FH AACHEN
UNIVERSITY OF APPLIED SCIENCES

FH Aachen, Campus Jülich
Medizintechnik und Technomathematik
Angewandte Mathematik und Informatik (AOS)

EEG artifact detection via time series segmentation

Seminar paper
of
Aymane HEMMOUDA

Under the supervision and guidance of:
Prof. Dr. rer. nat. Stephan Bialonski
&
M. Sc. Niklas Grieger

October 2, 2025

Abstract

Monitoring brain activities for diagnosis and studies is primarily done through EEG recordings. The latter, however, is highly susceptible to extraneous disturbances called artifacts. An automatic pipeline that detects and removes these artifacts from the normal brain activities promises to improve the quality of the diagnosis and the accuracy of the analysis.

In this paper, we explore the first part (detecting artifacts) by training and evaluating the publicly available U-Net based model, SUMO2, on a dataset of annotated, real-life EEG recordings: The Temple University Artifact Corpus (v2.0.0). The model achieves acceptable results. We discuss possible causes behind the achieved results, as well as potential future improvements.

Contents

1 Introduction

Monitoring brain activities for diagnosis and studies is primarily done through Electroencephalography (EEG). It is a noninvasive method that consists of placing a certain amount of electrodes in specific locations on the scalp of a patient; these electrodes then read and capture the electrical signals that the brain produces. The recorded EEG data, however, is often contaminated with electrical signals that do not come from normal brain activities; these extraneous, unwanted signals are called artifacts. These can originate from muscle movements, eye movements, faulty or unfastened electrodes, and chewing, among other events. Artifacts can greatly distort the EEG signals, leading to misinterpretation and unreliable results [1]. Removing them is thus very important for a more accurate analysis and diagnosis.

In this paper, we explore a Machine Learning (ML) solution to identify these artifacts in EEG data through time series segmentation. This is now more feasible than ever before, thanks to the *recent* release of the Temple University Artifact Corpus v2.0.0 (TUAR); a dataset of annotated and labeled real life EEG data. We train and evaluate SUMO2, a publicly available U-Net based model, on this dataset, and we achieve acceptable results.

2 Overview of EEG Recordings

To record someone's EEG data, a technician first places a certain amount of electrodes in specific locations on the scalp; the amount of electrodes, their locations, as well as how the electrical signals are referenced and computed, determine what is referred to as a montage. Different montages help accentuate different events of interest. Every signal that is derived from one or multiple electrodes is called a channel. An EEG recording is therefore made up of multiple channels, each representing the electrical activity of a specific part of the brain over time [1].

Even though EEG data may be recorded using one montage, it is possible to view it in light of another (i.e., by recomputing it in the desired montage). And since different montages highlight different brain activities and events, technicians may use one montage during recording and another when viewing data to label it, for example.

