EventNet: Detecting Events in EEG

Nick Seeuws, Maarten De Vos, and Alexander Bertrand

***Abstract*— Neurologists are often looking for various *events of interest* when analyzing EEG. To support them in this task various machine-learning-based algorithms have been developed. Most of these algorithms treat the problem as classification, thereby independently processing signal segments and ignoring temporal dependencies inherent to events of varying duration. At inference time, the predicted labels for each segment then have to be postprocessed to detect the actual events. We propose an end-to-end event detection approach (EventNet), based on deep learning, that directly works with events as learning targets, stepping away from ad-hoc postprocessing schemes to turn model outputs into events. We compare EventNet with a state-of-the-art approach for artefact and epileptic seizure detection, two event types with highly variable durations. EventNet shows improved performance in detecting both event types. These results show the power of treating events as direct learning targets, instead of using ad-hoc postprocessing to obtain them. Our event detection framework can easily be extended to other event detection problems in signal processing, since the deep learning backbone does not depend on any task-specific features.**

***Index Terms*— Electroencephalography (EEG), Deep Learning, Neural Networks, Artefact Detection, Seizure Detection**

I. Introduction

Electroencephalography (EEG) provides an invaluable tool for human brain monitoring. Neurologists make use of EEG (among others) in the treatment of epilepsy and sleep disorders. Electrodes on a patient's scalp measure electric potential at different locations. The combination of these different potentials gives the full EEG signal. Neurologists can then use these potential signals to infer information about the brain itself [1].

Various specific tasks in EEG processing focus on *event detection*. Neurologists look for specific events of interest and often annotate the events' start and stop time in a long recording. Epileptic seizure detection is such a task where neurologists are interested in finding seizures in long EEG recordings. Analyzing these seizures is an important step in diagnosis and treatment of epilepsy. In sleep monitoring there are also such events of interest, like arousals or spindles. Next to these direct areas of interest for the health professionals, there is also artefact detection. Various sources from inside or outside the body can disturb EEG, obfuscating patterns of interest and making automated processing more difficult. Current approaches to automated processing of EEG are very sensitive to artefacts, raising false alarms in the case of seizure detection, for example. Therefore, a common preprocessing task in EEG processing is the detection and removal of such artefacts events.

Automating event detection with machine learning is not straightforward. Classical machine learning tasks are posed as classification or regression problems for signal segments, i.e, an input of fixed dimension, where the goal is to learn a function that maps this fixed-size input to a specific target. Common classification and regression approaches are not set up to deal with events appearing as (t_{start}, t_{stop}) tuples. Events of variable duration will show some specific structure or patterns across (or within) signal segments. Independently classifying signal segments ignores this temporal dependency inherent to event detection, and time series processing in general.

Previous methods did frame the event detection problem for EEG as classification [2]–[6]. All these methods cut an EEG recording into segments of a fixed, predetermined duration and classified these different segments independently. Deciding on the segment duration is then a non-trivial design decision, which can have large impact. One wants to give a classifier a long context to base its decision on, but that will limit the temporal resolution of the model (only one label can be predicted for the entire segment). Improving the temporal resolution also means to shorten the input, limiting the context a model can work with. High variability in event duration makes finding a good segment duration even more difficult. For example, in seizure detection event durations can span up to two orders of magnitude. With long segments, the shorter events might not stand out enough for a classifier, and with short segments the classifier might be blind to the (possibly time-varying) characteristics of longer events. Deciding on a proper segment duration for a given problem also requires domain expertise, and knowledge about the specific use case. Additionally, a classification model cannot tell a user where an event starts and stops in a straightforward manner. In some cases, an ad hoc postprocessing method is added in order to combine subsequent segments to determine in what segment a full event starts and stops. Nevertheless, such postprocessing methods will always be limited in temporal resolution due to the segment size.

Inspired by object detection in computer vision, some recent

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 802895) and from the Flemish Government (AI Research Program).

N. Seeuws , A. Bertrand and M. De Vos are with the Dept. of Electrical Engineering (ESAT), Stadius Center for Dynamical Systems, Signal Processing and Data Analytics (STADIUS), KU Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

M. De Vos is also with the Dept. of Development and Regeneration, Faculty of Medicine, KU Leuven

A. Bertrand and N. Seeuws are affiliated to Leuven.AI - KU Leuven institute for AI, B-3000, Leuven, Belgium.

methods in EEG-based sleep event detection took first steps to move away from this context-resolution tradeoff. They directly detect EEG events [7], [8] based on the YOLO object detector [9]. They define *default events* for a fixed-duration input, at various predefined points in time and for a fixed duration. The network predicts whether an input signal contains an event linked to a specific *default event*, and how much said event's duration deviates from the *default event*'s fixed duration. Both works only allow for small deviations from the *default event* duration. Additionally, both approaches can only process fixed-duration inputs.

Other recent methods in seizure [10] and sleep arousal detection [11] cope with the context-resolution tradeoff *and* are able to tackle large variability in duration. Inspired by image segmentation the authors of [10], [11] adapted the U-Net architecture towards the event detection task. For a long recording as input, the U-Nets predict, for every time sample in the input, whether the sample is part of the event of interest (e.g., a seizure or a sleep arousal). While the U-Nets look at a large context, the networks are still able to predict at a fine temporal resolution (at the signal's sampling frequency). This sample-based classification approach allows both methods to naturally cope with a large variability in event durations. However, while this U-net-inspired approach circumvents the context-resolution tradeoff and is able to deal with duration variability, it does not fully model events due to the time-sample-based classification, which can easily lead to false alarms or gaps in a longer event. In [10], for example, heavy postprocessing is used to avoid false alarms (the U-net can easily predict a few false-positive time steps) and double detections (due to gaps in longer events). Designing a postprocessing algorithm is an important step in using a U-Net for event detection and, as shown in [10], is far from a trivial task.

In this work, we greatly simplify the event detection problem. Inspired by works in visual object detection, our method encodes events using the events' center and duration. Both are predicted jointly by a single convolutional neural network (ConvNet), which we call EventNet. Encoding training events simply involves mapping the different events to an *event center* and *duration* signal, after which EventNet can be trained end-to-end without postprocessing. Crucially, we can easily cope with large variability in event duration.

To summarize, our key contributions are as follows:

- An event detection network, EventNet, that can be trained end-to-end
- EventNet can cope with a large variability in event duration
- Detecting events with EventNet does not require postprocessing

Section II introduces EventNet and discusses our experiments. Section III shows results of these experiments. Section IV discusses these results and explains benefits and drawbacks of using an event-based framework like EventNet. Section V concludes the paper.

