UNIVERSITY OF ICELAND

Deep Neural Networks for Seizure Detection: A Study on Training Strategies and Architectural Designs

Davíð Hringur Ágústsson

May 2023

M.Sc. thesis
in Computational Engineering

FACULTY OF INDUSTRIAL ENGINEERING, MECHANICAL ENGINEERING AND COMPUTER SCIENCE

Deep Neural Networks for Seizure Detection: A Study on Training Strategies and Architectural Designs

Davíð Hringur Ágústsson

60 ECTS thesis submitted in partial fulfillment of a
Magister Scientiarum degree in Computational Engineering

Supervisor
Steinn Guðmundsson

M.Sc. Committee
Hafsteinn Einarsson
Steinn Guðmundsson

Examiner
Magnús Örn Úlfarsson

Faculty of Industrial Engineering, Mechanical Engineering and Computer Science
School of Engineering and Natural Sciences
University of Iceland
Reykjavik, May 2023

Deep Neural Networks for Seizure Detection: A Study on Training Strategies and Architectural Designs

60 ECTS thesis submitted in partial fulfillment of a M.Sc. degree in Computational Engineering

Faculty of Industrial Engineering, Mechanical Engineering and Computer Science
School of Engineering and Natural Sciences
University of Iceland
Dunhagi 5
107, Reykjavik Iceland

Telephone: 525 4000

Bibliographic information:
Davíð Hringur Ágústsson (2023) *Deep Neural Networks for Seizure Detection: A Study on Training Strategies and Architectural Designs*, M.Sc. thesis, Faculty of Industrial Engineering, Mechanical Engineering and Computer Science, University of Iceland.

Reykjavik, Iceland, May 2023

Abstract

Epilepsy is a neurological disorder affecting over 50 million people globally. There is significant need for efficient automatic seizure detection algorithms for timely intervention and accurate diagnosis. This study examines the potential of deep neural networks for seizure detection from electroencephalography (EEG) signals, integrating architectural designs and training techniques from other domains. Our results suggest a limit to the benefits of increased architectural complexity when training on the TUH EEG seizure corpus without augmentations. Training strategies such as mixup, segment translation and ensembling led to substantial improvements in model performance over baseline, particularly when considered in conjunction with suitable model architectures. These methods enhanced both the generalization performance and the calibration of the models. These findings underscore the importance of a balanced approach in designing seizure detection models by considering network architecture and training methods simultaneously. integrating architectural designs and training techniques from other domains. Our results suggest a limit to the benefits of increased architectural complexity when training on the TUH EEG seizure corpus without augmentations. Training strategies such as mixup, segment translation and ensembling training strategies led to substantial improvements in model performance over baseline, particularly when considered in conjunction with suitable model architectures. These methods enhanced both the generalization performance and the calibration of the models. These findings underscore the importance of a balanced approach in designing seizure detection models by considering network architecture and training methods simultaneously.

Útdráttur

Flogaveiki er taugasjúkdómur sem hrjáir um 50 milljónir manna á heimsvísu. Þörf er á sjálfvirkum aðferðum við greiningu á flogaköstum til að bæta greiningu og meðhöndlun á sjúkdómnum. Í þessu verkefni voru djúp tauganet notuð til að greina flogaköst út frá heilaritum (EEG). Áhersla var lögð á að kanna áhrif þjálfunaraðferða og netarkitektúrs sem reynst hafa vel í skyldum verkefnum. Niðurstöður benda til þess að netarkitektúr einn og sér hafi minna að segja um nákvæmni líkana en þjálfunaraðferðir. Þjálfunaraðferðir á borð við mixup, tímahliðrun og safnaðferðir gáfu umtalsverða bætingu á frammistöðu, en bætingin var einnig háð netarkitektúr. Bætingin fólst bæði í aukinni greiningarnákvæmni og kvörðun á líkönum. Niðurstöður verkefnisins sýna mikilvægi þess að skoða netarkitektúr og þjálfunaraðferðir samtímis þegar verið er að hanna greiningarlíkön fyrir flogaveiki.

Contents

List of Figures

List of Tables

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor, Prof. Steinn Guðmundsson, for his guidance, encouragement, and patience throughout the course of my research. His valuable feedback and insights have been invaluable in shaping this work.

I would like to extend my sincere thanks to Ana Borovac, a PhD student, for her generous help and assistance. Her support and collaboration have been instrumental in the completion of this research.

My heartfelt thanks go to my brother, Eiríkur Þór Ágústsson, for his help and valuable input. His unwavering support and insightful perspectives have contributed significantly to this work.

I would also like to acknowledge the Icelandic High Performance Computing Centre (IHPC) for granting access to their HPC cluster. The computational resources provided were essential in conducting the experiments in this study.

On a personal note, I would like to express my utmost appreciation to my parents, Guðbjörg Þórisdóttir and Ágúst Þór Eiríksson. Their continuous love, support, and belief in me have been my strength and motivation throughout my academic journey.

Finally, I would like to express my gratitude to everyone who has contributed to this research in one way or another. Your support has made this journey a rewarding one, and I hope this thesis stands as a testament to your invaluable contributions.

1. Introduction

Epilepsy is a neurological disorder characterized by a predisposition to recurrent, unprovoked seizures, affecting over 50 million people worldwide [1]. Epilepsy can have various causes, such as genetic factors, brain injuries, or developmental disorders, and is typically diagnosed when a person has had two or more unprovoked seizures separated by at least 24 hours [2]. Detecting these seizures is crucial for prescribing appropriate treatment [3], [4]. Various methods for detecting seizures exist, with electroencephalography (EEG) being the most common. Continuous EEG monitoring of critically ill patients is frequently used to detect seizures in intensive care units [5]. EEG is a measurement of the brain's electrical activity through electrodes placed on the scalp. The EEG is then analyzed for abnormal patterns indicative of seizures [6]. Generally, in EEG interpretation, seizure-like patterns must be continuously present for at least 10 seconds to be typically classified as a seizure [7].

An accurate automatic seizure detection algorithm (SDA) is very useful during EEG recordings to ensure timely intervention in the event of a seizure. While skilled neurologists can manually identify characteristic seizure patterns, this process is time-consuming and labor-intensive, particularly in case of long-term EEG recordings which can span several days. An accurate detection system can significantly aid physicians in diagnosis and prescription. Currently, the most SDAs employed for practical applications demonstrate sub-optimal performance [8]. Several research organizations have released public datasets to facilitate the development of efficient SDAs [9]–[12].

Various approaches to developing SDAs have been explored. Feature-based methods have long been the foundation of seizure detection algorithms, involving significant effort in hand-crafting features to identify seizure patterns in EEG data [13]. These methods extract specific characteristics or numerical quantities, such as time-domain and frequency-domain features, from the EEG signals. Time-domain features may include statistical measures like mean, variance, and skewness, while frequency-domain features encompass spectral power and spectral entropy. These extracted features, in vector form, are then fed into standard pattern classifiers such as support vector machines, decision trees, or k-nearest neighbors to classify the data.

In contrast, deep learning algorithms, such as Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs), have gained attention as an alternative to feature-based methods in recent years. These algorithms are designed to automatically learn relevant features from the data. CNNs are particularly suited for seizure detection, as they can capture local temporal and/or spatial information and hierarchical patterns in the EEG

data through the use of convolutional layers and pooling operations [14], [15]. RNNs are also capable of modeling the temporal dependencies in EEG signals, which allows them to capture the dynamic nature of seizure events [16], [17].

The adoption of deep learning algorithms for seizure detection brings about several advantages over traditional feature-based methods. Firstly, deep learning models can automatically learn and extract relevant features from raw EEG data without relying on expert knowledge, reducing the need for manual feature engineering. This can potentially lead to models which are more robust and have better generalisation abilities. Secondly, deep learning models can potentially capture more complex relationships in the data, which might be difficult or impossible for feature-based methods to model. Finally, deep learning models can be easily scaled to handle large amounts of data, making them suitable for processing the vast amounts of EEG data generated in clinical settings.

In addition to feature-based and deep learning methods, other approaches to seizure detection include template matching and statistical or model-based strategies. Template matching identifies pre-defined seizure patterns [18], while statistical and model-based methods analyze deviations in the data from expected pattern [19].

Research in the field also addresses challenges to improve various aspects of seizure detection and analysis. One challenge involves classifying different seizure types, such as generalized seizures, partial (or focal) seizures, absence seizures, myoclonic seizures, atonic seizures, tonic seizures, and clonic seizures. Distinct treatments and prescriptions may be necessary for each type [20], [21]. Another problem researchers tackle is the handling of artifacts from EEG recordings, caused by factors such as muscle and eye movements or the motion of electrodes and cables. Artifact removal is useful in EEG analysis, as these unwanted signals can distort the true underlying brain activity and lead to inaccurate seizure detection [22].

Calibration of SDAs has also been of interest recently, focusing on producing meaningful probabilities when classifying seizures. Ideally, when the classifier predicts correctly, the probability estimate returned by the classifier is high, and when seizures are incorrectly classified, probability estimates should be lower. Often, this is not the case. Research by Guo et al.[23] indicates that DNNs, when trained on tasks of image and document classification, often result in overconfident predictions, even when their classification accuracy is high. This observation was further corroborated for image classifiers in [24]. In a clinical setting, calibration helps clinicians make informed decisions based on the certainty of a seizure occurrence [25]. A well-calibrated model not only improves the interpretability of the SDA but can also enable more efficient labeling of seizures, with a trained neurologists focusing on labeling segments for which the detector is uncertain. The classifier could then be further trained on those segments, continually improving the classifier.

The objective of this study was to create a highly accurate deep neural network (DNN)-based SDA and study how calibration is effected between networks and training metods. We aim to address the following research questions:

1. To what extent can architectural designs from other domains be used for EEG?
2. To what extent can training methods from other domains improve performance?
3. Are the probability estimates returned by the SDAs clinically useful?

We will explore different architectural designs and training techniques that can potentially enhance the performance of SDAs. We will also study how well the DNNs are calibrated and investigate methods that are known to improve model calibration in other settings.

By investigating these aspects, we hope to contribute to the development of more accurate and reliable SDAs for seizure detection.

2. Methods

2.1. Data and preprocessing

In this study, we used the TUH EEG seizure corpus version 2.0.0 [9] as our source of EEG data. Compared to other publicly available datasets, the TUH EEG seizure corpus is characterized by a wide range of recording setups, a relatively large number of patients and includes a variety of seizure types, with focal non-specific seizures being the most common. We only include recordings using the Averaged Reference (AR) montage to ensure that all data has the same recording setting. The EEG recordings are stored in the EDF format (European Data Format), while the annotations (labels) are stored in text files and contain the start and stop timestamps for each seizure event.

The recordings were annotated by human experts using a bipolar temporal central parasagittal montage. In our study, we adopted this same montage by converting the Averaged Reference montage to a bipolar temporal central parasagittal montage. We used all the signals derived from this montage as inputs to the detectors in our experiments, that is, channels Fp1-F7, F7-T3, T3-T5, T5-O1, Fp2-F8, F8-T4, T4-T6, T6-O2, T3-C3, C3-Cz, Cz-C4, C4-T4, Fp1-F3, F3-C3, C3-P3, P3-O1, Fp2-F4, F4-C4, C4-P4, P4-O2, A1-T3 and T4-A2. The dataset comes with predefined training, validation, and test sets. Different patients are assigned to each set, ensuring unbiased testing.

To preprocess the EEG signals, a Butterworth bandpass filter with cut-off frequencies of 0.5 Hz and 30 Hz was employed. The signals were then downsampled to 62 Hz. Following filtering and downsampling, each recording was segmented into 16-second-long segments which are used as inputs to the detectors (figure 2.1). This is analogous to the setup used by Borovac et al. in [25]. We did not include segments that contain the onset or offset of a seizure, only segments containing exclusively seizure or non-seizure activity, both for training and testing.

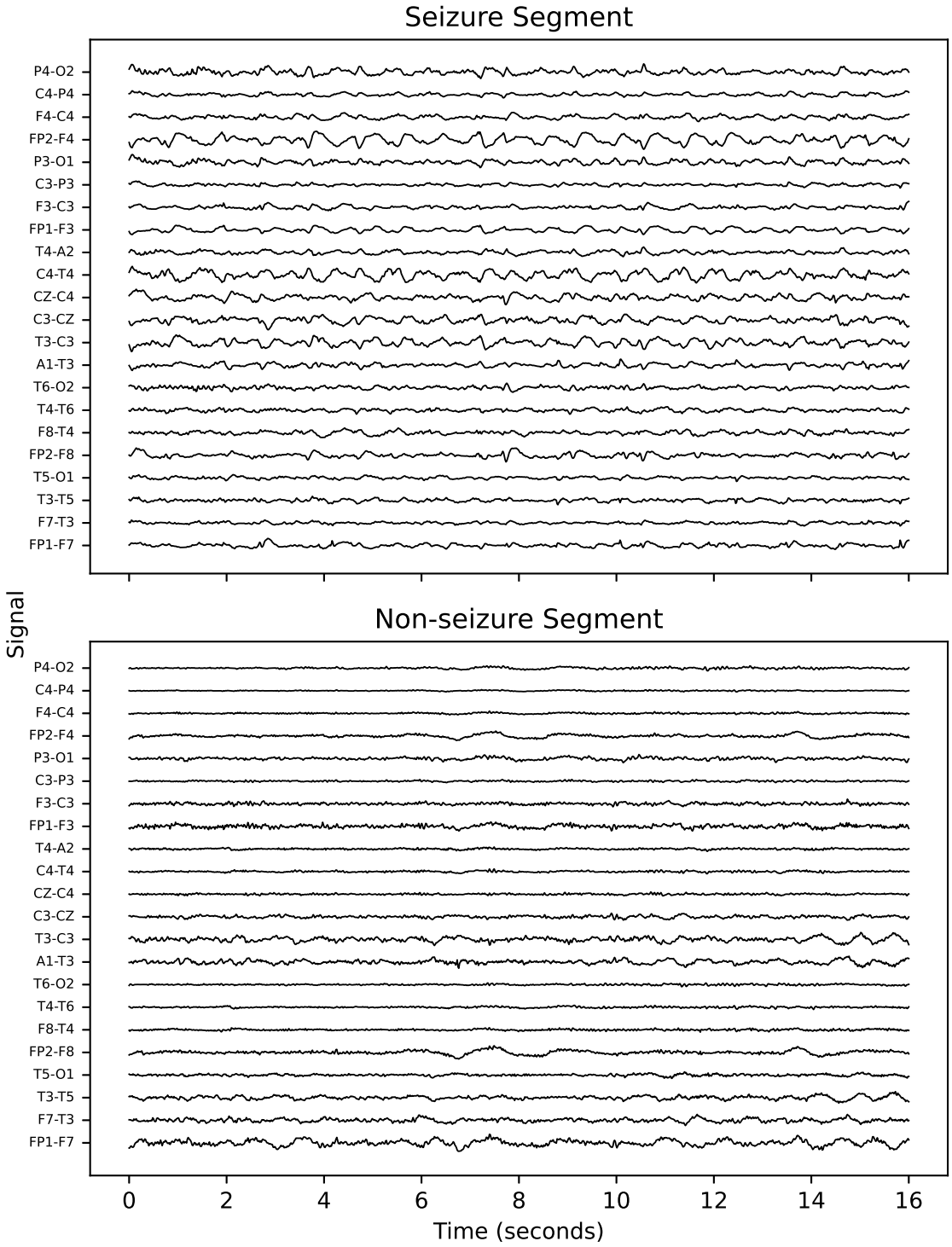


Figure 2.1: Examples of seizure and non-seizure samples from the test set.