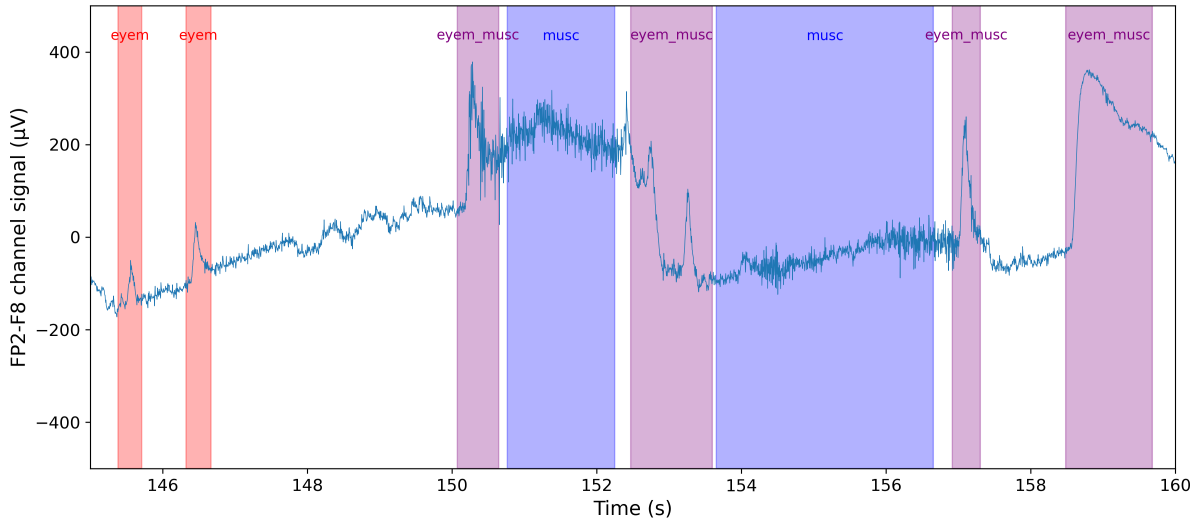

Figure 1: An example that showcases a snippet of EEG data with artifacts. Here, for clarity purposes, only a single channel, namely the FP2-F8 channel, is shown.

3 Methods

3.1 Dataset

The TUAR dataset is a subset of the world's largest open source corpus of real-life EEG data [2]. It consists entirely of clinical data collected over the span of multiple years starting from 2002 at the Temple University Hospital. The dataset contains 310 labeled EEG recordings, totalling almost 100 hours of recordings, collected from 213 different patients — 115 (54%) females, and 98 (46%) males, whose ages range from 16 to 88 years old, with an average of 51.68. It's a diverse dataset, containing not only annotations for EEG artifacts, but also for seizure events, amassing a staggering 160073 labeled artifact instances. Figure 2 below shows the frequency of each kind of artifact, while table 1 explains what the artifacts' labels actually represent.