II. Methods

A. EventNet

In this subsection, we describe the EventNet conceptually, and we refer to Figure 1 for a schematic illustration of the different concepts introduced in this subsection.

1) Encoding and decoding events: Events are represented by their center point and duration in EventNet. A complete EEG recording is input into the model, after which center and duration predictions are produced across the length of the recording in a single pass through the model. Both outputs (i.e., center and durations) are treated as "signals", in the sense that they span the entire input. This is similar to the way the U-Nets of [10], [11] are used, where the models predict at each point in time whether that point is part of an event. In the case of EventNet, whether a point in time corresponds to the center of an event is indicated by the *center signal*. The *duration signal* is used to represent an event's duration *if that point in time would be a center point*. Note that this signal is meaningless at time points far away from a center point. Our approach is inspired by CenterNet [12] for object detection in images. This image object detection model predicts centers of detected objects, and predicts object sizes at those specific centers.

At inference time, EventNet's outputs are decoded by searching for the peaks in the center signal. The detected event centers are then represented by the different peaks, and EventNet's confidence is displayed by the specific signal values at these peaks (on a relative scale, these values do not necessarily represent detection probabilities like they do for classification problems.) For each detected center, the predicted event duration can then be found in the duration signal at that specific point.

2) Losses:

a) Training targets: To train EventNet's center and duration prediction, training targets need to be defined. Center prediction is treated as a sample-based classification task, similar to the U-Nets of [10], [11]. In constrast to a U-Net-based approach, some slack should be allowed on the center target. Predicting a center that is just a few samples off-target is better than, e.g., predicting a false event in an hour of background signal, and thus should be penalized less. The weighting method of [12] for object detection in images is modified and applied to event center prediction. For an event with ground-truth center t^*, the target center signal is defined as

$$c(t) = \exp(-\frac{(t-t^*)^2}{2\sigma^2}) \tag{1}$$

with σ depending on the target event's duration. Following [13], the hyperparameters are set as $\sigma = \alpha d/6$, with $\alpha = 0.5$ and d the event's actual duration, measured in terms of *time points*. In the case of multiple events in a signal, these center target signals as in Eq.1 are defined for each event independently, and are combined by taking the maximum target value at every time point.

The duration targets only need to be defined at the target centers t^*, since only duration predictions at ground-truth center points t^* will be considered in the duration loss.

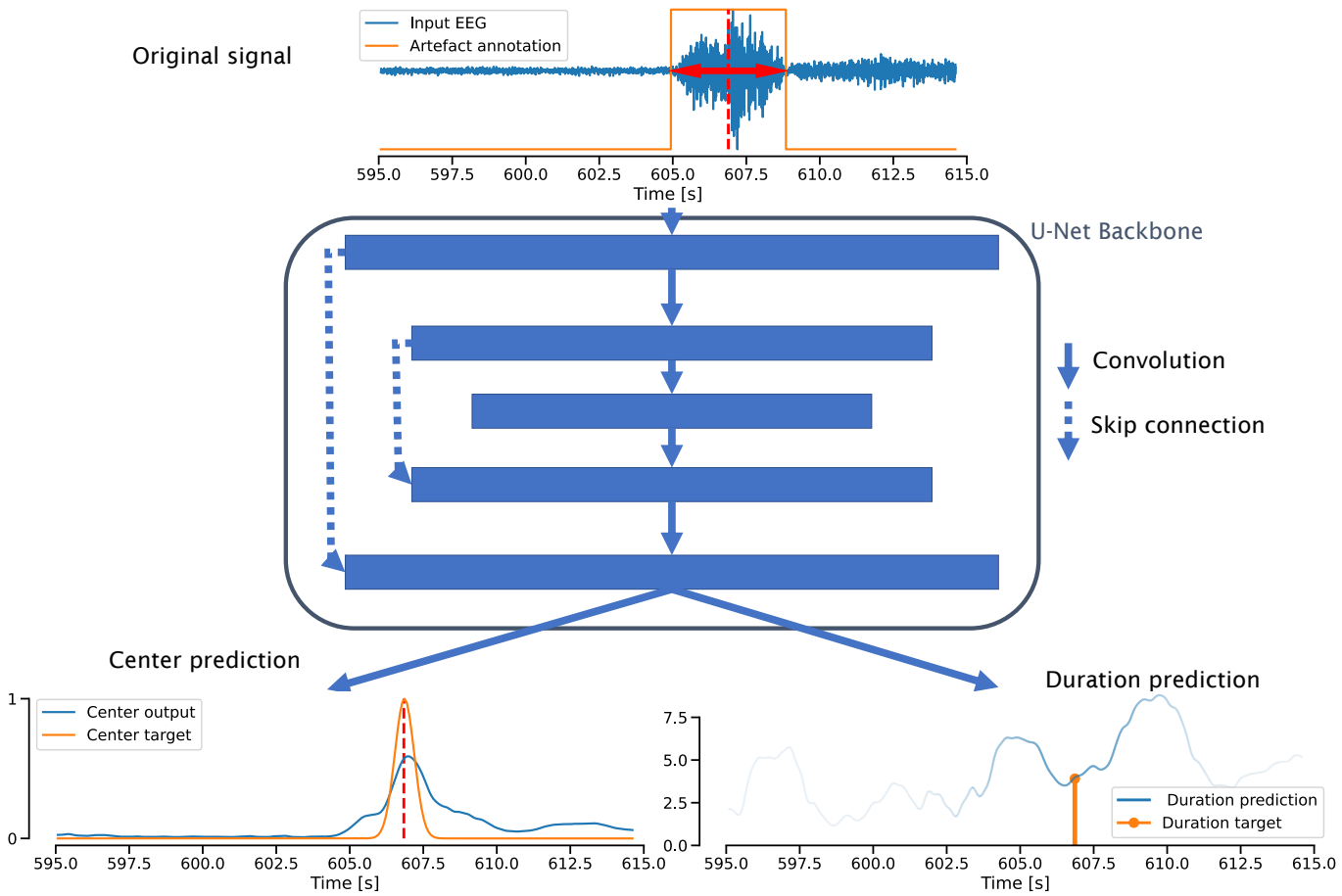


Fig. 1: EventNet overview. The input EEG signal at the top contains a single artefact event, annotated in orange (the event's center and duration are highlighted in red). The U-Net-like backbone combines local and global information using an encoder-decoder structure and skip connections. Two independent convolutional layers output the *center* and *duration* signals (in blue). The training targets for both signals are plotted in orange. Training the *center* signal involves comparing it to the entire target signal. The *duration* signal is only trained and evaluated at event centers. Note that this example only spans 20 s and contains a single event. EventNet can process longer inputs and detect more events at once.

