online]. Available: https://doi.org/10.1088/1741-2552/ab260c

[19] O. Faust, H. Razaghi, R. Barika, E. J. Ciaccio, and U. R. Acharya, "A review of automated sleep stage scoring based on physiological signals for the new millennia," *Computer methods and programs in biomedicine*, vol. 176, pp. 81–91, 2019.

[20] B. Kemp, A. H. Zwinderman, B. Tuk, H. A. Kamphuisen, and J. J. 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.

[21] A. L. Goldberger, L. A. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[22] E. Combrisson, R. Vallat, J.-B. Eichenlaub, C. O'Reilly, T. Lajnef, A. Guillot, P. M. Ruby, and K. Jerbi, "Sleep: an open-source python software for visualization, analysis, and staging of sleep data," *Frontiers in neuroinformatics*, vol. 11, p. 60, 2017.

[23] B. Kemp, A. Varri, A. C. Rosa, K. D. Nielsen, and J. Gade, "A simple format for exchange of digitized polygraphic recordings," *Electroencephalography and clinical neurophysiology*, vol. 82, no. 5, pp. 391–393, 1992.

[24] H. Nahrstaedt, "pyedflib," https://github.com/holgern/pyedflib, 2020, accessed July 10, 2021.

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015. [Online]. Available: http://arxiv.org/abs/1512.03385

[28] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016. [Online]. Available: http://arxiv.org/abs/1608.06993

[29] F. N. Iandola, M. W. Moskewicz, K. Ashraf, S. Han, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and <1mb model size," *CoRR*, vol. abs/1602.07360, 2016. [Online]. Available: http://arxiv.org/abs/1602.07360

[30] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *CoRR*, vol. abs/1704.04861, 2017. [Online]. Available: http://arxiv.org/abs/1704.04861