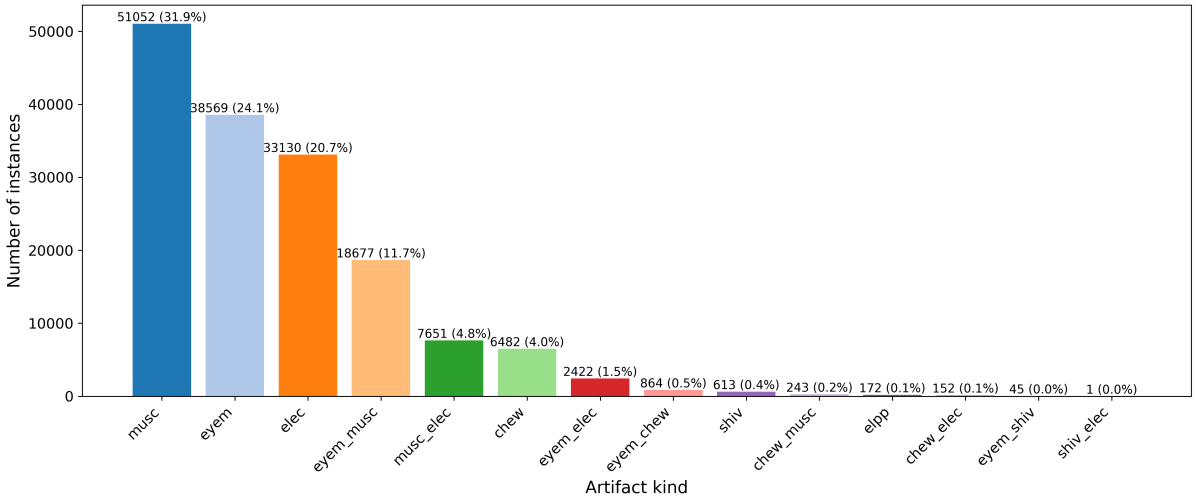

Figure 2: The number of EEG artifacts in TUAR by kind

Simple label	Significance		Compound label	Significance
musc	stands for muscle, represents patient movement		eyem_musc	both eyem and musc happening at the same time
eyem	stands for and represents eye movement		musc_elec	both musc and elec happening at the same time
elec	represents all kinds of electrical events, e.g. an electrode popping		eyem_elec	both eyem and elec happening at the same time
chew	stands for and represents chewing		eyem_chew	both eyem and chew happening at the same time
shiv	stands for and represents shivering		chew_musc	both chew and musc happening at the same time
elpp	unspecified		chew_elec	both chew and elec happening at the same time
			eyem_shiv	both eyem and shiv happening at the same time
			shiv_elec	both shiv and elec happening at the same time

Table 1: The different labels of TUAR EEG artifacts and what they represent

Since this is tackled as an artifact detection problem, no distinction is made between the different artifact classes (artifact kinds). We formulate the problem as an artifact vs. non-artifact problem, or artifact vs. background (i.e., normal, uncontaminated brain signals) problem. And, thanks to the high quality work done by the people behind the TUAR dataset, one can be confident that every instance of an artifact that exists in the recordings is known and labeled [2, p. 2]; meaning that any part of the recordings that is not labeled with one of the artifact classes contains no artifact and is just background data.

For our purposes, recordings in which seizures occurred have been discarded, as they are considered abnormal events and can mess with the models' performance. But even with that, the dataset still contains 1901 hours total of data [1], of which 338 hours (17.8%) are artifacts, and the remaining 1563 hours (82.2%) is background data.

3.2 Preprocessing

The dataset is originally segregated montage wise, grouping the different patients' files by the montage used while recording the EEG data (a folder for the recordings done with the AR montage, another for the LE montage, and another for the AR_A montage). However, since we wanted to ensure that the different splits (training, validation, and

[1] While the dataset contains just shy of 100 hours of recordings, each recording has multiple channels, so the actual **data** duration for each recording is its length (duration) multiplied by the number of channels it has.

test split) were representative of the overall dataset distribution with respect to age and gender, as well as ensuring that a patient's data only appears in one split (i.e., no data leakage), it made more sense to first group the different files patient wise (a folder for each patient's recordings and artifact annotations).

After removing the recordings in which a seizure occurred and patients that only had seizure recordings, only 268 recordings from a total of 196 patients were left. It's worth mentioning that, even though that is 1901 hours of data total, the high imbalance of the dataset (82.2% of the data is background data) made the task of detecting artifacts more challenging for the model.

Since all of the artifacts in the dataset were annotated and labeled while viewing the EEG data in the `TCP` montage, it only made sense to transform the 268 `EDF` files containing the recordings' data from the montage that they were in into the `TCP` montage.

Before feeding the data to the model, the dataset was split into training (with 118 patients), validation (with 39 patients), and test (with 39 patients) sets, corresponding to 60%, 20%, and 20%, respectively, after which the following preprocessing steps were applied:

1. All channels' signals were passed through a 5th-order Butterworth band-pass filter (0.5-80 Hz).
2. A 60 Hz notch filter (quality factor 15) was then applied.
3. Signals were then downsampled to 170 Hz, slightly above twice the maximum frequency of the filter (80 Hz).
4. Values outside of -500 to 500 µV were clipped.
5. And finally, the channels' data (signal) was segmented into 10-second segments with no overlap.

The Python code for these preprocessing steps can be found under: hemmouda.com/EAD

3.3 Model

The ML model used for this artifact detection problem is the publicly available SUMO2 model [3]. It is a U-Net [4] based model with two encoder and two decoder blocks that was originally designed for sleep spindle segmentation. Nonetheless, it can also be used for our task because it accepts input sequences of arbitrary length and outputs two segmentation masks that indicate, for each data point in the input sequence, whether the *element* of interest (in our case, artifacts) was present or not [2].

[2]This information was extracted and copied from the paper that conceived SUMO2 [3]. Information about how to get access to the model is also available in said paper.