EventNet works with a maximum duration it can predict. This can either be set to the maximum duration in a given data set, or determined with expert information (*"what can reasonably be expected as an upper bound for these events?"*). The duration predictions, similar to the center predictions, are constrained to the range $[0, 1]$ (the target durations are divided by the predefined maximum duration.) For every target event in a data set, the duration signal value at the event's center point is set to the event's normalized duration.

b) Training losses: Center prediction is trained using focal loss [14], in the modified form of [15]. The full center signal prediction loss L_c for an input signal containing N events, EventNet center prediction $c'(t)$, and target center signal $c(t)$ is defined as:

$$L_c = -\frac{1}{N}\sum_t \begin{cases} (1-a_c)(1-c')^\alpha \log(c') & \text{if } c = 1 \\ a_c(1-c)^\beta c'^\alpha \log(1-c') & \text{otherwise} \end{cases} \quad (2)$$

(with the dependence on t of $c'(t)$ and $c(t)$ dropped for legibility). Hyperparameters are set as $a_c = 0.1$, $\alpha = 2$ and $\beta = 4$ following [14], [15]. The original focal loss is a classification loss where mistakes are adaptively penalized based on the model's confidence using the factors exponentiated with α. With the modification of [15], false alarms close in time to the target center t^* are penalized less than false alarms further away (using the factor $(1-c)^\beta$ and the exponential in Eq 1). [1]

Duration predictions are trained using *Intersection over Union* (IoU) as loss. IoU is a popular loss formulation in object detection [13]. Crucially, IoU is based on *relative* duration errors, ensuring that the batch loss will not be dominated by long events. Calculating the intersection and union of predicted and target events can be simplified due to the use of the target event's center. The duration loss L_d, for N events, set of known target points $\mathcal{T}$, predicted duration signal $d'(t)$ and

[1] This form of the focal loss can cause numerical instabilities when performing gradient descent during training. To avoid instabilities, one needs to rewrite the loss in terms of the logits instead of sigmoid outputs. Refer to Appendix B for more details.

target $d(t)$, is formulated as

$$L_d = \frac{1}{N} \sum_{t^* \in \mathcal{T}} \frac{\min[d'(t^*), d(t^*)]}{\max[d'(t^*), d(t^*)]}. \quad (3)$$

Note that the predicted duration signal $d'(t)$ is only evaluated at the center points, i.e., the value of $d(t)$ and $d'(t)$ has no meaning at points which are not treated as center points, even though the network will produce an output $d'(t)$ for every time point t.

The center prediction loss L_c and duration loss L_d are combined into the full loss L as a weighted sum,

$$L = L_c + \lambda_d L_d$$

, where λ_d is a hyper-parameter to control the relative influence of the two tasks. For both of our data sets, the two loss terms are approximately equal in magnitude by setting $\lambda_d = 5$. This value can be raised or lowered to increase or decrease, respectively, the influence of the duration prediction task.

3) Neural backbone: An input signal is transformed into the predictions $c'(t)$ and $d'(t)$ using a ConvNet. To ensure EventNet can process a large context while also being attuned to small-scale details, a U-Net-like *backbone* is used to extract a *feature signal*. Global information of the input signal is captured by the encoder-decoder structure of this backbone. Local information is propagated through the network using skip-connections, allowing global and local information to be combined in the backbone network. The extracted feature signal is mapped to the actual prediction signals $c'(t)$ and $d'(t)$ by two independent convolutional layers (one for each). A sigmoid activation function is used in both of these final layers to ensure an output range of $[0, 1]$.

A schematic overview of the full model can be found in Figure 1. Due to differences in scale of the target events in our data sets, which are discussed below, the specific implementations of EventNet and its U-Net backbone are tailored to the specific data sets. Details can be found in Appendix A. For both data sets, the Adam optimizer is used with an exponentially decaying learning rate over 100 epochs. All network weights are initialized randomly, except for the center prediction's bias term. Following [14], this value is set so the prior probability of the background is reflected in the initialization (making the initial gradient updates more stable). The final network weights are the ones with lowest validation loss over the entire training process.

Our model implementation and experiment code can be found at https://github.com/N1xu5/EventNet.

B. Data

We evaluated EventNet on two data sets.

1) Artefacts: The Temple University Artefact Corpus (TUAR) consists of various EEG artefact events, described in [16]. In the full data set, many multi-channel EEG recordings with channel-level annotations of artefact events are present. In this paper, the muscle and chewing artefacts are the events of interest due to their large variability in length. Both types are joined into a single *artefact* class. EventNet is trained on the individual channels of this data set to predict the channel-level artefact annotations. A lognormal-like distribution is observed for the duration of the events in the original data set. This means that EventNet would not have enough training examples for the longer events compared to shorter events. Therefore, a cutoff at $10\,\mathrm{s}$ is introduced to make the event detection problem more feasible for this data set. The cutoff is used at the channel level, i.e., only the specific channels containing events longer than $10\,\mathrm{s}$ are dropped. In total, a data set is constructed with 3252 single-channel EEG recordings, spanning 925 hours and containing 11684 events in total. The recordings are divided into training, validation, and test sets, making sure that recordings of the same individual are not split among the sets. All signals are resampled to $200\,\mathrm{Hz}$.

2) Seizures: The second data set is the Temple University Seizure Corpus (TUSZ) containing epileptic seizures [17]. This is the same data set as used in [10]. The data set is made up of multi-channel EEG recordings, with epileptic seizures annotated at a general level (only indicating at which points in time a seizure occurs, not on which channel(s)). For seizures, EventNet is trained to detect these "general" cross-channel events using a multi-channel input. Similar to TUAR, fewer examples for longer event durations are present in TUSZ. However, to ensure a fair comparison to the approach of [10], the entire data set is used. A maximum duration for EventNet predictions is set on $250\,\mathrm{s}$. Longer durations cannot be predicted by EventNet, but EventNet will still be evaluated on the full data set containing longer events.