	Training	Validation (patients with seizures)	Test (patients with seizures)
Number of patients	281	28	32
Total duration of recordings [hours]	570.61	331.26	94.45
Total duration of seizures [hours]	25.58	11.52	7.23
Fraction of seizure activity [%]	4.48	3.47	7.65
Average duration of recordings	2.03	11.83	2.95
Average duration of seizures	0.09	0.41	0.23
Number of seizure samples	19643	2324	1477
Number of non-seizure samples	120681	71070	19109

Table 2.1: The TUH dataset used in this study. A patient "with seizures" has at least one 16 sec seizure segment. The validation and test samples have no overlap.

2.2. Models

Seizure detection can be considered as a form of supervised learning called multiple instance learning (MIL), where each seizure event may manifest differently across various channels, depending on the type of seizure and individual patient characteristics. The problem of EEG artifact identification has been formulated in this way [22]. Localized seizures, such as focal seizures, typically affect a specific area of the brain and may only be captured by a subset of the recorded channels. In contrast, generalized seizures involve the whole brain and are likely to be detected across most or all channels. In either case, the goal is to identify the presence of a seizure by analyzing the information from multiple channels simultaneously.

MIL approaches tackle this problem by considering a collection of instances (channel features in our case) as a "bag", where at least one instance within the bag is indicative of the target event (seizure). Pooling methods are often employed in MIL to aggregate information from multiple instances, aiming to identify the most informative instances that contribute to the detection of the target event. Max pooling is a popular pooling method in this context, as it retains the maximum instance from the bag, highlighting the most prominent features or activations across the bag. By utilizing max pooling, the model can potentially focus on the most informative channels for detecting the presence of a seizure, whether it is localized or generalized, while disregarding less relevant channels that may not contribute significantly to seizure detection. This method has the advantage of being simple and computationally efficient but it discards the feature information from other channels which might be relevant.

2.2.1. Baseline model:

CNNs have been shown to be effective in analyzing EEG signals for the detection of epileptic seizures [15]. We propose a simple CNN architecture as a baseline model for seizure detection. The motivations for choosing a simple model include its ease of understanding and implementation, fast training, and the ability to expand the model. By using a simple model, we can assess the extent to which increased complexity contributes to the performance of more advanced models.

The proposed model architecture begins with an input layer that takes EEG as input, followed by a 1D convolutional block, which consists of a convolution operation followed by batch normalization and a ReLU activation function. Following this is a global average pooling layer. This reduces each channel feature to a single value for the following channel-wise max pooling layer. Each EEG channel is separately processed through these initial layers. Subsequently, a flatten and gather operation combines the features derived from all 22 EEG channels into a unified set of features. Max pooling then operates on this combined feature set, treating each EEG channel independently. The final layer is fully connected and provides a scalar output between 0 and 1 that indicates the probability of a seizure (figure 2.2). The max pooling operator could allow the model to identify and retain the most informative features from the feature maps produced by the convolutional block, potentially highlighting the most relevant channels for detecting the presence of a seizure.

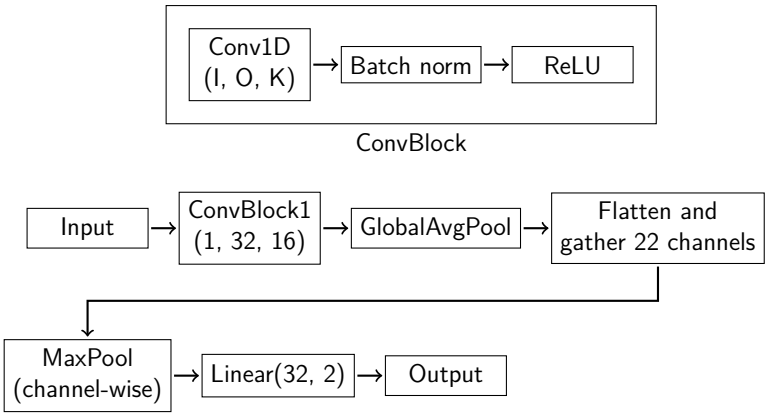


Figure 2.2: Schematic representation of baseline model. ConvBlock:(input channels, output channels, kernel size), Linear:(input size, output size). Each EEG channel is individually processed through the feature extractor (ConvBlock1 and GlobalAvgPool). The extracted features are then combined into a vector, containing features for each EEG channel, which the max pooling operates on.

2.2.2. 4x conv model:

We build upon the baseline model by stacking 4 convolutional blocks. This refined model is then composed of 4 convolutional blocks, followed by a global average pooling layer, a flatten and gather operation, a max pooling layer, and a fully connected layer (see figure 2.3). By increasing the depth, we expect the model to be capable of learning more complex features. Hence, the increased capacity should enable the model to recognize more advanced patterns which the simpler model was not capable of learning.

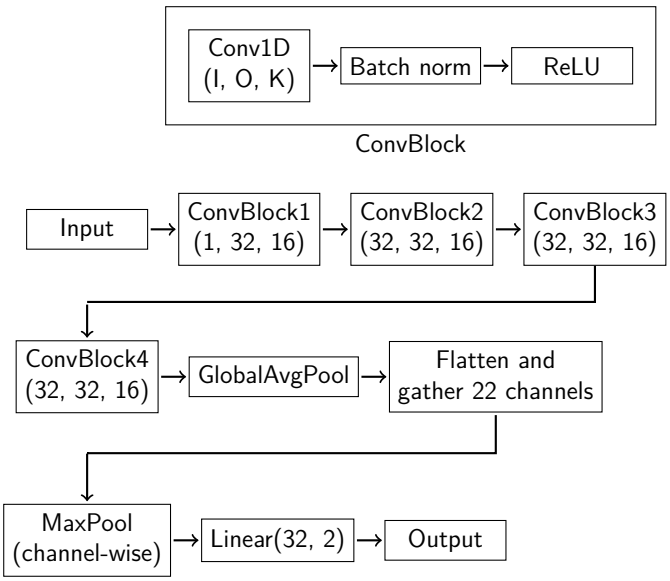


Figure 2.3: Schematic representation of 4x conv model. ConvBlock:(input channels, output channels, kernel size), Linear:(input size, output size). Each EEG channel is individually processed through the feature extractor (from ConvBlock1 to GlobalAvgPool). The extracted features are then combined into a vector containing features for each EEG channel which the max pooling operates on.

2.2.3. 7x conv model with max pooling:

To explore the effects of an even greater capacity, we design a model with 7 convolutional blocks. The 7x conv model consists of 4 initial convolutional blocks, followed by an average pooling layer with a kernel size of 8 and stride 3, and then 3 additional convolutional blocks. Finally, as before, we have an average pooling layer, a flatten and gather operation, a max pooling layer and a fully connected layer.

2.2.4. Model from Borovac et al. [26]:

This model combines a convolutional network with an attention module [26] and has been employed both for neonatal and adult EEG seizure detection with good results [25]. The feature extractor is from [27] and the attention layer is from [28]. The attention layer addresses the MIL problem [29] with the aim of focusing on the channels most relevant to seizure detection. Additionally, the attention layer can help explain the results of the classifier by highlighting which channels had most weight in the final classification, as shown in [28]. We use this model as a comparison to the simpler models and to evaluate various training methods. We also investigate the impact of using attention instead of max pooling to address the MIL problem.

The architecture of the model is shown in figure 2.5. Each convolutional block in the architecture consists of a 1D convolution (kernel size 3, 32 filters), a batch normaliza-

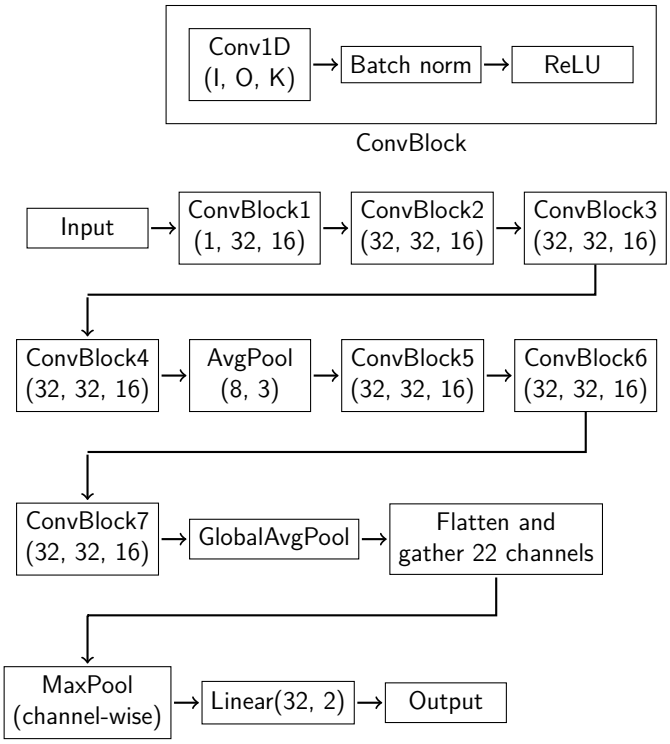


Figure 2.4: Schematic representation of 7x conv model. ConvBlock:(input channels, output channels, kernel size), Linear:(input size, output size). Each EEG channel is individually processed through the feature extractor (from ConvBlock1 to GlobalAvgPool). The extracted features are then combined into a vector containing features for each EEG channel which the max pooling operates on.

tion layer followed by a ReLU activation. We feed each EEG channel individually to the feature extractor and then collect outputs after feature extraction, same as the other models. The attention layer is incorporated after the Cork feature extractor to help the model focus on the most informative features, followed by a fully connected layer for binary classification to determine the presence of a seizure.

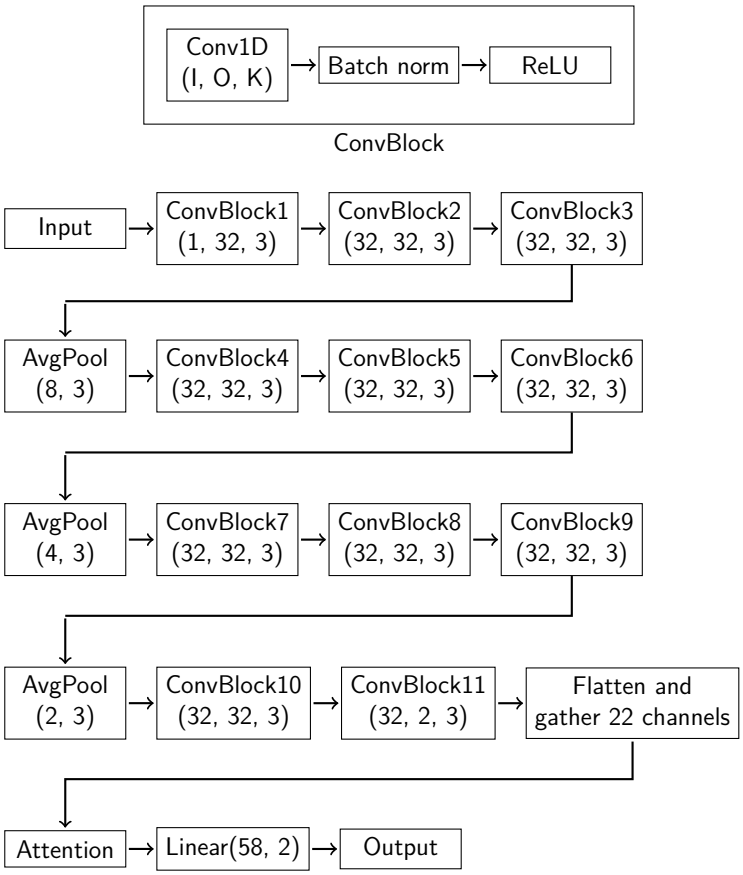


Figure 2.5: Schematic representation of model from Borovac et al. [26]. ConvBlock:(input channels, output channels, kernel size), AvgPool:(size, stride), Linear:(input size, output size). Each EEG channel is individually processed through the feature extractor (from ConvBlock1 to ConvBlock11). The extracted features are then combined into a vector, containing features for each EEG channel, for the attention layer.

The main difference between this model and the expanded baseline models is its depth and the usage of attention instead of max pooling in the classification component. This model also uses a smaller kernel size in its convolutional layers.

2.2.5. Cork-exp+attention:

We expand the feature extractor in the model from Borovac et al. with two more pooling layers and six more conv blocks. Our goal with this model is to determine if a deeper

feature extractor can generate more informative features and produce better performance. We use attention along with a linear layer for classification, the same as in the model from Borovac et al. The pooling layers have stride 2 instead of 3 so the feature extractor outputs similar size features as before. The architecture is shown in figure 2.6.

2.2.6. TCN+attention:

Here we attempt to improve the feature extractor in the model from Borovac et al. with the use of a Temporal Convolutional Network (TCN) block [30]. The motivation for using a TCN block is its ability to capture long-range dependencies in the input data, which can be particularly beneficial for time series and sequence data. TCN blocks utilize dilated convolutions, enabling them to learn complex temporal patterns without losing the resolution of the input signal [30]. By effectively modelling temporal dependencies, we hope to enhance the model's performance and generate more informative features. Additionally, seizure activity often evolves in time, with its frequency altering as it progresses where as some rhythmic artefacts (like respiratory artefacts) maintain a relatively constant frequency. The TCN blocks could then potentially detect these differences and result in fewer false positives. The model is shown in figure 2.7.

The TCN blocks consist of 3 layers with 32 channels and a kernel size of 16. The dilation factor exponentially increases from 1 in the first layer to 4 in the third layer, allowing the model to capture both local and long-range dependencies. The attention mechanism and linear layer for classification are retained, similar to the previous model.

2.2.7. Inception+attention:

This model, which again builds upon the architecture proposed by Borovac et al., introduces a modification in the feature extractor by replacing convolutional blocks with 1D Inception-style blocks. With this we aim to enhance the seizure detection capabilities by capturing local EEG features at varying scales.

The 1D Inception block is based on the original inception block [31] used in image classification, is a convolutional block that applies multiple filter sizes to the same input and concatenates the results, allowing the model to capture local features at different scales. The 1D Inception block consists of three parallel 1D convolution branches with different filter sizes (1, 3 and 5), each producing 32 output channels. The outputs of these branches are concatenated along the channel dimension. We encapsulate this Inception block between two convolutional blocks. The whole archtecture is layed out in figure 2.8.