3.4 Metrics

The metric used to evaluate the model's performance was that provided with the SUMO2 model, which is the F1 score. It's defined in the paper as such [3, p. 7]:

$$\text{F1} = \frac{2 \times (\text{precision} \times \text{recall})}{\text{precision} + \text{recall}}$$

Where precision and recall are:

$$\text{precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

Where TP, FP, and FN stand for True Positive, False Positive, and False Negative, respectively. And they represent the number of:

- **TP**: artifact predictions that are correct (the model predicted an artifact in a segment, and an artifact does indeed exist in that segment).
- **FP**: artifact predictions that are incorrect (the model predicted an artifact in a segment, but that segment contains no artifact).
- **FN**: artifacts that the model failed to predict (the model predicted that a segment contains no artifact, but that segment does actually contain an artifact).

These numbers, however, are only defined for a given overlap threshold percentage between the real artifact and the model's prediction of said artifact. Figure 3 below helps illustrate the concept better:

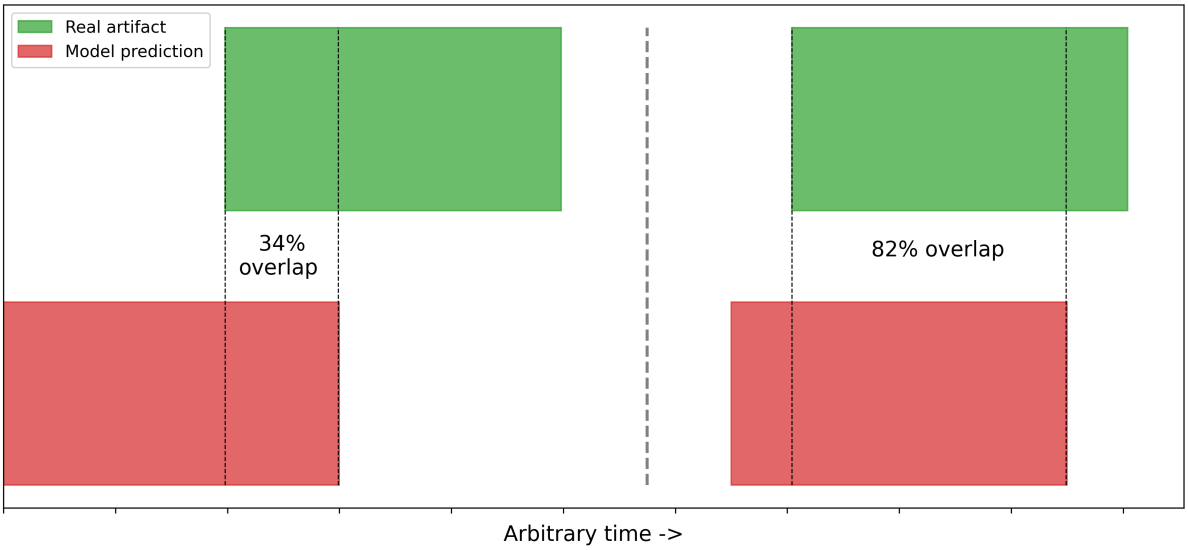

Figure 3: If the overlap threshold percentage is 20%, for example, then both predictions (the one on the left and the one on the right) would count as TP, that is because, in both cases, the prediction's overlap with the artifact is greater than the threshold. But if, for example, the threshold is more constraining, say 75%, then only the prediction on the right would count as a TP, whereas the one on the left would count as a FP.

The F1 score is a good metric because it's the harmonic mean of precision and recall, both of which measure different aspects of the model's performance: A high precision value indicates that if the model predicted an artifact, then it is indeed present, whereas a high recall value indicates that the model is able to detect most of the artifacts. And thus, a high F1 value indicates high precision and recall, and a low F1 value indicates the opposite.