The same preprocessing approach as [10] is used, producing 5612 18-channel recordings sampled at $200\,\mathrm{Hz}$, containing 3050 seizure events in total. Since labels for the final test set of [10] are not available, its validation set is used as our test set. The training set of [10] is randomly split into our training and validation set. No patient-specific information is leaked to the test set. However, recordings of the same individual might be present in both our training set and in our validation set.

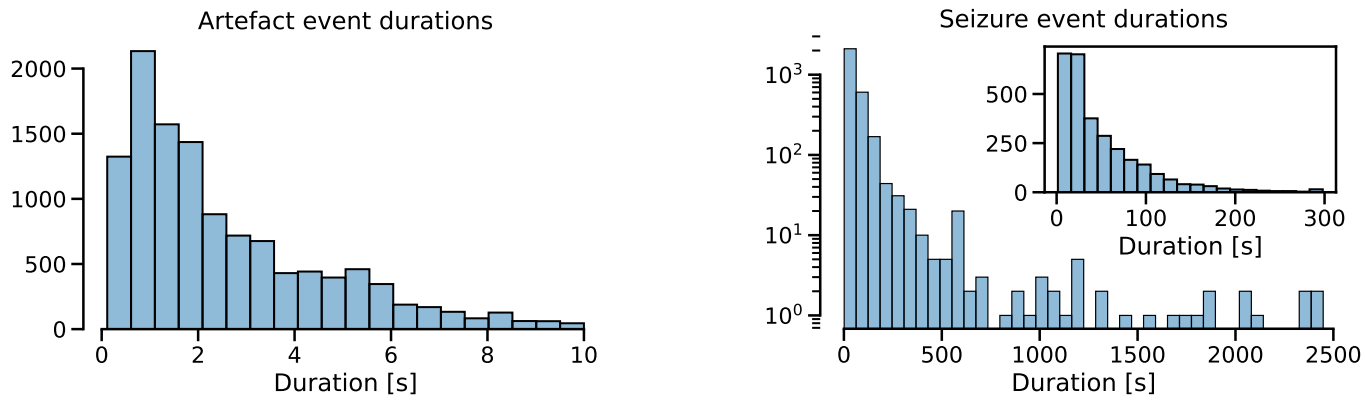
The duration distribution for both data sets is displayed in Figure 2.

C. Experiments

In this section, we provide a general description of the conducted experiments. For specific details regarding the training procedures and network architectures, we refer to Appendix A.

1) Evaluating EventNet's performance: To gauge performance of EventNet, its event detection performance is compared against U-Net-based approaches. Two recent competitions on event detection in EEG have been won by U-Net-based approaches [10], [11], making it a good benchmark. The U-Net is the best existing approach in literature that also succeeds at circumventing the context-resolution tradeoff, *and* can cope with variable duration. Regarding postprocessing, a simple median filter is used, to best evaluate the added benefit of EventNet's dual outputs compared to U-Net's single output. For artefact detection, a filter length of $0.1\,\mathrm{s}$ is used, for seizure detection $1\,\mathrm{s}$.

Defining a performance measure for event detection is not straightforward. In classification problems, defining a *hit* or



(a) (b)

Fig. 2: Histograms of event durations for both data sets. For the artefact data we only use events shorter than 10 s. For the seizure data, we use the entire data set. The inset histogram shows duration distributions for events lasting shorter than 300 s. Note the log scale for histogram counts in the main seizure figure.

miss in the context of predictions is quite easy. A classification prediction is unambiguously matched to a ground-truth label. For event detection problems, however, saying whether or not a predicted detection is a good match to the ground-truth data is more difficult. A set of predicted events needs to be matched with a set of ground-truth events, taking into account overlap between events of both sets. Depending on the way the sets are matched, and how much overlap there is between a predicted and ground-truth event, different performance measures can be defined. In epileptic seizure detection, for example, multiple measures are commonly used, each representing different application-specific properties of a "good" model [18].

In order to compare the methods in our experiments, we use a simple, general performance measure. Given a set of predicted and ground-truth events, the best match between events in the two sets is found. To match events, the total overlap between events from the two sets is maximized. Ground-truth events that are not matched to a predicted event are counted as false-negatives. Predicted event that are not matched to a ground-truth events are counted as false-positives. For the matched events, the Intersection-over-Union (IoU) is investigated. An IoU higher than 0.5 is considered a correct detection, lower than 0.5 is considered as an incorrect detection. To achieve a high IoU, an event's center *and* duration need to be predicted well. Some overlap will be caused by good center prediction and poor duration prediction but will lead to a low IoU value. The IoU-based measure is the main focus. Next to that, the *any-overlap* measure [18] will also be considered, where a *hit* is considered as any overlap between a matched predicted and ground-truth event. This measure can be considered as the limiting case of the IoU threshold going to zero (instead of 0.5 used above). Comparison of the *any-overlap* measure with this IoU measure can be used to evaluate which events were localized well, but showed a poor duration prediction.

2) Duration predictions: Rough information about the quality of duration predictions is captured by the distinction between IoU ≥ 0.5 and *any-overlap*, but this distinction does not tell the whole story. To better analyse duration predictions, the predicted durations for hits corresponding to the *any-overlap* measure are compared to durations of the corresponding ground-truth events. For good duration predictions, a clean linear relation between the predicted and ground-truth durations is expected. For the seizure events, two correlation coefficients are computed; one includes all events, and the other only includes events shorter than 250 s.This is because long events are few in number and cover a wide range of durations, such that the correlation coefficient can be disproportionally impacted by these few long events, while practical duration prediction is not concerned with a small number of mistakes for very long events.

3) Performance growth: Our approach to event detection is inspired by object detection work by the image processing community, where researchers have access to massive data sets. The MS COCO benchmark [19], for example, is made up of 2.5 million labeled objects, whereas our data sets only contain a few thousand target events. To better understand the impact of training set size on EventNet and the U-Net approaches, performance growth with growing training set size is investigated. For varying sizes for both data sets, both EventNet and U-Net are trained and later evaluated on the full test set. To investigate the relation of performance with training set size, a scalar performance measure is required. To this end, the F1-score of both models is computed at the optimal operating point (corresponding to the maximum F1-score for each training run), and plotted as a function of training set size.