This modified architecture brings together the benefits of the Inception block's multi-scale feature extraction and the attention mechanism's ability to focus on the most relevant features for seizure detection. The inception block has also been used in the feature extractor from Isaev et al. [28], but here we use more filters and have a pointwise convolution instead of the max pooling branch.

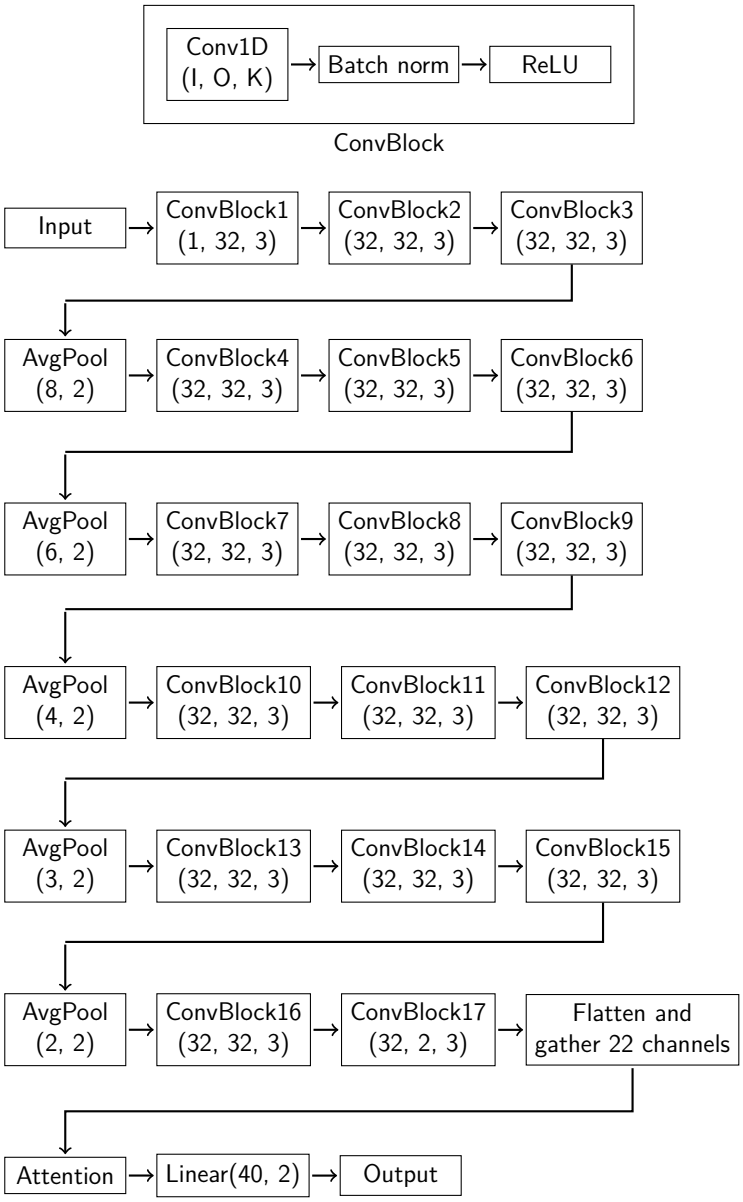


Figure 2.6: Schematic representation of Cork-exp+attention model. ConvBlock:(input channels, output channels, kernel size), AvgPool:(size, stride), Linear:(input size, output size). Each EEG channel is individually processed through the feature extractor (from ConvBlock1 to ConvBlock17). The extracted features are then combined into a vector, containing features for each EEG channel, for the attention layer.

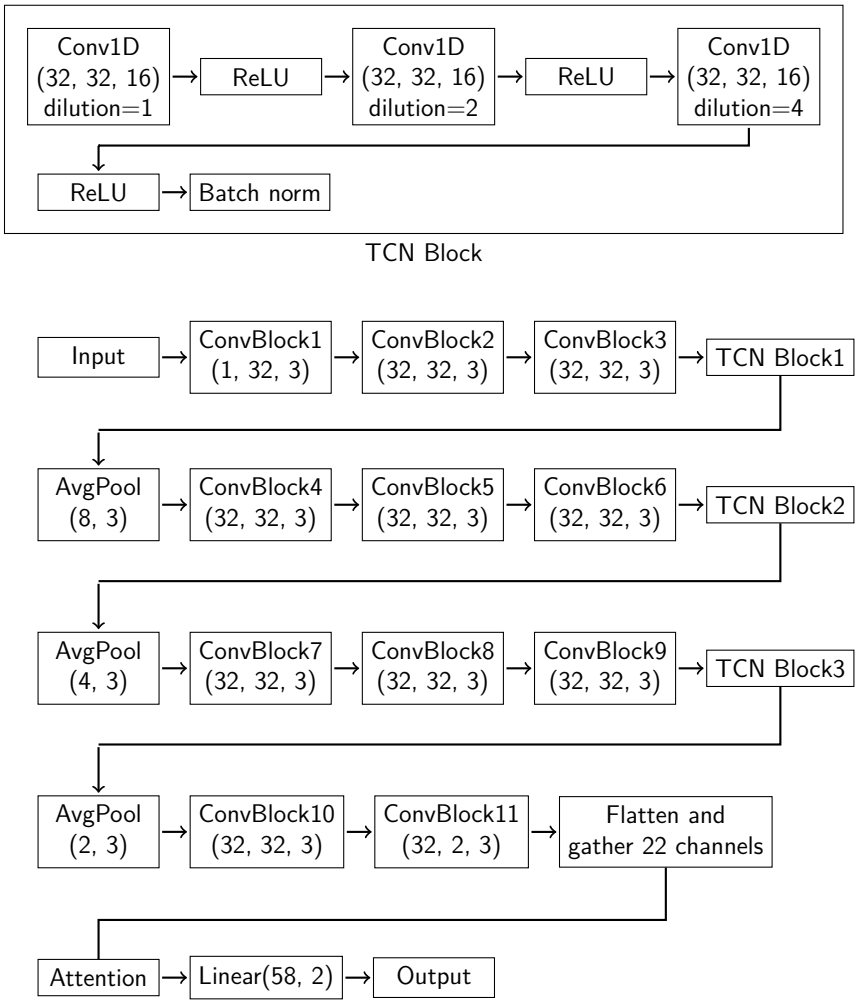


Figure 2.7: Schematic representation of TCN+attention model. ConvBlock:(input channels, output channels, kernel size), AvgPool:(size, stride), Linear:(input size,output size). Each EEG channel is individually processed through the feature extractor (from ConvBlock1 to ConvBlock11). The extracted features are then combined into a vector, containing features for each EEG channel, for the attention layer.

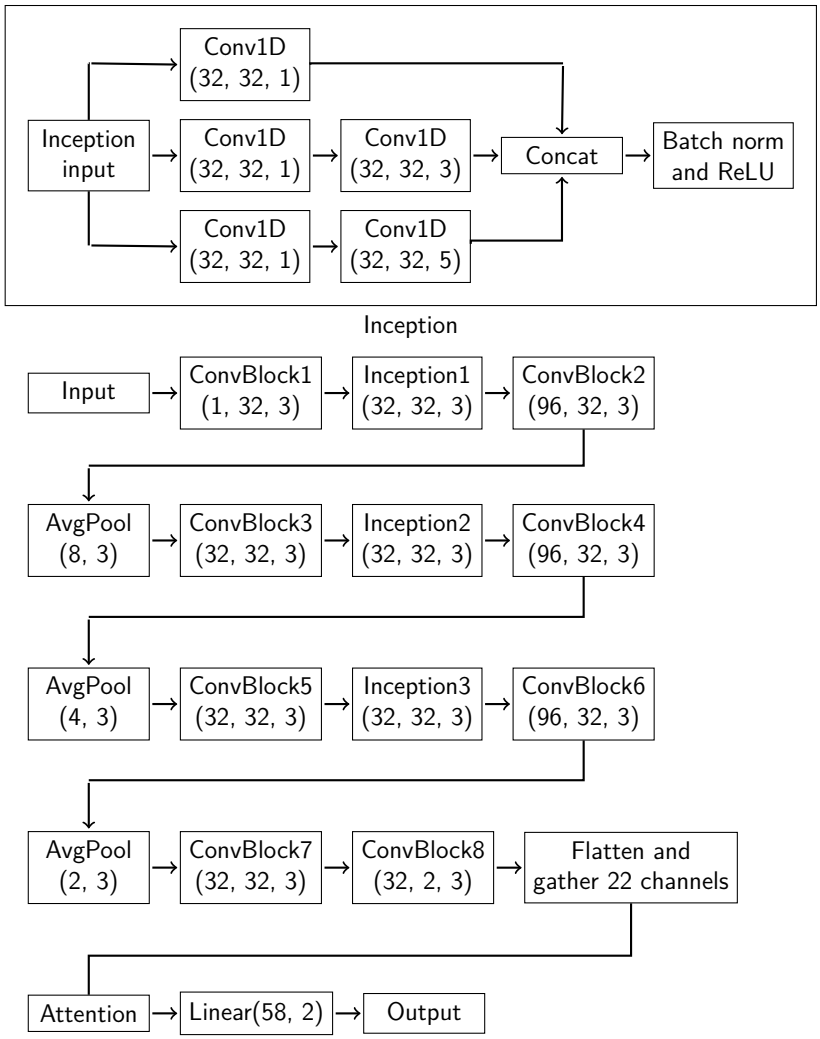


Figure 2.8: Schematic representation of Inception+attention model. ConvBlock:(input channels, output channels, kernel size), AvgPool:(size, stride), Linear:(input size,output size). Each EEG channel is individually processed through the feature extractor (from ConvBlock1 to ConvBlock8). The extracted features are then combined into a vector, containing features for each EEG channel, for the attention layer.

2.3. Training methods

In our study we explore various training strategies. Here we describe these methods and their implementation, along with our hyper-parameters settings.

Sliding window sampling (our default sampling strategy):

The sliding window method is a popular technique for sampling time series data, particularly in the field of EEG. In this approach, a window of specified length, moves over the data and a sample is generated for each window (16 seconds in our case). Typically, the window is shifted with overlap, meaning that consecutive windows share some data points. The sliding window method with overlap has become a standard practice when sampling EEG data.

In our experiments we balance the training set and use 12 second overlap (window is shifted 4 seconds between samples) for seizure samples and no overlap for non-seizure samples. This approach is used to artificially increase the number of seizure segments before we balance the dataset, resulting in a larger training set.

The validation and test sets are sampled with this method in all experiments but in this case no overlap is used.

Segment translation sampling:

Instead of sliding a window over the data, this method picks random points in the time series and generates samples from windows starting at these points. This approach offers a more flexible and location-independent way of sampling the time series. We set a 50/50 probability for drawing seizure and non-seizure segments, mimicking a balanced dataset.

Using this method we choose the amount of samples that we want to generate for each epoch, with the training samples varying between epochs. In our experiments we choose to generate 40,000 samples per epoch, the same number of samples that the sliding window method produces, to have a fair comparison between the sampling methods. We make sure that each sample is entirely seizure or non-seizure, excluding seizure onsets and offsets.

Mixup:

Mixup is a data augmentation technique that has demonstrated its effectiveness in enhancing the generalization of various neural network architectures [32], particularly for image and text classification. Mixup can serve to regularize the neural network, thus aiding in the prevention of overfitting.

Mixup creates augmented training samples by forming linear combinations of input-target pairs. A new input-target pair $(\tilde{x}, \tilde{y})$ is generated as follows:

$$\tilde{x} = \lambda x^{(i)} + (1-\lambda)x^{(j)}, \tag{2.1}$$

$$\tilde{y} = \lambda y^{(i)} + (1-\lambda)y^{(j)}, \tag{2.2}$$

where $x^{(i)}$ and $x^{(j)}$ represent two randomly chosen training EEG samples, $y^{(i)}$ and $y^{(j)}$ are their corresponding 0/1 (non-seizure/seizure) labels, and $\lambda \in [0,1]$ is a random variable drawn from a Beta distribution with hyper-parameter α.

Selecting an appropriate α value is crucial for achieving optimal results; in this study, we fixed $\alpha = 0.3$ after evaluating various values on the validation set.

Manifold Mixup:

Manifold Mixup is an extension of the Mixup technique [33]. Unlike Mixup, which operates only on input features and target labels, Manifold Mixup linearly interpolates the hidden activations of a model at a randomly chosen layer.

The Manifold Mixup procedure is as follows:

Uniformly at random, select a layer from the model. For each pair of samples $(x^{(i)}, y^{(i)})$ and $(x^{(j)}, y^{(j)})$, compute the corresponding hidden activations $h^{(i)}$ and $h^{(j)}$ at the chosen layer. Generate a new pair of samples with the interpolated hidden activations and target labels using the same Mixup technique:

$$\tilde{h} = \lambda h^{(i)} + (1-\lambda) h^{(j)}, \tag{2.3}$$

$$\tilde{y} = \lambda y^{(i)} + (1-\lambda) y^{(j)}, \tag{2.4}$$

where $\lambda \in [0,1]$ is a random variable drawn from a Beta distribution with hyper-parameter α. The α parameter was set to 0.3.

Manifold Mixup has been shown to enhance the performance of various neural network architectures in image classification [33].

Feature-level Mixup:

Feature-level mixup is another variant of the Mixup technique [34]. Feature-level Mixup can be considered as a constrained version of Manifold Mixup, where the layer selected for augmentation is fixed to be the output of the feature extractor. In this approach, the Mixup method is applied specifically to the feature representations of the input samples, rather than the input samples themselves or the hidden layers within the feature extractor. Since we only train the feature extractor on non-augmented inputs, it might produce more realistic features. We then attempt to gain the benefits of mixup in the classification stage by augmenting the features, making the latent space richer.

To create a new feature-target pair $(\tilde{z}, \tilde{y})$ using Feature Mixup, the process is as follows:

$$\tilde{z} = \lambda z^{(i)} + (1-\lambda) z^{(j)}, \tag{2.5}$$

$$\tilde{y} = \lambda y^{(i)} + (1-\lambda) y^{(j)}, \tag{2.6}$$

where $z(i)$ and $z(j)$ are outputs of the feature extractor for two randomly chosen two randomly selected training EEG segments and $y(i)$ and $y(j)$ are the corresponding class labels. The α parameter was set to 0.3.

Deep ensembles

Ensemble learning is a technique that combines multiple models to improve overall performance and reduce overfitting in machine learning tasks [35]. A "deep ensemble", which is a combination of multiple Deep Neural Networks (DNNs), has been shown to offer slight improvements in classification performance in comparison to the top-performing individual model within the ensemble [36]. In our approach, we employed a five-model ensemble, and the final prediction was derived by averaging the logit outputs before passing them through a softmax function.

Dropout:

Dropout is a commonly used regularization technique in neural networks to prevent overfitting. It functions by randomly deactivating a fraction of neurons during training, thereby reducing co-adaptation of features and increasing the model's generalization capability [37].

We used the model from Borovac et al. when experimenting with dropout. In our approach, dropout was applied at two stages in the network. First, a dropout rate of 0.1 was used between the layers in the feature extractor. This helped to ensure that the model did not overly rely on specific features or paths within these layers. Subsequently, a higher dropout rate of 0.5 was applied to the fully connected classification layer, further increasing the model's robustness by preventing over-dependence on certain neurons for classification.

Z-score Normalization:

Z-score normalization, also known as standardization, is a technique which translates and scales the input data so the mean is zero and standard deviation is one. This transformation helps to compare different samples on a common scale and can improve the performance of certain machine learning algorithms [38]. The z-score normalization formula is as follows:

$$z_i = \frac{x_i - \mu}{\sigma} \tag{2.7}$$

Where z_i is the normalized value of the i-th data point, x_i is the original value, μ is the mean of the sample, and σ is the standard deviation of the sample. We normalize each channel in each segment individually.

2.4. Calibration methods

Here we list methods that have been used to improve model calibration in other domains, particularly deep neural networks for image classification. We will investigate the effects of these methods on the calibration of our models.

Mixup, has been shown in previous work to improve the calibration of classifiers for both image and text data [24]. By introducing more diversity into the training data and effectively increasing the size of the training set, mixup could help to better align the model's predicted probabilities with the true outcome probabilities.

Deep ensembles, have been reported to have the potential to improve both model calibration and performance in certain contexts. As observed by Lakshminarayanan et al. [39], using an ensemble of just five models, all trained with identical settings, can lead to a notable improvement in calibration in some scenarios.

Temperature scaling is a post-processing method that adjusts the confidence estimates of a pre-trained neural network [23]. It introduces a single parameter, the temperature, which is optimized on a validation set after the model is trained, with the aim of minimizing a certain loss (typically the negative log-likelihood). Once the optimal temperature is found, it is used to adjust the softmax output of the model for future predictions. This method has been found effective in some cases for improving the calibration of neural networks across various tasks and could potentially enhance calibration of our models.

2.5. Evaluation methods

We assess our model using the dedicated evaluation (test) data provided in the TUH corpus. The evaluation is conducted on both a segment-basis and a patient-basis:

- **Segment-basis**

  For each patient in the evaluation set, we sample using a sliding window with no overlap, ensuring that we do not test the model with segments containing the same EEG data points. After collecting all the samples, we calculate the evaluation metrics for the entire set of segments. By evaluating the model this way, we put more emphasis on patients with longer recordings since they produce more samples.

- **Patient basis**

  For each patient in the evaluation set, we sample using a sliding window with no overlap. We then compute the evaluation metrics for each patient individually and average these metrics across all patients in the test set. Evaluating the model this

way can inform us on how the model performs in a clinical setting. Since all patients weight the same, we can better predict the models performance on an unseen patient.

The patient-based metrics are computed from the averages across a few patients (32 patients), and thus, each patient carries substantial weight. This means that subpar performance on just a handful of patients can significantly skew the overall patient-based metrics. Therefore, in our analysis, we choose to focus on segment-based metrics, which provide a more consistent and reliable evaluation of model performance. In the test set, we exclude patients who do not exhibit seizures since the patient-based evaluation metrics necessitate the presence of both seizure and non-seizure classes. We do not include these patients in the segment-based metric to maintain consistency and enable a meaningful comparison of the results between the two approaches.

2.5.1. Performance metrics

Area under the curve:

The Area under the curve (AUC) is a widely used performance metric in seizure classification when the classification threshold can be modified [40]. Compared to the accuracy metric, it is less sensitive to class imbalance. AUC quantifies the ability of a classifier to rank samples based on output probability, reflecting the probability that a randomly chosen positive (seizure) sample will have a higher score than a randomly chosen negative (non-seizure) sample. It measures the classifier's performance across all possible classification thresholds, capturing its ability to distinguish between the positive (seizure) and negative (non-seizure) classes. A higher AUC indicates better overall classification performance, with a value of 1.0 representing perfect classification and a value of 0.5 corresponding to classify samples uniformly at random. We use AUC as our main metric when comparing different architectures and methods.

Sensitivity:

Sensitivity (SE), also known as true positive rate or recall, measures the proportion of actual positive instances (seizures) that are correctly identified by the classifier. In seizure detection, sensitivity is important, as it reflects the model's ability to detect seizures when they occur. Sensitivity is calculated using the following equation:

$$\text{Sensitivity} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{2.8}$$

Specificity:

Specificity (SP), also known as true negative rate, measures the proportion of actual negative instances (non-seizures) that are correctly identified by the classifier. In seizure detection, specificity is important, as it indicates the model's ability to correctly identify

non-seizure EEG segments and avoid false alarms. Specificity is calculated using the following equation:

$$\text{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}} \tag{2.9}$$

F1 score:

The F1 score is a harmonic mean of precision and recall (sensitivity). It provides a balanced measure of a classifier's performance, particularly in the presence of imbalanced classes. It is calculated as follows:

$$\text{F1 score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.10}$$

where Precision (also known as Positive Predictive Value) is the proportion of true positive predictions among all positive predictions made by the classifier:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{2.11}$$

The F1 score is useful in seizure detection, as it takes into account both the classifier's ability to identify seizures and to avoid false alarms. A higher F1 score indicates better overall classification performance.

In our study, we focus more on the AUC as it provides a single value summarizing the classifier's performance across all possible classification thresholds, rather than just at a specific threshold as the F1 score does. AUC allows us to evaluate the model's performance at all classification thresholds and better understand its trade-offs between sensitivity and specificity.

2.5.2. Calibration metrics

Before introducing the calibration metrics, it is important to note that these metrics are only calculated on a segment basis rather than on a patient basis in the experiments. This approach is taken because the calibration methods require a sufficient number of samples for both seizure and non-seizure classes to yield accurate results. In our dataset, some patients have very few seizure segments, making it difficult to evaluate the model's calibration performance on a per-patient basis.

Expected Calibration Error:

The Expected Calibration Error (ECE) is a metric that measures the average difference between the predicted probabilities and the observed frequencies [41]:

$$\mathrm{ECE} = \sum_{k=1}^{K} \frac{|B_k|}{N} \left|\mathrm{acc}(B_k) - \mathrm{conf}(B_k)\right|, \tag{2.12}$$

where K is the number of bins, B_k is the set of instances in bin k, N is the total number of instances, $\mathrm{acc}(B_k)$ is the observed frequency for bin k, and $\mathrm{conf}(B_k)$ is the average confidence estimate for bin k. We set the number of bins to $K = 5$ for all calibration evaluation metrics.

Overconfidence Error (OE):

The Overconfidence Error (OE) is a metric that measures the average difference between the predicted probabilities and the observed frequencies for bins in which the model is overconfident [24]:

$$\mathrm{OE} = \sum_{k=1}^{K} \frac{|B_k|}{N} \max(\mathrm{conf}(B_k) - \mathrm{acc}(B_k), 0) \tag{2.13}$$

In medical settings, we prefer classifiers that don't overestimate their accuracy. Therefore we include this metric in our analysis.

Static Calibration Error (SCE):

The Static Calibration Error (SCE) [42] is a metric that measures the average difference between the predicted probabilities and the observed frequencies for each class, addressing imbalanced data issues:

$$\mathrm{SCE} = \frac{1}{K} \sum_{k=1}^{K} \sum_{c=1}^{C} \frac{|B_{c_k}|}{N_c} |\mathrm{acc}(B_{c_k}) - \mathrm{conf}(B_{c_k})|. \tag{2.14}$$

In this definition, the weighting factor $\frac{|B_{c_k}|}{N_c}$ is used, where N_c is the number of segments of class c (seizure or non-seizure). The weights are proportional to the number of segments in each class, rather than the total number of segments. This ensures that all classes have equal weight in the overall sum, effectively addressing the imbalanced data issue.

We primarily focus on the SCE metric to evaluate model calibration in our analysis since our data set is highly imbalanced. By taking into account the imbalanced nature of the data and providing equal weight to each class, the SCE offers a more informative assessment of the model's calibration performance for both seizure and non-seizure classes.

Brier score:

The Brier score is a metric that measures the mean squared difference between the predicted probabilities and the actual outcomes:

$$\mathrm{BS} = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{y}^{(i)} - y^{(i)}\right)^2, \tag{2.15}$$

where N is the total number of instances, $\hat{y}^{(i)}$ is the predicted probability of instance i, and $y^{(i)}$ is the actual outcome of instance i.

Negative log-likelihood:

The Negative Log-Likelihood (NLL) is a metric that measures the logarithm of the likelihood of the true labels given the predicted probabilities:

$$\mathrm{NLL} = -\sum_{i=1}^{N} y^{(i)} \log \hat{y}^{(i)} + \left(1 - y^{(i)}\right) \log\left(1 - \hat{y}^{(i)}\right), \tag{2.16}$$

where N is the total number of instances, $\hat{y}^{(i)}$ is the softmax output of instance i for class 1 (seizure class), and $y^{(i)}$ is the target label of instance i.

Confidence plot:

The Confidence Plot is a visual tool used to understand the calibration of a model's predicted probabilities against the actual outcomes. In the context of seizure detection, the confidence plot can help determine if the predicted probabilities of seizure occurrence from the model align with the observed frequencies.

The x-axis represents the bins of instances with similar predicted probabilities. The y-axis corresponds to the observed accuracy within each bin, displayed as histogram bars. The model's confidence (mean predicted probability) for each bin is denoted by horizontal black lines spanning the bin. A model that exhibits excellent calibration would have its black lines (predicted confidence) aligning closely with the top edges of the corresponding histogram bars (observed accuracy). In this case, the model's confidence would be indicative of its actual performance within each bin. If the black lines are consistently above the bars, the model is overconfident and underestimates the prediction errors. Conversely, if the lines fall below the bars, the model is underconfident and overestimates the errors.

2.6. Setup

The experiments were conducted on a cluster with 13x NVIDIA A100 Tensor Core GPUs. To implement and train the models, the PyTorch 1.13.0 framework was used along with TensorBoard for monitoring the training process and evaluating the models.

Each model was trained for 50 epochs using the Adam optimizer. A learning rate of $1e-3$ was applied and halved every 20th epoch. We used a batch size of 256 throughout the training. To speed up the training process and conserve GPU memory, all EEG data was

loaded onto the GPU without including repetitions from overlapping segments. Segments were then sampled on the fly directly from the GPU memory. Each epoch took 16 – 95 seconds, depending on model complexity.

3. Results and Discussion

3.1. Models

In this section, we discuss and compare experiments conducted on several models without employing any specific training methods. Our primary focus is on evaluating the performance metrics of these models to directly compare different architectures.

Each model was trained five times, with random initialization of weights and seeds, to assess the mean performance and variance of different model architectures. This approach accounts for variability that may occur in individual trials, offering a more robust assessment of the model architectures' stability and performance consistency across different initializations. When training the models, we sample the data with the sliding window method and we do not normalize the inputs. We refer to this setup as the basic training procedure.

We attempted to evaluate the ResNet-short+LSTM network presented by Lee et al. [43] as a point of reference; however, we encountered difficulties in successfully implementing the model in our environment. Given the project's time limitations, we refer to the results presented in the paper as an indication of state-of-the-art performance.

3.1.1. Baseline model

In this experiment, our objective was to obtain a baseline performance using a simple model while exploring the impact of various hyper-parameters on its performance. By comparing the performance of this basic model with more complex models, we aimed to assess the advantages associated with an increase in model capacity. The model uses a convolutional block with a kernel size of $k = 16$ and a filter size of $C = 32$. The results of the 5 trials are presented in Table 3.1.

Using the baseline model, we got surprisingly high AUC scores with segment-based AUC reaching 0.882. The AUC validation curve varied smoothly throughout the training process (figure A.1). The balanced training accuracy (an average over the balanced mini batches) was slightly higher than its validation counterpart, with mean balanced train accuracy at 71.72% and mean balanced validation accuracy at 70.71%. Hence, the capacity of the model is very likely the bottleneck for this problem.

	Baseline $C = 32, k = 16$
Parameters	674
Patient-based (patients with seizures)	
Area under the curve	0.891 (0.005)
Sensitivity [%]	50.94 (5.61)
Specificity [%]	92.66 (1.48)
F1-score [%]	36.01 (2.75)
Segment-based (patients with seizures)	
Area under the curve	0.882 (0.003)
Sensitivity [%]	51.67 (11.39)
Specificity [%]	92.06 (1.41)
F1-score [%]	40.14 (4.39)

Table 3.1: Baseline model. Average metrics across 5 trials, standard deviation in parenthesise.

During the training of the baseline model, the validation scores were better than the training scores (see figure A.1). This suggests that the validation data is less challenging to classify. When training more complex models, overfitting the training data became possible (see Appendix Figure A.2).

The sensitivity was very low (52%) while the specificity was high (92%). Since we trained on a balanced dataset, there are several possible explanations for the low sensitivity: The diversity of seizure segments in the training set might be limited, i.e. the training set is too small. The capacity of the model could be insufficient to learn different seizure types. Seizure segments might closely resemble specific non-seizure noise patterns. Many non-seizure segments might be easily identifiable, as they clearly lack any seizure activity. The test data may differ from the training data to a certain extent, we don't have information about how the test set was constructed. In any case, this simple model is fairly good at detecting non-seizures. We produce a very low F1 score of 40.14%. This is expected since it reliant on the sensitivity which is also low.

We next investigated the effect of the filter size k for the convolutional block. Similar results were obtained with $k = 16$ and $k = 32$, but $k = 4$ resulted in lower AUC values (table 3.2).

	kernel size, $k = 16$			Number of filters, $C = 32$		
	C=16	C=32	C=64	k=4	k=16	k=32
Segment-based (patients with seizures)						
Area under the curve	0.885	0.886	0.89	0.843	0.886	0.887
Sensitivity [%]	30.56	31.78	36.45	34.28	31.78	34.21
Specificity [%]	93.87	94.05	93.77	91.19	94.05	93.55
F1-score [%]	32.03	33.42	36.89	31.05	33.42	34.64

Table 3.2: The effects of kernel size and and number of filters for the convolutional block in the baseline model. Each model is trained once (opposed to 5 trials).

We also investigated the effect of varying number of convolutional filters, fixing $k = 16$ (table 3.2). There was no substantial difference in performance between $C = 16$ and $C = 32$ but $C = 64$ yielded marginal improvements in AUC. In the following experiments we used $C = 16$ filters since this minimizes the number of parameters between convolutional layers.

3.1.2. Extending the baseline model

In this experiment we expanded on the baseline model by adding more convolutional blocks, thus increasing the capacity of the model. We experimented with both 4x-conv and 7x-conv models using 16 and 32 channel convolutional blocks. The results are shown in table 3.3.