The F1-score is expected to grow approximately like an S-curve (logistic growth) as a function of training set size. Performance is expected to start out low, grow quickly at first, and slowly tend to an asymptotic limit, the maximal performance for this specific problem and network. The logistic growth model is formulated as

$$y(n) = \frac{y^*}{1 + \exp(-k(n - n_0))} \tag{4}$$

with y the scalar performance measure, n the number of events

in the training set, y^* the asymptotic maximum of the growth model, k the growth rate, and n_0 the curve's midpoint. The logistic growth models (one for each data set) are fitted using a classical least-squares error method. Crucially, fitting such a growth model is done for illustration purposes only (to get a feeling for the asymptotic behavior of the two methods). It is not meant as a confident extrapolation (predicting the networks' performance for specific training set sizes).

For every training run, we train the models with the same amount of batches as training on the full training set would take (the number of epochs is corrected for the smaller training set sizes).

4) Extensive postprocessing: When using a U-Net for event detection, it is often desirable to design an extensive task-specific postprocessing scheme. Such a scheme is mostly used to reject false alarms, as is the case in [10]. The main conceptual benefit of EventNet compared to an approach like the U-Net is that a task-specific scheme is not required for EventNet; event properties are automatically learned during training. Hence, EventNet is also compared to a U-Net making use of a more extensive postprocessing scheme, heavily tuned on the specific use case.

We use the scheme of [10], which was designed specifically for U-Net-based seizure detection on this specific data set (based on a combination of expert knowledge and experiments), making it a good example of a highly-tuned, task-specific postprocessing scheme. Specifically, the postprocessing involves merging events that are close to each other (since they are likely part of the same seizure), removing low-confidence predictions (in comparison to other predictions in a given recording), and shortening detected events to avoid duration overshoot.

III. Results

1) Detection performance: Artefact and seizure detection precision is displayed in Figure 3 for different recall levels (computed using the IoU-based detection criterion, with 0.5 as threshold). Overlaps over 0.5 IoU are counted as *correct* detections. The *any-overlap* precision at these recall levels is also included. Detected events having less than 0.5 IoU with a matched reference event, but which are considered as a *hit* by the *any-overlap* measure (i.e., positive IoU), are counted as *almost*-correct detections. How well the models predict the duration of a detected event is gauged by the comparison between the *correct* and *almost* detections. Predicted events that are not considered a *hit* by the *any-overlap* measure are counted as *false-positives*. A network that cannot detect events at a specific recall level is indicated as *No detections*.

For the majority of the U-Nets' detections IoU < 0.5 is observed (red + orange class in Fig. 3). At every recall level, EventNet is outperforming the U-Nets, both on the IoU measure (green class) as well as on the *any-overlap* measure (green + orange class). Additionally, more events are found by EventNet compared to the U-Nets, shown by EventNet's higher recall level. Note that both models do not reach $100\,\%$ recall. Unlike for binary classification, not all targets (events) get detected by moving the decision threshold to zero because of the IoU threshold to count predicted events as *hits*.

TABLE I: Correlation between ground-truth and predicted durations. We consider all matched ground truth and predicted events that show any overlap (green + orange class in Fig. 3). For seizure events, we show results for all events and for ground-truth events that are shorter than $250\,\mathrm{s}$. All results are significant at the 0.01 level based on a two-sided t-test.

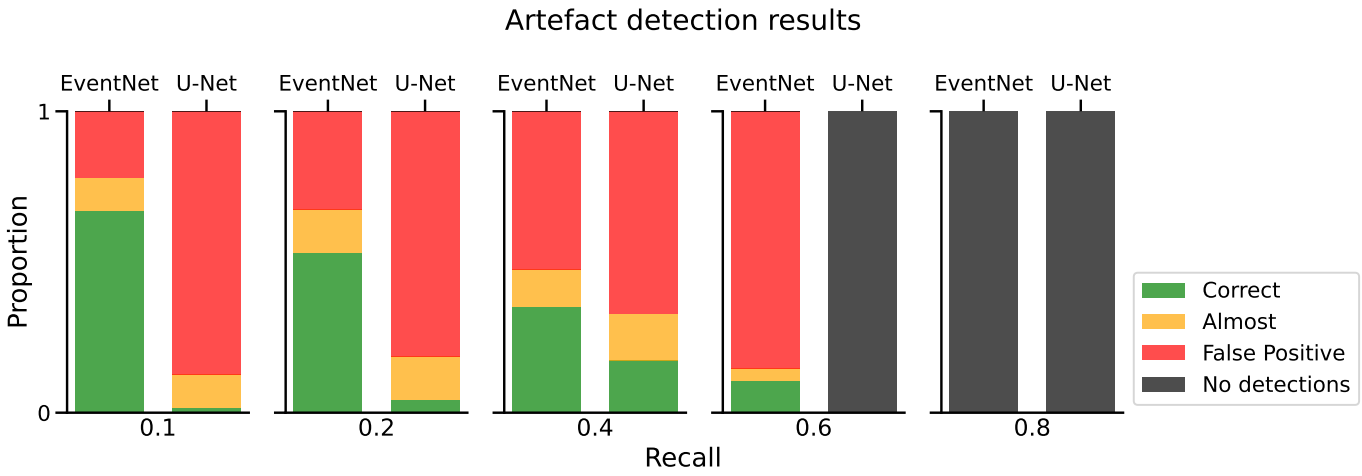
	Artefacts	Seizures	
		All events	Duration < 250 s
EventNet	0.61	0.37	0.76
U-Net	0.47	0.26	0.24

The seemingly rising precision-recall curve for the artefact U-Net is an unintuitive finding, as well as the fluctuating proportion of correct seizure detections. Normal precision-recall curves are expected to show high precision at low recall, and low precision at high recall. This behavior is not shown by the U-Net. This behavior can be explained by the decoding process of the U-Net's predictions. To predict events at specific recall levels, *confidence thresholds* need to be varied, which are set to correspond to a specific cutoff of the U-Net output to distinguish between *event* and *background*. Because of some noise in this output signal, two events can easily be detected as a single long event if the threshold is low. On the other side, a single event can easily be split by the U-Net into multiple shorter ones if the threshold is rather high. This sensitivity to a cutoff, and the fact that a $100\,\%$ recall can not be achieved, can result in an irregular precision-recall curve with a single Pareto-optimal operating point.