	4x conv blocks $C = 16, k = 16$	7x conv blocks $C = 16, k = 16$	4x conv blocks $C = 32, k = 16$	7x conv blocks $C = 32, k = 16$
Parameters	13K	25K	50K	100K
Patient-based (patients with seizures)				
Area under the curve	0.928 (0.01)	0.922 (0.006)	0.932 (0.01)	0.918 (0.012)
Sensitivity [%]	60.63 (5.01)	58.98 (4.39)	59.76 (9.95)	63.9 (4.93)
Specificity [%]	97.3 (1.04)	97.73 (0.5)	97.61 (1.54)	96.54 (1.12)
F1-score [%]	60.69 (2.29)	58.96 (3.21)	58.64 (2.65)	58.48 (1.37)
Segment-based (patients with seizures)				
Area under the curve	0.958 (0.003)	0.951 (0.005)	0.955 (0.008)	0.951 (0.007)
Sensitivity [%]	77.32 (4.6)	74.83 (3.54)	76.37 (7.45)	78.62 (3.51)
Specificity [%]	97.47 (0.91)	97.8 (0.51)	97.64 (1.5)	96.71 (0.94)
F1-score [%]	73.75 (1.41)	73.64 (0.38)	74.11 (2.66)	71.21 (2.19)
Expected calibration error [%]	4.02 (0.56)	1.56 (0.26)	3.0 (1.01)	1.21 (0.31)
Overconfidence error [%]	0.0 (0.0)	0.05 (0.08)	0.02 (0.02)	0.0 (0.0)
Static calibration error [%]	6.3 (1.07)	7.85 (1.3)	6.79 (2.06)	6.52 (1.45)
Brier score	0.03 (0.0)	0.03 (0.0)	0.03 (0.01)	0.04 (0.01)
Negative log likelihood	0.14 (0.01)	0.12 (0.01)	0.13 (0.02)	0.14 (0.02)

Table 3.3: Extending the baseline model. Metrics are averaged over 5 trials, with the standard deviation shown in parentheses.

We observed an increase in performance for all metrics with regard to the baseline model. However, the 7x-conv model yielded lower scores to the 4x-conv model. There was no significant difference between 16 and 32 filters despite the large increase in parameters for $C = 32$. While capacity is clearly the bottleneck for the baseline model, that does not seem to be the case for the 4x conv network with $C = 16$. It is possible that the network has already extracted most the learnable information from the data using only 13K parameters in the 4x-conv model with $C = 16$ when training with the basic training method.

As we transitioned from the baseline model to the extended models, there is a noticeable increase in segment-based AUC, which then exceeds the patient-based AUC. This can be

attributed to the presence of some patients who are more challenging to classify, even when using a superior classifier, which then lowers the patient-based AUC score (see figure 3.2). However, in the segment-based metric for the the extended models, the easier-to-classify patients overshadow these harder cases, making the impact less apparent in the score.

Our results indicate that while the performance improves when moving beyond the baseline model, the 7x-conv model yields lower scores than the 4x-conv model. Furthermore, there is no significant difference between using 32 filters and 16 filters, suggesting that the capacity bottleneck may not be an issue for the 4x conv network with $C = 16$.

3.1.3. Cork model

The purpose of this experiment was to explore a deeper feature extractor along with a different classification strategy. We first tested the Cork feature extractor [27] with an attention mechanism [28] (model from Borovac et al. [26]) and then with the simpler max-pooling mechanism for classification. This feature extractor has more layers than previous models (11 convolutional block layers) with the models having 30 - 32K parameters in total. We trained 5 instances of both models and evaluated them as before (table 3.4).

	Max instance	Attention
Parameters	32 K	30 K
Patient-based (patients with seizures)		
Area under the curve	0.938 (0.005)	0.93 (0.008)
Sensitivity [%]	72.67 (2.86)	71.1 (1.78)
Specificity [%]	97.23 (0.59)	96.43 (0.97)
F1-score [%]	65.32 (2.54)	61.84 (2.06)
Segment-based (patients with seizures)		
Area under the curve	0.959 (0.003)	0.957 (0.002)
Sensitivity [%]	78.71 (2.57)	79.26 (2.42)
Specificity [%]	97.4 (0.56)	96.57 (0.91)
F1-score [%]	74.21 (1.76)	71.03 (2.65)
Expected calibration error [%]	1.09 (0.15)	1.73 (0.17)
Overconfidence error [%]	1.05 (0.17)	1.67 (0.15)
Static calibration error [%]	7.02 (0.87)	7.05 (0.74)
Brier score	0.03 (0.0)	0.04 (0.01)
Negative log likelihood	0.12 (0.01)	0.15 (0.01)

Table 3.4: Cork feature extractor using different classifiers: Max instance vs. Attention. Metrics are averaged over 5 trials, with the standard deviation shown in parentheses.

The results were practically identical between max pooling and attention. This suggests that the feature extractor is likely responsible for the bulk of the model's performance.

The ablation study in [28] compared the attention module with the averaging of per-channel features after the feature extractor and then passing the result through through a fully connected layer for final classification. They found that the feature extractor they

used was the primary source of classification power in their approach, while the attention layer provided only marginal improvement in AUC score. This observation was made using neonatal data.

We also experimented with combining the 4x convolutional network and the attention module, but found that this approach significantly reduced performance. However, when we modified the model by using a larger kernel size for the last convolutional layer, which resulted in a smaller input size for the attention module, the performance improved. This observation suggests that the attention module's efficacy is highly dependant on the architecture.

Our results showed that the Cork feature extractor, combined with max pooling or attention, does not outperform the simpler 4x convolutional model with 16 filters (30 – 32K v.s. 13K parameters), despite the differences in architectures and capacity. This finding suggests that more complex architectures may not yield significant improvements over the 4x-conv model when training on this dataset without any augmentations.

3.2. Training strategies

We evaluated several training strategies that have been used in other domains to enhance model performance, together with a randomized strategy for selecting seizure segments during training. We compared the effects of the following methods, individually and in combination: Normalization, segment translation, mixup, manifold mixup, feature-level mixup, dropout and ensemble.

We trained the model from Borovac et al. [26] with these methods and compared them to the basic training procedure used in previous experiments, where the input data is not normalized and we sample with a sliding window. Each method was evaluated by averaging over 5 trials, except for the ensemble which uses all 5 models. The results are shown in table 3.5.

Compared to basic training, normalizing the input data led to worse performance across all metrics. Potential reasons for this might be that input normalization discards the amplitude difference between seizure and non-seizure samples, which could be an important characteristic for seizure detection. Additionally, the network already employs batch normalization, which normalizes data with respect to the entire batch instead of individual training samples, preserving the amplitude difference. Therefore, providing raw samples to the network might allow batch normalization layers to work with more information, particularly for the first layer.

Our findings show that, compared to basic training, mixup, feature-level mixup, manifold mixup, dropout, segment translation and ensemble improved the AUC (see figure 3.1). Ensemble and mixup and dropout produced the highest increase in AUC when applied individually. While the feature-level mixup strategy improved the AUC, it did not outperform standard mixup. We also observed no significant difference between the AUC values

for Mixup and Manifold mixup. Using both mixup and segment translation together gave the largest improvements in segment and patient-based AUC when no ensembles were involved. While we attempted to further enhance these results by adding dropout with mixup and segment translation, we found that this approach was less effective.

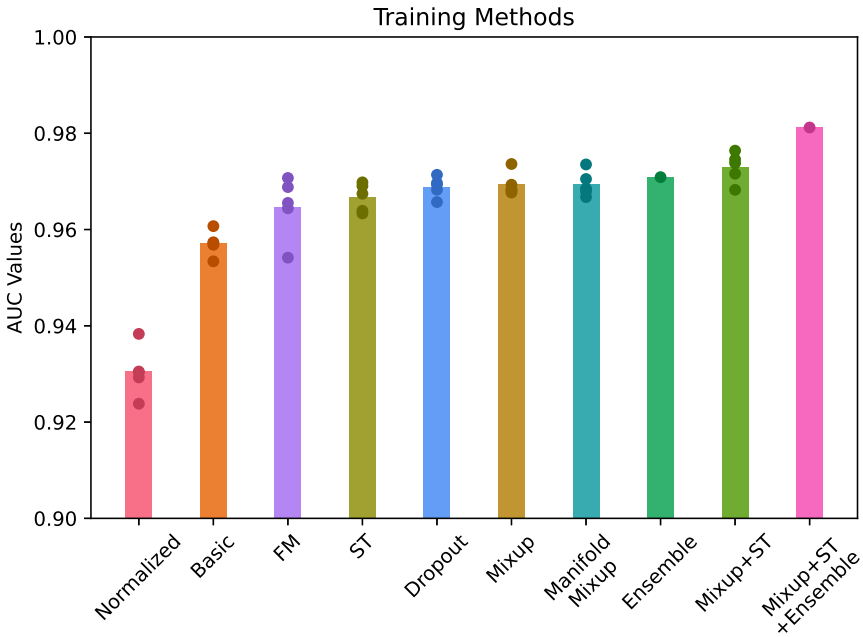


Figure 3.1: Model from Borovac et al. [26] trained with different training methods. Average segment based AUC across 5 models. Each dot represents a single trial. Abbreviations: Feature-level Mixup (FM), Segment Translation (ST).

An ensemble of 5 models along with segment translation and mixup produced the best results. Using these methods gave an increase in AUC of 0.008 compared to the average AUC values for the models in the ensemble (from 0.973 to 0.981). Compared to the basic training we had an increase of 0.024 (from 0.957 to 0.981).

When we examine the F1 score, we observed big fluctuations between the methods. Sensitivity and specificity fluctuate considerably between the methods, and the same is true for recall, which is closely related to specificity. As a result, the F1 score also fluctuates. Manifold mixup results in the highest F1 score of 80.62%.

Previously we observed no significant difference comparing the attention module vs. max pooling method with the Cork feature extractor. We compared them again but now when both were trained with mixup and segment translation (table 3.6 and corresponding column in 3.5). When these methods were employed, attention produced better results than the max pooling method. The reason could be that attention can better capture the increased variability of the data introduced by these training methods.

We had also observed no difference between using the simple 4x-conv model ($C = 16$,

$k = 16$) and using the model from Borovac et al. with the basic training procedure. We next compared these two models along with the 7x-conv model when all were trained with segment translation and mixup, together with their corresponding ensembles (table 3.7 and matching columns in table 3.5).

Using the training methods, the model from Borovac et al. outperformed the 4x-conv model. This was expected since the Borovac model uses attention, which already showed benefits when combined with mixup and segment translation. Comparing the Cork+Max instance model (table 3.6) with the 4x-conv model (table 3.7), we observe similar results, which support the efficacy of attention when these training methods are utilized.

Again, we observed, with more decisive difference in AUC, that the 7x-conv model performed worse than the 4x-conv model. This could be due to the former model being more prone to overfitting, with the simpler architecture in the 4x-conv model producing features that generalize better.

Einizade et al. [44] trained a CNN-LSTM model on the TUH dataset and reported sensitivity of 80% and specificity of 94.98% compared to 85.31% and specificity of 97.39% of our best performing model. The values cannot be compared directly since they train only on a subset of the seizures (focal and generalized seizures), use 4 seconds of EEG data for input compared to our 16 seconds here, employ an older version of the dataset, and utilize different preprocessing methods.

Lee et al. [43] evaluated the performance of several DNN architectures on the TUH data set. The best performing model was a combination of a ResNet feature extractor and an LSTM module which resulted in an AUC score of 0.92 whereas the best performing model in our study had an AUC score of 0.98. Again, these results are not directly comparable since we use different versions of the TUH dataset (they use dataset V1.5.2 and we are using V2.0.0.) and our preprocessing methods differ.

This illustrates the usefulness of data augmentation for SDAs based on DNNs. By effectively expanding the dataset and introducing more variability with mixup and segment translation, models are trained in a more challenging environment, which, as our results indicate, can lead to an improved ability to detect seizures in unseen data.

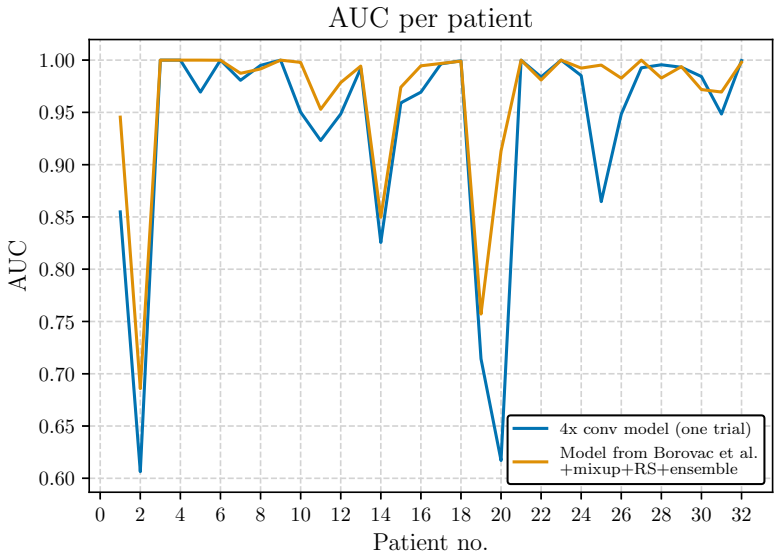


Figure 3.2: AUC for each patient in the test set for the 4x-conv model ($C = 16$) and the model from Borovac et al. [26] with mixup, segment translation and using ensembles. Some patients have very low AUC, disproportionately reducing the patient-based AUC in comparison to the segment-based AUC.