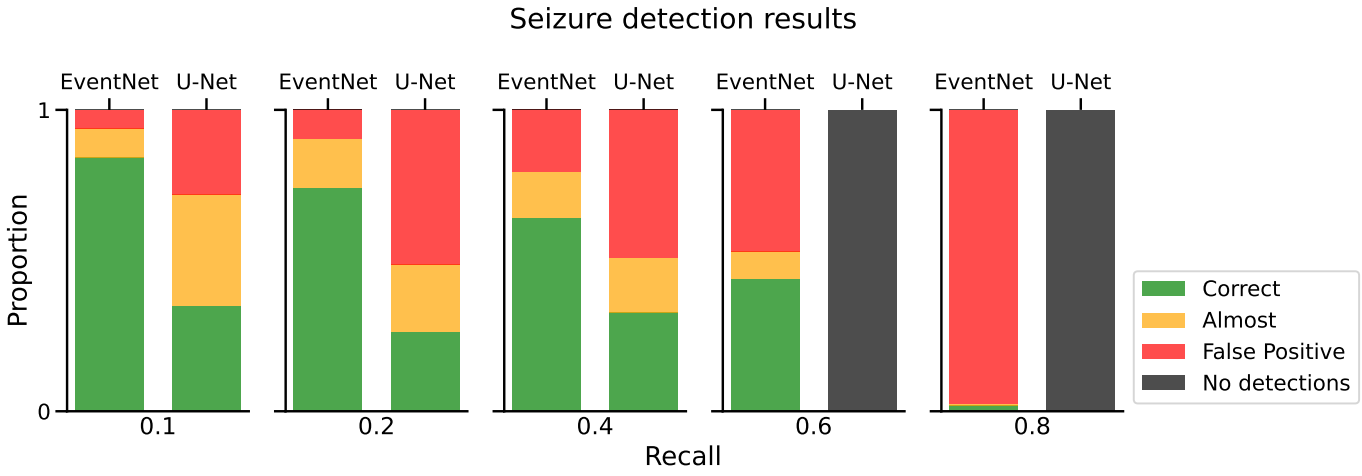
2) Duration predictions: Correlation values between ground truth and predicted durations are shown in Table I for both networks. Both networks are evaluated at the operating point corresponding to 0.4 recall (shown in Figure 3.) For artefact events, better correlation is clearly shown by EventNet compared to the U-Net. For seizure events, better correlation is shown by EventNet than by the U-Net for all events, and also for events shorter than $250\,\mathrm{s}$.

3) Performance growth: Performance growth results for varying training set size are shown in Figure 4. For artefact detection, asymptotic F1-scores are estimated for EventNet at 0.35, and at 0.25 for the U-Net (note that we do not claim these asymptotes to be real performance limits, they are here for illustration purposes). For seizure detection, a maximal F1-score for EventNet of 0.45 is estimated by the growth model, and at 0.33 for the U-Net. Visually, it can be concluded that the artefact data set seems to contain enough events to properly train a network. Asymptotic behavior is shown by both the scatter points and growth model. For seizure detection, the training set is shown by the growth model to be large enough to train a U-Net, while the same asymptotic behavior is not yet shown for EventNet. The actual performance measurements, however, are rather scattered, making it difficult to draw hard conclusions for the seizure detection task.

4) Extensive postprocessing: Seizure detection results of the comparison between EventNet and the U-Net with an extensive postprocessing (from [10]) are shown in Figure 5. Compared to the results in Figure 3, the U-Net's false positives are indeed decreased by the postprocessing. Additionaly, the quality of the U-Net's detections is improved (shown through

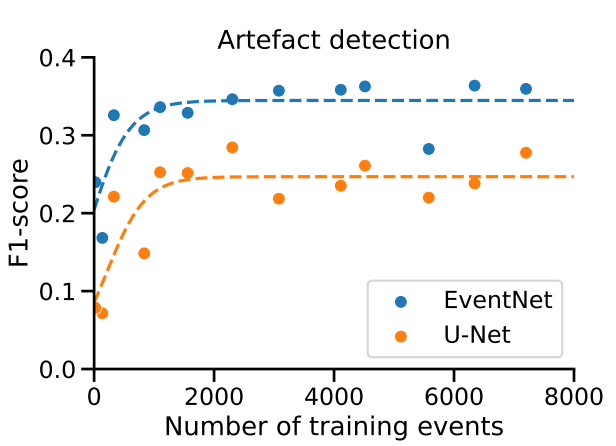


(a)

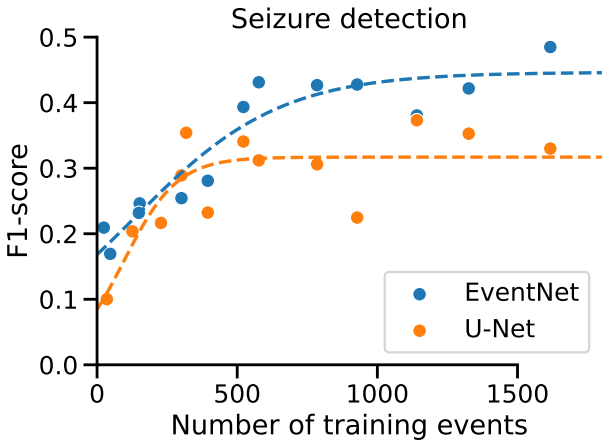


(b)

Fig. 3: Detection results for artefact and seizure events at different recall levels (computed using the IoU-based detection criterion). *Correct* detections correspond to IoU ≥ 0.5. *Almost* correct detections correspond to an *any-overlap* hit but IoU < 0.5. *False positive* detections have no overlap with ground truth events. *No detections* indicates that the network cannot detect events at that recall level.



(a)



(b)

Fig. 4: Performance growth for growing training set size. Scatter points represent the measured values for specific numbers of training events. Dashed lines show a logistic growth model fitted to the scatter points.

the higher proportion of detections with IoU $\geq$ 0.5, the green class in Fig. 3). At the 0.1 recall level, the U-Net's detections are comparable to EventNet, showing the strength of a heavily-tailored postprocessing scheme. At higher recall levels, however, the U-Net's precision is not improved to EventNet's precision scores. The maximum recall level is severely decreased compared to the "simplistic" postprocessing with a median filter.

IV. Discussion

A. Discussion of experimental results

EventNet outperforms the U-Net benchmarks in both use cases, and even when an extensive postprocessing is added to the U-Net. Figure 3 shows that the proportion of *correct* detections of EventNet is higher than the total proportion of *any-overlap* detections (green + orange class) of the U-Nets at every recall level. Additionally, EventNet manages to detect more events (higher maximal recall) than the U-Nets do. Most of EventNet's detections estimate the duration relatively reliably, looking at the ratio of *correct* (IoU ≥ 0.5, green class) detections to *any-overlap* detections (IoU > 0, orange + green class). The U-Nets seem to struggle with correctly estimating durations, based on the large proportion of *almost* detections (orange class).