	Normalized	Standard	FM	Mixup+FM	ST
Patient-based (patients with seizures)					
Area under the curve	0.909 (0.012)	0.93 (0.008)	0.939 (0.008)	0.938 (0.011)	0.934 (0.009)
Sensitivity [%]	72.18 (5.15)	71.1 (1.78)	77.54 (1.85)	78.91 (4.23)	72.93 (2.78)
Specificity [%]	92.05 (1.31)	96.43 (0.97)	94.98 (1.02)	91.28 (2.38)	97.11 (0.76)
F1-score [%]	45.47 (2.02)	61.84 (2.06)	61.25 (2.52)	57.55 (1.91)	65.77 (2.72)
Segment-based (patients with seizures)					
Area under the curve	0.93 (0.005)	0.957 (0.002)	0.965 (0.006)	0.965 (0.005)	0.967 (0.003)
Sensitivity [%]	78.04 (3.79)	79.26 (2.42)	86.04 (3.53)	89.59 (2.68)	81.99 (2.66)
Specificity [%]	92.01 (1.52)	96.57 (0.91)	95.1 (1.03)	91.3 (2.73)	97.36 (0.8)
F1-score [%]	55.63 (2.11)	71.03 (2.65)	69.14 (2.05)	59.99 (5.48)	75.97 (2.24)
Expected calibration error [%]	5.85 (0.77)	1.73 (0.17)	0.66 (0.14)	6.77 (2.34)	0.79 (0.27)
Overconfidence error [%]	5.85 (0.77)	1.67 (0.15)	0.08 (0.08)	0.0 (0.0)	0.71 (0.28)
Static calibration error [%]	10.92 (1.13)	7.05 (0.74)	3.07 (1.25)	5.03 (1.53)	5.01 (0.85)
Brier score	0.07 (0.01)	0.04 (0.01)	0.04 (0.0)	0.07 (0.02)	0.03 (0.0)
Negative log likelihood	0.36 (0.03)	0.15 (0.01)	0.15 (0.01)	0.27 (0.05)	0.11 (0.01)

(a)

	ST +Dropout	MM	Mixup	Dropout	Mixup+ST +Dropout
Patient-based (patients with seizures)					
Area under the curve	0.949 (0.008)	0.938 (0.009)	0.942 (0.007)	0.95 (0.006)	0.947 (0.005)
Sensitivity [%]	78.61 (3.17)	66.82 (1.9)	71.91 (3.94)	79.29 (1.25)	81.8 (3.95)
Specificity [%]	95.1 (0.88)	97.69 (0.5)	95.5 (1.85)	95.09 (0.14)	91.1 (2.89)
F1-score [%]	60.41 (1.53)	63.73 (1.83)	60.24 (2.47)	61.17 (1.33)	55.51 (3.55)
Segment-based (patients with seizures)					
Area under the curve	0.968 (0.004)	0.969 (0.002)	0.969 (0.002)	0.969 (0.002)	0.97 (0.002)
Sensitivity [%]	86.38 (2.4)	78.75 (2.36)	84.13 (4.24)	86.72 (1.16)	91.4 (1.8)
Specificity [%]	95.07 (0.76)	97.92 (0.5)	95.71 (1.85)	95.15 (0.07)	91.03 (3.23)
F1-score [%]	69.13 (1.57)	76.62 (1.76)	70.76 (4.68)	69.52 (0.71)	60.34 (6.27)
Expected calibration error [%]	2.4 (0.57)	3.47 (0.53)	3.56 (0.44)	1.6 (0.49)	6.8 (1.19)
Overconfidence error [%]	0.01 (0.02)	0.0 (0.0)	0.0 (0.0)	0.03 (0.05)	0.0 (0.0)
Static calibration error [%]	3.01 (0.72)	4.32 (0.62)	3.72 (0.93)	2.89 (0.32)	6.01 (0.55)
Brier score	0.04 (0.0)	0.03 (0.0)	0.04 (0.01)	0.04 (0.0)	0.07 (0.02)
Negative log likelihood	0.15 (0.01)	0.12 (0.01)	0.16 (0.03)	0.15 (0.0)	0.26 (0.05)

(b)

	Ensemble	MM+ST	Mixup+ST	MM+ST (Ensemble)	Mixup+ST (Ensemble)
Patient-based (patients with seizures)					
Area under the curve	0.95	0.945 (0.009)	0.954 (0.005)	0.959	0.965
Sensitivity [%]	71.48	65.94 (4.73)	71.55 (5.62)	66.97	71.79
Specificity [%]	97.8	98.34 (0.15)	95.88 (1.61)	98.89	97.1
F1-score [%]	66.93	65.24 (2.45)	62.03 (2.21)	69.88	65.63
Segment-based (patients with seizures)					
Area under the curve	0.971	0.972 (0.002)	0.973 (0.003)	0.98	0.981
Sensitivity [%]	80.97	74.87 (2.32)	84.25 (3.38)	75.76	85.31
Specificity [%]	98.03	98.51 (0.14)	96.14 (1.66)	99.06	97.39
F1-score [%]	78.43	77.11 (1.45)	72.43 (4.77)	80.62	77.87
Expected calibration error [%]	1.09	2.93 (0.67)	2.99 (0.87)	3.16	3.63
Overconfidence error [%]	0.91	0.0 (0.0)	0.01 (0.01)	0.0	0.0
Static calibration error [%]	5.8	5.27 (0.56)	3.52 (1.04)	4.24	2.89
Brier score	0.03	0.03 (0.0)	0.04 (0.01)	0.02	0.03
Negative log likelihood	0.1	0.11 (0.0)	0.14 (0.03)	0.09	0.11

(c)

Table 3.5: Experimenting with different training strategies for the model from Borovac et al. [26]. Average (std) over 5 trials (except ensembles). Abbreviations: Feature-level Mixup (FM), Manifold mixup (MM), Segment Translation (ST).

	Cork+Max instance (Mixup+ST)
Patient-based (patients with seizures)	
Area under the curve	0.941 (0.008)
Sensitivity [%]	67.25 (5.78)
Specificity [%]	96.63 (0.63)
F1-score [%]	59.41 (2.49)
Segment-based (patients with seizures)	
Area under the curve	0.967 (0.004)
Sensitivity [%]	81.5 (4.0)
Specificity [%]	96.81 (0.64)
F1-score [%]	73.18 (0.75)
Expected calibration error [%]	3.94 (0.74)
Overconfidence error [%]	0.01 (0.02)
Static calibration error [%]	5.02 (1.19)
Brier score	0.04 (0.0)
Negative log likelihood	0.14 (0.01)

Table 3.6: Cork feature extractor with max pooling mechanism trained with mixup and segment translation (ST). Average (std) over 5 trials.

	4x conv blocks (mixup+ST)	7x conv blocks (mixup+ST)	4x conv blocks (mixup+ST +ensemble)	7x conv blocks (mixup+ST +ensemble)
Patient-based (patients with seizures)				
Area under the curve	0.944 (0.005)	0.93 (0.004)	0.953	0.94
Sensitivity [%]	61.71 (8.79)	60.11 (2.8)	63.22	62.42
Specificity [%]	96.31 (3.31)	97.28 (0.86)	97.28	98.14
F1-score [%]	59.28 (2.57)	57.96 (2.01)	63.4	63.43
Segment-based (patients with seizures)				
Area under the curve	0.967 (0.004)	0.959 (0.002)	0.973	0.966
Sensitivity [%]	80.43 (5.75)	77.16 (3.04)	82.53	78.47
Specificity [%]	96.35 (3.23)	97.17 (0.82)	97.35	98.05
F1-score [%]	72.33 (8.17)	72.31 (1.94)	76.12	77.06
Expected calibration error [%]	9.2 (0.99)	7.32 (1.04)	10.18	7.89
Overconfidence error [%]	0.0 (0.0)	0.0 (0.0)	0.0	0.0
Static calibration error [%]	6.63 (0.71)	5.91 (0.43)	7.27	5.3
Brier score	0.05 (0.02)	0.04 (0.01)	0.04	0.04
Negative log likelihood	0.2 (0.04)	0.17 (0.02)	0.18	0.15

Table 3.7: 4x conv model ($C = 16$) vs model from Borovac et al. trained with mixup and segment translation (ST). Average (std) over 5 trials (except ensembles).

3.3. Attempts at improving the feature extractor

In this section we analyze three modifications to the Cork feature extractor in the model from Borovac et al.. In the first modification we added 2 more pooling layers and 6 additional conv blocks, thereby increasing model capacity (referred to as "Cork-exp" below). In the second, we used temporal convolutions which add the ability to capture long-range dependencies in the input data. In the third modification we replaced 3 conv blocks with inception blocks which enables the model to learn features at multiple scales. To ensure sufficient variability in our data and reveal differences in the architectures' performance, we train the models using mixup and segment translation. The results are shown in table 3.8.

The Cork-exp model performed worse than the model from Borovac et al., with a lower AUC and greater variance between trials (table 3.8a). The F1 scores were also significantly lower, with four trials showing poor results and one trial yielding a much higher score of 71.79%, contributing to the overall high variance. The poor performance of the Cork-exp model in comparison to Borovac et al.'s model could be explained with its increased capacity, which could make the model more susceptible to overfitting.

The TCN model did not outperform the model from Borovac et al. either. It produced greater variation and lower mean AUC values. Given that the segments are only 16 seconds long, there may be limited long-range dependencies within these segments for the TCN block to detect. Utilizing longer segments might be more beneficial for this architecture.

Using the inception block however did result in improvements. The AUC values were higher than the model from Borovac et al. and had minimal variance. The F1 score was also higher. The improvements produced with the inception block could be due to its ability to learn features at multiple scales simultaneously. This might enable the model to capture a wider array of seizure-related characteristics in the EEG signals, result in in the improved stability and performance.

Applying ensemble methods improved both models, bringing the Cork-exp model on par with the model from Borovac et al. (table 3.8b). Despite the boost, the TCN model still performed worse. The significant improvement in the F1 score for the Cork-exp ensemble can likely be attributed to a single model achieving a strong F1 score (71.79%), which was further enhanced through the ensemble method. The Inception ensemble outperformed others with a slightly better segment-based AUC, a modest increase in patient-based AUC, and a superior F1 score.

	Cork-exp+attention (mixup+ST)	TCN+attention (mixup+ST)	Inception+attention (mixup+ST)
Parameters	49K	68K	74K
Patient-based (patients with seizures)			
Area under the curve	0.934 (0.012)	0.95 (0.006)	0.962 (0.004)
Sensitivity [%]	74.91 (3.87)	72.47 (2.55)	73.77 (3.89)
Specificity [%]	92.47 (3.1)	97.24 (0.28)	96.84 (0.97)
F1-score [%]	50.46 (7.52)	65.11 (2.92)	64.68 (1.44)
Segment-based (patients with seizures)			
Area under the curve	0.963 (0.007)	0.969 (0.003)	0.978 (0.001)
Sensitivity [%]	84.4 (3.93)	82.44 (1.1)	85.2 (2.57)
Specificity [%]	93.77 (2.12)	97.33 (0.31)	96.96 (0.99)
F1-score [%]	49.39 (12.78)	76.02 (1.46)	76.1 (2.76)
Expected calibration error [%]	2.9 (1.24)	2.91 (0.63)	5.44 (1.07)
Overconfidence error [%]	0.43 (0.45)	0.07 (0.06)	0.0 (0.0)
Static calibration error [%]	2.91 (0.47)	4.19 (0.64)	4.11 (0.3)
Brier score	0.05 (0.01)	0.03 (0.0)	0.03 (0.01)
Negative log likelihood	0.17 (0.03)	0.12 (0.01)	0.14 (0.02)

(a) Average(std) over 5 trials.

	Cork-exp+attention (mixup+ST +ensemble)	TCN+attention (mixup+ST +ensemble)	Inception+attention (mixup+ST +ensemble)
Patient-based (patients with seizures)			
Area under the curve	0.966	0.961	0.969
Sensitivity [%]	75.41	74.52	74.53
Specificity [%]	97.61	98.31	97.58
F1-score [%]	69.11	71.22	68.47
Segment-based (patients with seizures)			
Area under the curve	0.981	0.979	0.982
Sensitivity [%]	85.24	82.74	86.53
Specificity [%]	97.84	98.44	97.78
F1-score [%]	79.99	81.55	80.4
Expected calibration error [%]	5.21	3.59	5.9
Overconfidence error [%]	0.0	0.0	0.0
Static calibration error [%]	3.3	4.14	4.15
Brier score	0.03	0.02	0.03
Negative log likelihood	0.12	0.1	0.12

(b) Ensemble of 5 trials

Table 3.8: Expanding the Cork feature extractor. Training with mixup and segment translation (ST).

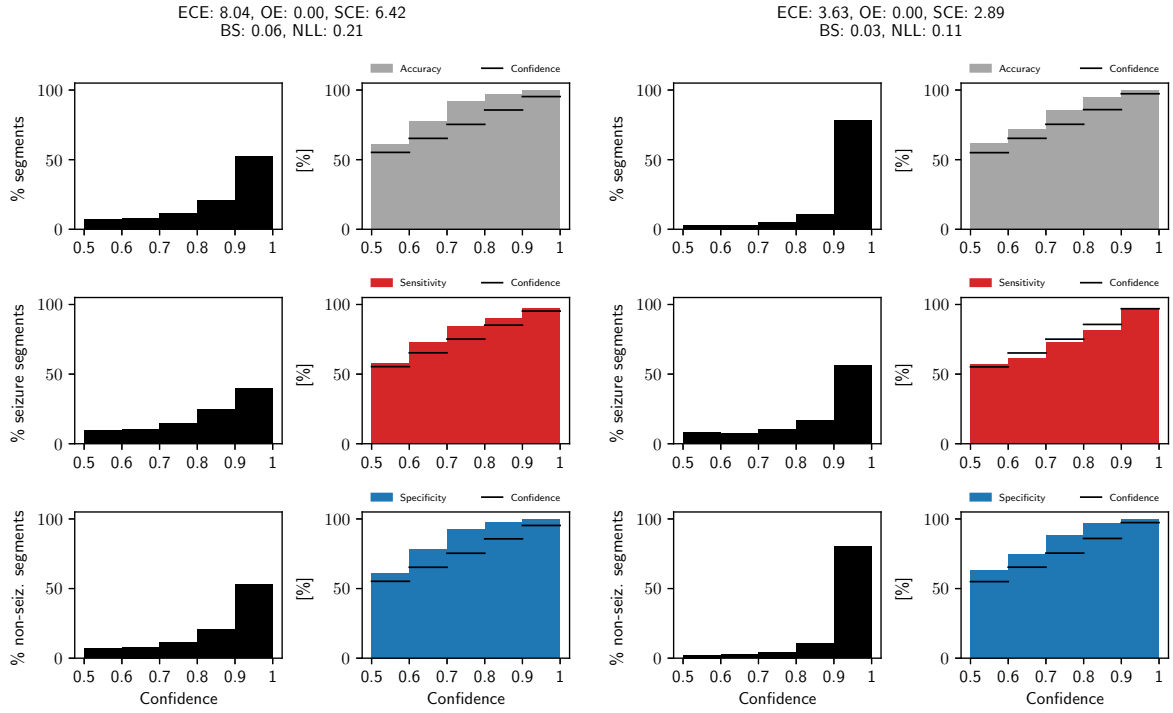
3.4. Calibration methods

We will now evaluate the calibration of various model architectures, followed by an examination of how effective training and calibration methods are in improving calibration. We will focus particularly on the SCE, given the unbalanced nature of our data.

Comparing different architectures when trained with the basic training method, we noticed a significant decrease on the SCE between the baseline model and the 4x conv model. More complex architectures did not significantly improve SCE. Variations of the 4x and 7x conv models, as well as models employing the Cork feature extractor, all yielded an

SCE of approximately 6.5% (table 3.3 and 3.4).

When comparing the different training and calibration methods applied to the model from Borovac et al., we observed noticeable differences in SCE (figure 3.4, table 3.5). When used individually, dropout, mixup, and feature-level mixup most effectively reduced SCE, with dropout being the most effective. Combining ensembles with mixup and segment translation yielded SCE similar to dropout, but with considerably higher AUC scores. This suggests that this combination may both enhance performance and yield a better-calibrated model. Although temperature scaling and dropout reduced SCE when applied independently, they increased SCE when used in combination with mixup and segment translation. Even when applied to the ensemble of the models, temperature scaling failed to improve SCE.



(a) One trial with trained with the basic training procedure.

(b) Ensemble of 5 trials trained with mixup and segment translation.

Figure 3.3: Reliability diagram of for the model from Borovac et al.. Left plots are of one trial with basic training method and the right plots are with mixup, segment translation and ensemble. Black plot shows proportion of segments in each confidence bin. Grey plot shows confidence plot for all segments, red for seizure segments only and blue for non-seizure segments only. Note that The top row represents all predictions with only 7.65% being seizures

The confidence plot for the Borovac model (figure 3.3) shows that using the combination of ensemble, mixup and segment translation leads to an increased number of predictions

within the 0.9 – 1 confidence range. This last bin has accurate calibration and the reduction in SCE could largely be attributed to the model being justifiably more confident, reducing the underconfidence for the other bins. Focusing on the seizure predictions (red), the confidence is a good indicator of observed frequencies for all bins, regardless if we use training methods or not. This is important in a clinical setting so that doctors can get an indication of how reliable a specific seizure detection is. The non-seizure predictions (blue) are under-confident in both cases, and we notice that using the training methods improves the calibration for them to a larger extent.

Interestingly, employing mixup and segment translation in training 4x-conv models did not reduce SCE, despite increased AUC scores (table 3.7). Ensembles also failed to reduce SCE. However, these methods seemed effective with the 7x-conv model, suggesting that model capacity or structure may influence their effectiveness.

The calibration of the Inception+attention model was worse than the Borovac model when trained using segment translation and mixup, even though the Inception+attention model consistently produced higher AUC scores. However, temperature scaling, when combined with mixup, segment translation, and ensembles, reduced SCE in the Inception+attention model (figure 3.5). This led to comparable calibration results to the Borovac model, with SCE reaching 3.13. This demonstrates that temperature scaling can improve calibration, even when used with mixup, segment translation and ensembles.

In our study, we observe that mixup contributes to a lower static calibration error (SCE) when applied to models with sufficient capacity or appropriate structure. The paper by Thulasidasan [45] suggests that mixup not only enhances the generalization performance of deep neural networks but also leads to better calibration. Our results corroborate these findings, as mixup both yields a lower SCE (3.72%) and produces higher AUC values.

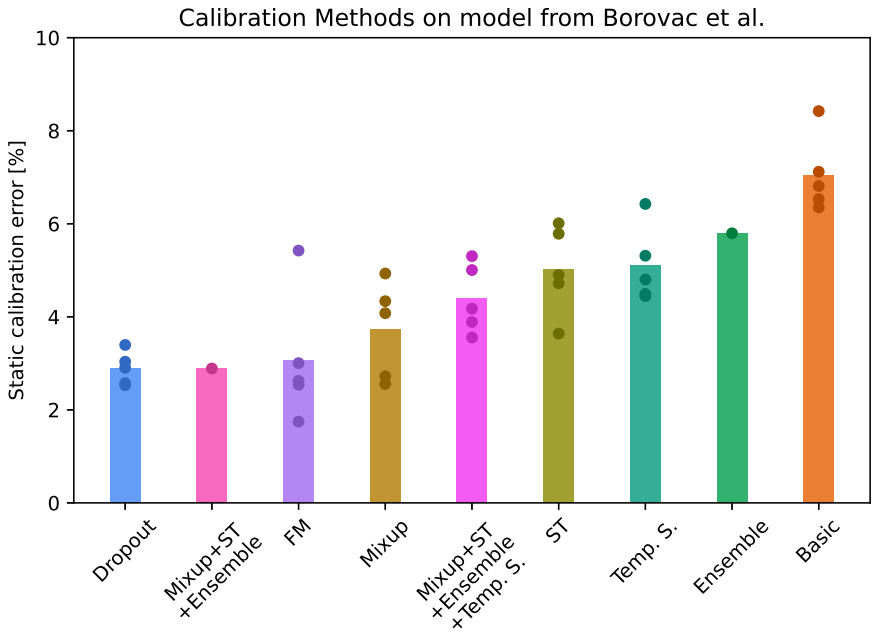


Figure 3.4: Model from Borovac et al. trained with different training methods. Average static calibration error across 5 models (except ensemble). Each dot represents a single model. Abbreviations: Feature-level Mixup (FM), Manifold Mixup (MM), Segment Translation (ST), Temperature Scaling (Temp. S.).

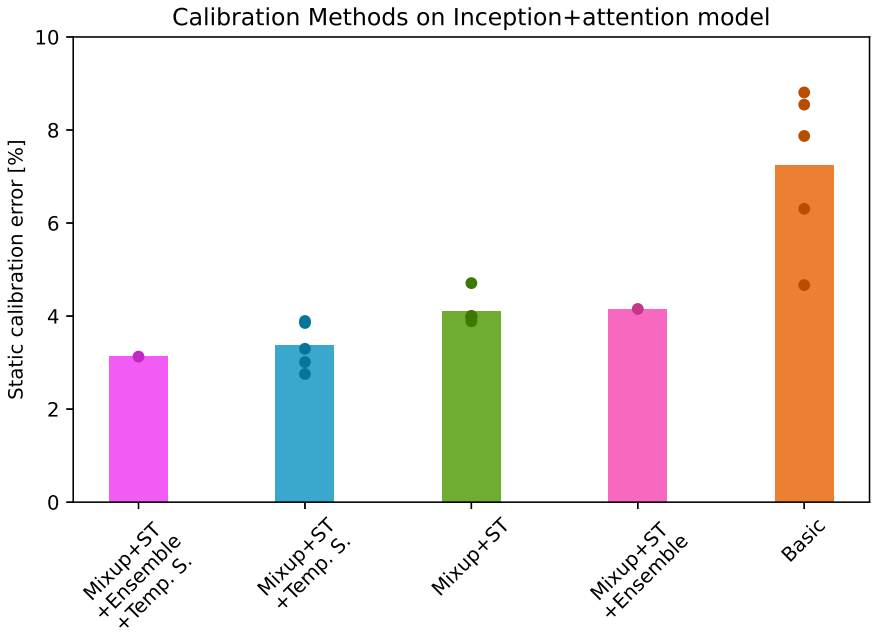


Figure 3.5: Inception+attention models trained with different training methods. Average static calibration error across 5 models (except ensemble). Each dot represents a single model. Abbreviations: Segment Translation (ST), Temperature Scaling (Temp. S.).

4. Conclusion

This study investigated the potential of incorporating architectural designs and training strategies from other domains to improve the performance of EEG seizure detection models. Our findings suggest that the benefits of increasing architectural complexity are small when training on this data, unless some of the training strategies described in section 3.2 are used. A more effective approach may be to focus on additional training strategies which can lead to substantial improvements in model performance, as demonstrated here.

The results reveal that mixup, manifold mixup, feature mixup, segment translation, and ensemble methods all improve the performance of the network, as measured by the AUC. Among these methods, a combination of mixup, segment translation, and ensemble methods demonstrates the largest improvement in performance, producing segment-based AUC of 0.981 and patient-based AUC score of 0.965. Furthermore, this combination lead to the highest F1 scores and the lowest static calibration error (SCE). Notably, the efficacy of these combined methods depends on the model's capacity or structure, which emphasizes the importance of considering both architecture and training strategies simultaneously.

To investigate whether the attention mechanism of Isayev et al. [28], has a significant effect on model accuracy, as measured by the AUC, we compared the model from Borovac et al., which combines the Cork feature extractor with attention, with another model where the Cork feature extractor is combined with max pooling. When training with the standard procedure, the results showed no significant difference between the two approaches. However, when we further compared the models employing mixup and segment translation, the attention-based models outperformed their max pooling counterparts.

When comparing the 4x-conv model, 7x-conv model and the model from Borovac et al., using both mixup and segment translation, we found the simpler 4x-conv model superior to the 7x-conv model. However, the Borovac model, which employs an attention mechanism, outperformed the 4x-conv model when trained with the same methods. This result, demonstrates the efficacy of attention mechanisms.

Introducing inception blocks to the model from Borovac et al. resulted in substantial performance improvements when trained with mixup and segment translation compared to the Borovac model. This Inception+attention model demonstrated higher AUC and F1 scores, while having very small variance between trials. The inception block's capacity to learn features at multiple time scales, could partially explain this improvement. Although the other modifications to the feature extractor did not enhance performance, applying ensemble methods universally improved outcomes, with the Inception ensemble model

delivering the best results, producing a segment-based AUC of 0.982 and a patient-based AUC of 0.969.

In addition to the previously discussed findings, our study also examined the methods for calibration, focusing on the SCE due to class imbalance in the data. We found that all training and calibration methods, except for normalization, improved the SCE, in comparison to the basic training procedure, when applied to the model from Borovac et al.. Among these methods, we observed that a combination of mixup, segment translation, and ensemble was the most efficient in decreasing the SCE. Furthermore, we found that incorporating temperature scaling into this combination could yield even greater reductions in SCE, as demonstrated with the Inception+attention model. The reliability diagram showed that these models produce meaningful probabilities, with the seizure predictions outputs closely resembling the observed frequencies in the test data.

In summary, this study underscores the importance of incorporating training methods and advanced architectural designs in the improvement of EEG-based seizure detection models. Striking a balance between the complexity and design of the architecture and the application of these training methods is essential for optimizing model performance. Additionally, the work demonstrates how these methods can improve the calibration of the models. These findings contribute to the ongoing efforts to improve automatic methods for analysing seizures in EEG recordings, ultimately aiming to enhance medical diagnosis and treatment for those affected by epilepsy.

4.1. Directions for future work

In this work, we used 16 seconds of EEG data as input to our models. However, in clinical practice, neurologists analyze EEG recordings in a much wider context. An approach for enhancing model performance could involve expanding the input data by extracting features from multiple adjacent segments and then classifying a specific segment. This could be accomplished by stacking three adjacent segments, for instance, and then classifying segment in the middle. The classification could involve attention-based methods and/or a fully connected layer. Employing RNNs could prove useful here.

Segments containing the onset and offset of seizure activity were not used in this study. However, identifying the onset of a seizure is an important aspect of seizure detection. Future studies could focus on identifying the onset more accurately, incorporating those segments.

We observed here that several patients were particularly challenging to classify, resulting in a lower patient-based AUC. Another interesting study would be to incorporate EEG artifacts into the training process as hard-negatives. Incorporating and prioritizing hard-negative samples could increase the performance on these difficult patients and potentially make the model more robust.

Another intriguing approach to improving performance involves the use of contrastive

learning [46], which could be applied in both supervised and unsupervised settings with access to unlabeled data. This could refine the model's ability to differentiate between similar segments, improving the distinction between noise and seizure activity for example.

Lastly, investigating different augmentation methods for EEG data could yield more benefits. Augmentation methods used in image processing are not directly applicable here and hence, it could be beneficial to develop and experiment with novel augmentation strategies that are specific to EEG data.

References

[1] H. M. de Boer, M. Mula, and J. W. Sander, "The global burden and stigma of epilepsy," *Epilepsy behavior : EB*, vol. 12, no. 4, pp. 540–546, 2008. DOI: 10.1016/j.yebeh.2007.12.019.

[2] R. S. Fisher, C. Acevedo, A. Arzimanoglou, *et al.*, "Ilae official report: A practical clinical definition of epilepsy," *Epilepsia*, vol. 55, no. 4, pp. 475–482, 2014. DOI: 10.1111/epi.12550.

[3] D. Schmidt and S. C. Schachter, "Drug treatment of epilepsy in adults," *BMJ*, vol. 348, 2014. DOI: 10.1136/bmj.g254. eprint: https://www.bmj.com/content/348/bmj.g254.full.pdf.

[4] G. Liu, N. Slater, and A. Perkins, "Epilepsy: Treatment options," *American family physician*, vol. 96, no. 2, pp. 87–96, 2017.

[5] N. S. Abend, D. J. Dlugos, C. D. Hahn, L. J. Hirsch, and S. T. Herman, "Use of eeg monitoring and management of non-convulsive seizures in critically ill patients: A survey of neurologists," *Neurocritical care*, vol. 12, pp. 382–389, 2010.

[6] A. T. Tzallas, M. G. Tsipouras, and D. I. Fotiadis, "Epileptic seizure detection in eegs using time–frequency analysis," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, no. 5, pp. 703–710, 2009. DOI: 10.1109/TITB.2009.2017939.

[7] S. Sharma, M. Nunes, and A. Alkhachroum, "Adult critical care electroencephalography monitoring for seizures: A narrative review," *Frontiers in Neurology*, vol. 13, 2022.

[8] T. M. Ganguly, C. A. Ellis, D. Tu, *et al.*, "Seizure detection in continuous inpatient eeg: A comparison of human vs automated review," *Neurology*, vol. 98, no. 22, e2224–e2232, 2022.

[9] V. Shah, E. von Weltin, S. Lopez, *et al.*, "The temple university hospital seizure detection corpus," *Frontiers in Neuroinformatics*, vol. 12, p. 83, 2018. DOI: 10.3389/fninf.2018.00083.

[10] A. L. Goldberger, L. A. Amaral, L. Glass, *et al.*, "Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, e215–e220, 2000.

[11] U. of Helsinki, *Helsinki eeg corpus*.

[12] P. Detti, "Siena scalp eeg database," version 1.0.0, *PhysioNet*, 2020.

[13] A. Shoeibi, N. Ghassemi, R. Alizadehsani, *et al.*, "A comprehensive comparison of handcrafted features and convolutional autoencoders for epileptic seizures detection in eeg signals," *Expert Systems with Applications*, vol. 163, p. 113 788, 2021.

[14] 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. 051 001, 2019.

[15] M. Zhou, C. Tian, R. Cao, *et al.*, "Epileptic seizure detection based on eeg signals and cnn," *Frontiers in neuroinformatics*, vol. 12, p. 95, 2018.

[16] A. M. Abdelhameed, H. G. Daoud, and M. Bayoumi, "Deep convolutional bidirectional lstm recurrent neural network for epileptic seizure detection," in *2018 16th IEEE International New Circuits and Systems Conference (NEWCAS)*, IEEE, 2018, pp. 139–143.

[17] L. Vidyaratne, A. Glandon, M. Alam, and K. M. Iftekharuddin, "Deep recurrent neural network for seizure detection," in *2016 International Joint Conference on Neural Networks (IJCNN)*, IEEE, 2016, pp. 1202–1207.

[18] K. Indiradevi, E. Elias, P. Sathidevi, S. D. Nayak, and K. Radhakrishnan, "A multilevel wavelet approach for automatic detection of epileptic spikes in the electroencephalogram," *Computers in biology and medicine*, vol. 38, no. 7, pp. 805–816, 2008.

[19] H. Adeli, S. Ghosh-Dastidar, and N. Dadmehr, "A wavelet-chaos methodology for analysis of eegs and eeg subbands to detect seizure and epilepsy," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 2, pp. 205–211, 2007.

[20] I. Wijayanto, R. Hartanto, H. A. Nugroho, and B. Winduratna, "Seizure type detection in epileptic eeg signal using empirical mode decomposition and support vector machine," in *2019 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2019, pp. 314–319. DOI: 10.1109/ISITIA.2019.8937205.