Looking at the correlation between predicted durations and ground-truth durations, this conclusion becomes even stronger. For both data sets, EventNet shows better duration prediction than the U-Nets.

Our performance growth experiment further confirms what we see in Figure 3, i.e., that EventNet can achieve better performance than U-Net-based approaches, and this for all training set sizes. This experiment also showed that enough training events are present in both data sets, i.e., the training would not benefit significantly from additional data examples. For artefact detection, saturating performance growth can be observed for both methods. For seizure detection, saturating performance growth can be observed according to the growth models, but there is still substantial variation in performance for this amount of training events.

When comparing EventNet to a U-net with highly-tuned postprocessing (Figure 5), a similar precision is found at low recall values. Nevertheless, a higher maximal recall is still achieved by EventNet. The extensive postprocessing is mainly concerned with reducing false positive events (as can be seen by the higher precision but lower recall.) It is shown that comparable or better performance can be achieved by EventNet that is trained end-to-end, without any extensively tuned postprocessing, which can be viewed as a major advantage.

B. Using an event-based framework

We have proposed an event-based deep learning framework for time series, and we have applied it to artefact and seizure detection in EEG. Due to the end-to-end nature of EventNet, and EventNet's ConvNet backbone learning its own data representation, our framework is more broadly applicable than these two use cases.

One of the major benefits of working with an event-based framework like EventNet is the intuitive nature of labels. The way EventNet uses event labels matches closely to how human annotators would work with these labels (defining a start and stop time, which is equivalent to a center point and duration). This close match allows for easier feedback from human experts when developing a machine learning solution. For both U-Net-based and the more classical classification schemes, labeled events need to be translated to a sequence of classification targets, and back to events for inference. This can add friction to the development process, and hinder feedback.

The major difference between EventNet and previous classification-based approaches is the lack of case-specific postprocessing. EventNet can directly go from output to event, whereas classification-based approaches (not only a U-Net) will always need some kind of tailored translation step before events can be listed. EventNet drops the requirement for domain-specific postprocessing rules. These can involve the expected event duration, duration ranges, how quickly events can follow each other, etc. While EventNet might also benefit from such postprocessing, it is not a crucial step as opposed to existing approaches. Our approach can automatically learn most of those patterns from data.

The need for a large amount of data is a downside of EventNet (and also shared by U-Net-based approaches.) The performance growth experiment indicates that the artefact and seizure data sets contain enough training events to properly train a model. However, thousands of events are present in both data sets. For new use cases, this many training events might not be available. In contrast to EventNet, a more classical classification model that uses features crafted by experts can learn patterns much more easily from fewer events. For small-scale use cases, such classical models can even be preferred over EventNet. However, it might be possible to finetune an instance of EventNet for such small-scale use cases. This would require an EventNet that has been trained on a similar task using a large-scale data set. Such transfer learning schemes might then be able to bring the benefits of EventNet to small-scale use cases.

Next to sheer size, diversity of durations needs to be considered in a training set. EventNet learns to directly predict duration, so it requires a broad range of example durations to learn from. The artefact and seizure data sets both cover a wide range of durations, but are heavily skewed towards shorter events. The impact of duration distribution on performance is unknown at present. One can imagine, for example, that shorter durations are easier to predict if shorter events are more consistent in nature than longer events but this remains to be investigated.

V. Conclusion

In this paper, we have proposed EventNet, an event detection model for time series, allowing to detect EEG events. The model can directly detect events of variable duration in long recordings. It shows substantially improved performance compared to a benchmark method. We have shown improved localization of events in time, and improved duration prediction, for artefact and epileptic seizure events.

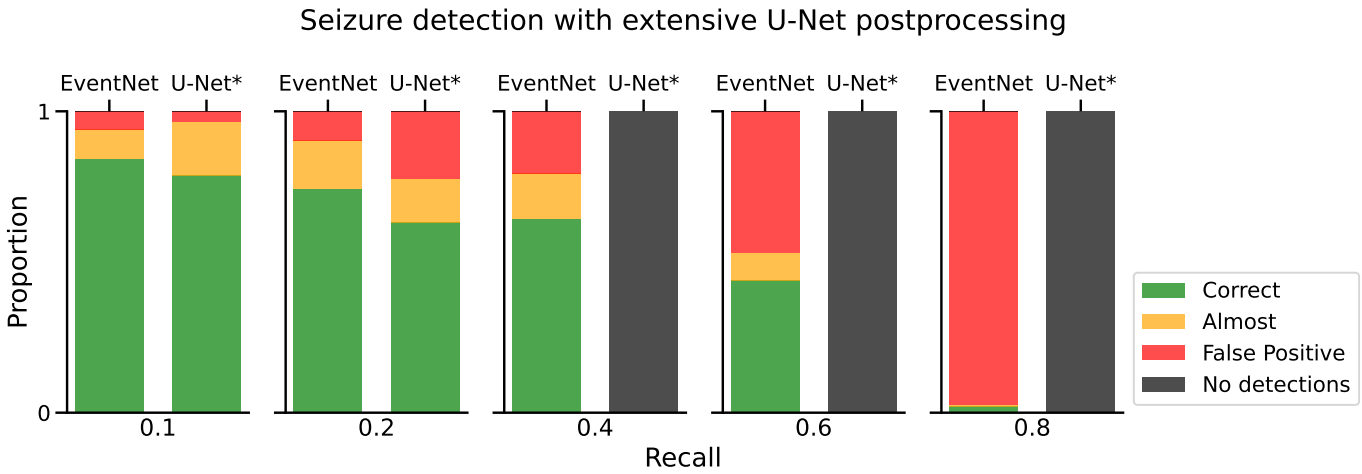


Fig. 5: Seizure detection results with more extensive postprocessing applied to the U-Net output (indicated as U-Net*).

In contrast to existing methods, we require no postprocessing routine to translate predictions into a set of events. Our model can be extended to other EEG event-detection tasks and to other signal processing tasks where signal events are involved.

References

[1] E. Niedermeyer and F. L. da Silva, *Electroencephalography: basic principles, clinical applications, and related fields*. Lippincott Williams & Wilkins, 2005.

[2] P. M. Kulkarni *et al.*, "A deep learning approach for real-time detection of sleep spindles," *Journal of neural engineering*, vol. 16, no. 3, p. 036 004, 2019.

[3] A. Malafeev *et al.*, "Automatic artefact detection in single-channel sleep EEG recordings," *Journal of sleep research*, vol. 28, no. 2, e12679, 2019.

[4] L. Webb *et al.*, "Automated detection of artefacts in neonatal EEG with residual neural networks," *Computer Methods and Programs in Biomedicine*, vol. 208, p. 106 194, 2021.

[5] K. Vandecasteele *et al.*, "Visual seizure annotation and automated seizure detection using behind-the-ear electroencephalographic channels," *Epilepsia*, vol. 61, no. 4, pp. 766–775, 2020.

[6] J. Zhang *et al.*, "Automatic annotation correction for wearable EEG based epileptic seizure detection," *Journal of Neural Engineering*, vol. 19, no. 1, p. 016 038, 2022.

[7] A. N. Olesen *et al.*, "MSED: A multi-modal sleep event detection model for clinical sleep analysis," *arXiv preprint arXiv:2101.02530*, 2021.

[8] S. Chambon *et al.*, "DOSED: A deep learning approach to detect multiple sleep micro-events in EEG signal," *Journal of neuroscience methods*, vol. 321, pp. 64–78, 2019.

[9] J. Redmon *et al.*, "You Only Look Once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016.

[10] C. Chatzichristos *et al.*, "Epileptic seizure detection in EEG via fusion of multi-view attention-gated U-Net deep neural networks," in *2020 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, 2020, pp. 1–7. DOI: 10.1109/SPMB50085.2020.9353630.

[11] H. Li and Y. Guan, "DeepSleep convolutional neural network allows accurate and fast detection of sleep arousal," *Communications biology*, vol. 4, no. 1, pp. 1–11, 2021.

[12] X. Zhou *et al.*, "Objects as points," *arXiv preprint arXiv:1904.07850*, 2019.

[13] Z. Liu *et al.*, "Training-time-friendly network for real-time object detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 2020, pp. 11 685–11 692.

[14] T.-Y. Lin *et al.*, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[15] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 734–750.

[16] A. Hamid *et al.*, "The Temple University artifact corpus: An annotated corpus of EEG artifacts," in *2020 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, IEEE, 2020, pp. 1–4.

[17] V. Shah *et al.*, "The Temple University hospital seizure detection corpus," *Frontiers in neuroinformatics*, vol. 12, p. 83, 2018.

[18] S. Ziyabari *et al.*, "Objective evaluation metrics for automatic classification of EEG events," *arXiv preprint arXiv:1712.10107*, 2017.

[19] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *European conference on computer vision*, Springer, 2014, pp. 740–755.

[20] J. Schlemper *et al.*, "Attention gated networks: Learning to leverage salient regions in medical images," *Medical image analysis*, vol. 53, pp. 197–207, 2019.

Appendix

A. Network and training details

1) Artefact detection: Artefact events are detected based on a single-channel input. Both EventNet and the U-Net use a similar backbone. The networks' base "building block" is constructed using a separable 1D convolution, a batch-normalization layer, and an ELU activation function. Every down- or upsample step is taken with a factor 4. At the original input resolution (before the first downsampling step), the networks are using one such block, at every other stage using two such blocks (two at the "downward" path, and two at the "upward" path). The full backbone architecture is shown in Table A1. Two "size 1" convolutions are used by EventNet on the output at stage 2' to produce the center and duration signals. A single "size 1" convolution is used by the U-Net to produce its output. Both networks are trained on input signal segments of 200 s and tested using a full EEG channel recording. EventNet's loss function and training details are described in the main text. The artefact U-Net is trained with the same optimizer and number of epochs as EventNet. The U-Net's loss function is binary cross-entropy with label smoothing, similar to the U-Net of [10].

For U-Net postprocessing, a median filter with a size corresponding to 0.1 s is used. The shortest event in the artefact data set lasts 0.2 s, so it is considered a reasonable filter size. The filter is used *after* thresholding the U-Net output at a certain operating threshold.

TABLE A1: Artefact detection backbone. The different stages are connected in a U-Net-like way, stage 5' for example concatenates the features of stage 5 and upsampled features of stage 6'. Every stage besides 0 contains two separable convolutions with the given hyperparameters. Stages $(\cdot)'$ have a dropout layer between the two convolutions. The *stride factor* indicates the total "downsample factor" for the given stage. This backbone produces a feature signal at 1/16th the original sampling frequency.

Stage	#Filters	Kernel size	Stride factor
0	32	20	1
1	64	20	4
2	64	15	16
3	64	15	64
4	64	10	256
5	64	5	1024
6'	64	5	4096
5'	64	5	1024
4'	64	10	256
3'	64	15	64
2'	64	15	16

2) Seizure detection: The seizure detection backbone is constructed following the network of [10]. This backbone is made up of a channel-independent encoder, and channel information is only merged at the deepest part of the backbone and the skip connections. Attention Gating [20] is used in the skip connections to merge channel information. The original network is only used until "stage 4", at 1/32th of the original sampling frequency. After this stage, EventNet's center and duration heads are appended. This is done to limit computational effort and memory footprint, since there is little performance impact compared to applying the heads at the original sampling frequency (as is done in the U-Nets of [10]). The seizure detection U-Net used in our experiment is the original model of [10], with an output at the same resolution as the original input. Similar to artefact detection, the seizure detection networks are trained on input segments of 200 s, and tested on full recordings.

For U-Net postprocessing, a median filter with a size corresponding to 2 s is used. Similar to the artefact detection setting, this filter is applied after thresholding the U-Net's output.

B. Stable focal loss

The softplus function is a part of focal loss. Instabilities are expected when computing the gradient of the focal loss because of this softplus function. The softplus function can be found in the "cross entropy factor" of the focal loss formulation (taking the logarithm of a sigmoid activation):

$$\log \sigma(l(x)) = \log e^l - \log(e^l + 1) = l - \text{softplus}(l)$$

and

$$\log(1 - \sigma(l(x))) = \log(1 - \frac{e^l}{e^l + 1})$$

$$= \log 1 - \log(e^l + 1) = -\text{softplus}(l)$$

The softplus function $\log(e^{(\cdot)} + 1)$ can be rewritten in a numerically more stable form that avoids a possibly exploding gradient as follows:

$$\text{softplus}_{stable}(x) = \log(e^{-|x|} + 1) + \max(x, 0)$$