[21] G. Cisotto, A. Zanga, J. Chlebus, I. Zoppis, S. Manzoni, and U. Markowska-Kaczmar, "Comparison of attention-based deep learning models for eeg classification," *arXiv preprint arXiv:2012.01074*, 2020.

[22] A. Jafari, S. Gandhi, S. H. Konuru, W. David Hairston, T. Oates, and T. Mohsenin, "An eeg artifact identification embedded system using ica and multi-instance learning," in *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2017, pp. 1–4. DOI: 10.1109/ISCAS.2017.8050346.

[23] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *International conference on machine learning*, PMLR, 2017, pp. 1321–1330.

[24] S. Thulasidasan, G. Chennupati, J. A. Bilmes, T. Bhattacharya, and S. Michalak, "On mixup training: Improved calibration and predictive uncertainty for deep neural networks," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[25] A. Borovac, T. Runarsson, G. Thorvardsson, and S. Gudmundsson, "Calibration of automatic seizure detection algorithms," in *2022 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, IEEE, 2022, pp. 1–6.

[26] A. Borovac, S. Gudmundsson, G. Thorvardsson, *et al.*, "Ensemble learning using individual neonatal data for seizure detection," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 10, pp. 1–11, 2022. DOI: 10.1109/jtehm.2022.3201167.

[27] A. O'Shea, G. Lightbody, G. Boylan, and A. Temko, "Investigating the impact of cnn depth on neonatal seizure detection performance," in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2018, pp. 5862–5865. DOI: 10.1109/EMBC.2018.8513617.

[28] D. Y. Isaev, D. Tchapyjnikov, C. M. Cotten, *et al.*, "Attention-based network for weak labels in neonatal seizure detection," *Proceedings of machine learning research*, vol. 126, p. 479, 2020.

[29] M. Ilse, J. M. Tomczak, and M. Welling, *Attention-based deep multiple instance learning*, 2018. arXiv: 1802.04712 [cs.LG].

[30] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[31] C. Szegedy, W. Liu, Y. Jia, *et al.*, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1–9.

[32] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[33] V. Verma, A. Lamb, C. Beckham, *et al.*, "Manifold mixup: Better representations by interpolating hidden states," in *Proceedings of the 36th International Conference on Machine Learning*, K. Chaudhuri and R. Salakhutdinov, Eds., ser. Proceedings of Machine Learning Research, vol. 97, PMLR, Sep. 2019, pp. 6438–6447.

[34] M. Xu, J. Zhang, B. Ni, *et al.*, "Adversarial domain adaptation with domain mixup," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 04, pp. 6502–6509, Apr. 2020. DOI: 10.1609/aaai.v34i04.6123.

[35] T. G. Dietterich, "Ensemble methods in machine learning," in *Multiple Classifier Systems*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2000, pp. 1–15, ISBN: 978-3-540-45014-6.

[36] C. Ju, A. Bibaut, and M. van der Laan, "The relative performance of ensemble methods with deep convolutional neural networks for image classification," *Journal of Applied Statistics*, vol. 45, no. 15, pp. 2800–2818, 2018.

[37] 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.

[38] G. Xu, T. Ren, Y. Chen, and W. Che, "A one-dimensional cnn-lstm model for epileptic seizure recognition using eeg signal analysis," *Frontiers in Neuroscience*, vol. 14, 2020, ISSN: 1662-453X. DOI: 10.3389/fnins.2020.578126.

[39] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," *Advances in neural information processing systems*, vol. 30, 2017.

[40] S. Cherukuvada and R. Kayalvizhi, "A review on eeg based epileptic seizures detection using deep learning techniques," in *2022 4th International Conference on Smart Systems and Inventive Technology (ICSSIT)*, IEEE, 2022, pp. 966–973.

[41] M. P. Naeini, G. Cooper, and M. Hauskrecht, "Obtaining well calibrated probabilities using bayesian binning," in *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[42] J. Nixon, M. W. Dusenberry, L. Zhang, G. Jerfel, and D. Tran, "Measuring Calibration in Deep Learning.," in *CVPR Workshops*, vol. 2, 2019.

[43] K. Lee, H. Jeong, S. Kim, D. Yang, H.-C. Kang, and E. Choi, "Real-Time Seizure Detection using EEG: A Comprehensive Comparison of Recent Approaches under a Realistic Setting," *arXiv preprint arXiv:2201.08780*, 2022.

[44] A. Einizade, M. Mozafari, S. H. Sardouie, S. Nasiri, and G. Clifford, "A deep learning-based method for automatic detection of epileptic seizure in a dataset with both generalized and focal seizure types," in *2020 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, 2020, pp. 1–6. DOI: 10.1109/SPMB50085.2020.9353629.

[45] S. Thulasidasan, G. Chennupati, J. Bilmes, T. Bhattacharya, and S. Michalak, *On mixup training: Improved calibration and predictive uncertainty for deep neural networks*, 2020. arXiv: 1905.11001 [stat.ML].

[46] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *International conference on machine learning*, PMLR, 2020, pp. 1597–1607.

A. Appendix

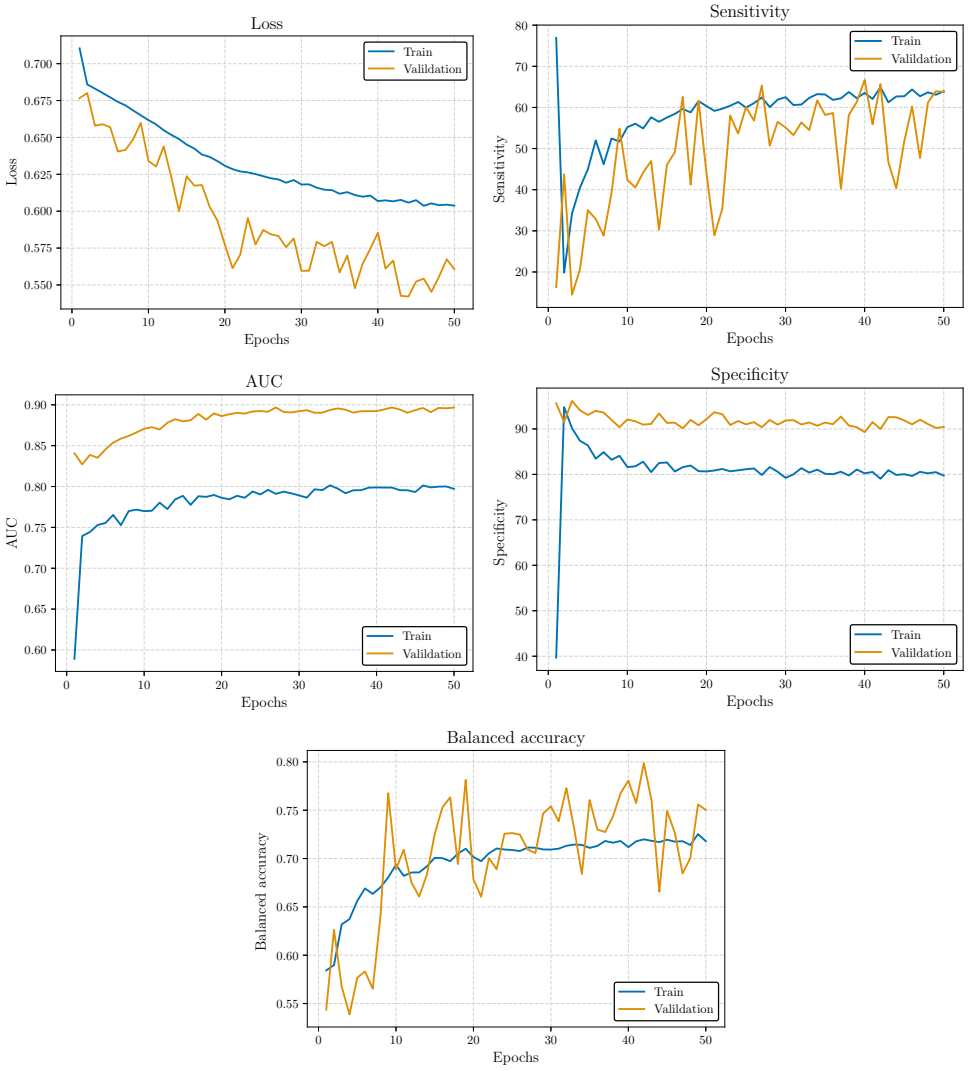


Figure A.1: Training process for the Baseline model (one trial). Basic training procedure used.

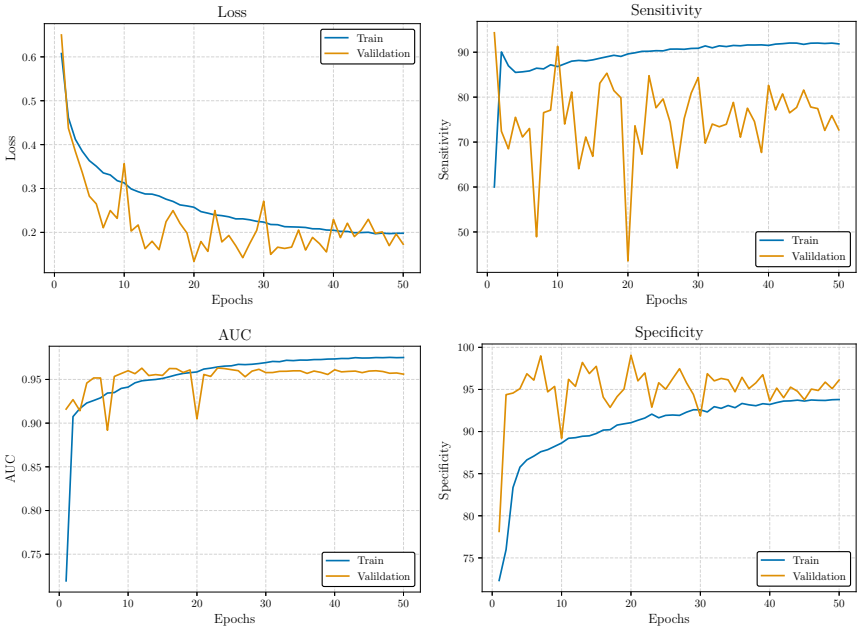


Figure A.2: 4x conv model training process (one trial). Basic training procedure used.

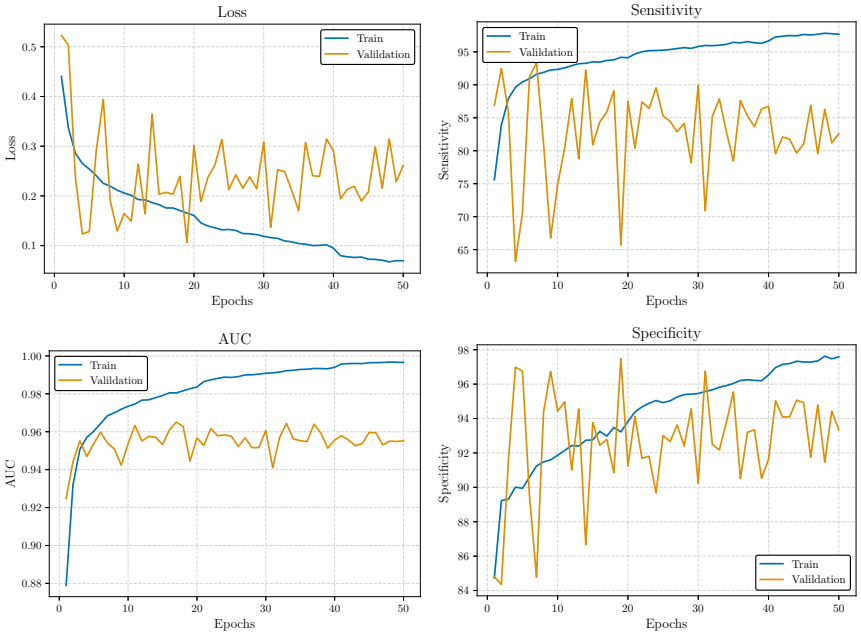


Figure A.3: Cork+attention conv model training process (one trial). Basic training procedure used.

	cork+attention (mixup+ST_b)	cork+attention (mixup+ST_b+ensemble)
Patient-based (patients with seizures)		
Area under the curve	0.948 (0.011)	0.964
Sensitivity [%]	74.47 (4.92)	73.43
Specificity [%]	94.97 (2.82)	97.08
F1-score [%]	59.26 (7.21)	67.34
Segment-based (patients with seizures)		
Area under the curve	0.969 (0.006)	0.981
Sensitivity [%]	85.12 (3.06)	86.66
Specificity [%]	94.9 (3.25)	97.39
F1-score [%]	69.24 (8.25)	78.65
Expected calibration error [%]	4.5 (2.02)	5.22
Overconfidence error [%]	0.0 (0.0)	0.0
Static calibration error [%]	4.13 (0.86)	3.64
Brier score	0.05 (0.02)	0.03
Negative log likelihood	0.17 (0.07)	0.13

Table A.1: Segment translation with event independent sampling (5 trials). ST_b: segment translation where all seizure and non-seizure events in recordings are equally likely to be sampled from. With this the length of seizure events is irrelevant to how often it is sampled. This reveled no difference compared to using the standard segment translation.

	FL	ST+FL	mixup+ST+FL	ST+FL +ensemble	mixup+ST+FL +ensemble
Patient-based (patients with seizures)					
Area under the curve	0.935 (0.015)	0.946 (0.005)	0.931 (0.016)	0.958	0.948
Sensitivity [%]	75.25 (3.7)	73.17 (2.84)	71.82 (4.59)	74.63	72.36
Specificity [%]	95.95 (1.09)	96.69 (0.69)	95.55 (1.06)	97.75	96.57
F1-score [%]	63.6 (1.41)	62.93 (2.07)	60.96 (1.13)	67.97	65.64
Segment-based (patients with seizures)					
Area under the curve	0.966 (0.002)	0.967 (0.003)	0.967 (0.004)	0.976	0.975
Sensitivity [%]	83.03 (3.28)	82.02 (1.69)	83.85 (3.77)	83.75	84.36
Specificity [%]	96.26 (1.11)	96.84 (0.61)	95.89 (1.09)	97.91	97.04
F1-score [%]	71.96 (2.89)	73.65 (1.84)	70.92 (2.31)	79.47	75.79
Expected calibration error [%]	8.04 (1.17)	1.01 (0.38)	3.8 (2.42)	1.46	3.83
Overconfidence error [%]	0.0 (0.0)	0.06 (0.06)	0.03 (0.04)	0.0	0.0
Static calibration error [%]	6.07 (1.15)	4.13 (0.52)	3.84 (1.5)	3.99	3.3
Brier score	0.04 (0.01)	0.03 (0.0)	0.04 (0.01)	0.02	0.03
Negative log likelihood	0.18 (0.02)	0.12 (0.01)	0.16 (0.03)	0.09	0.13

Table A.2: Experiments with focal loss on model from Borovac et al. (5 trials). FL: Focal loss, ST: Segment translation.