4 Training and Results

As mentioned in Preprocessing 3.2, the training set contained 60% of the remaining patients, totaling 1159 hours of data. Of this, 81.02% (939 hours) was background data and 18.98% (220 hours) contained artifacts. The model trained for 602 epochs, with early stopping set at 300, meaning the model's performance stopped improving at the 302nd epoch. Despite using less than 60% of the full dataset (due to removed seizure recordings), completing training still required 4 days and 15 hours on a high-end machine [3], highlighting the dataset's large size.

The figure 4 below shows the results that model achieved for different overlap thresholds. As the graph shows, the model achieved an F1 score of around 0.54 for an overlap threshold of 20%.

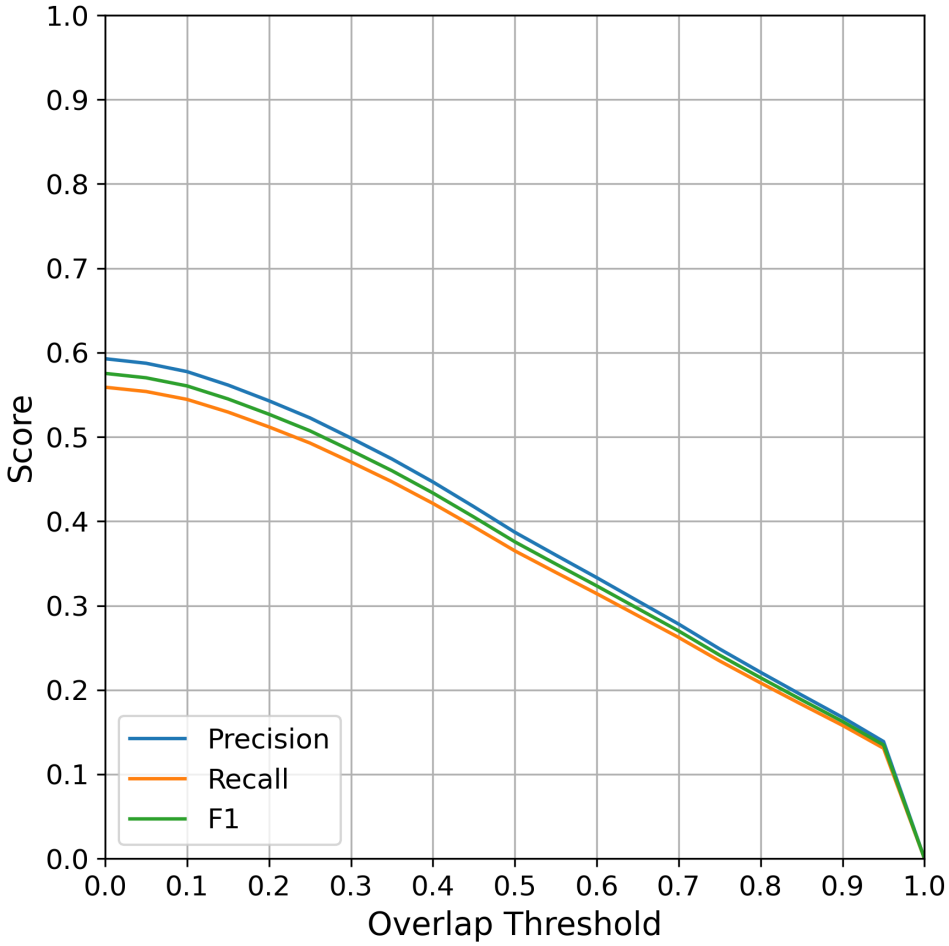

Figure 4: The model's F1 score over different overlap thresholds

[3] **CPU**: Intel Xeon Platinum 8168 (96) @ 3.700GHz. **GPU**: NVIDIA Quadro P5000. **GPU**: NVIDIA RTX A6000.

Referring back to the illustrative example of EEG data in figure 1, below is the model's prediction for the artifacts in that segment of data. As mentioned before, no distinction is made between the different artifact classes; therefore, all artifacts are seen by the model as being the same.

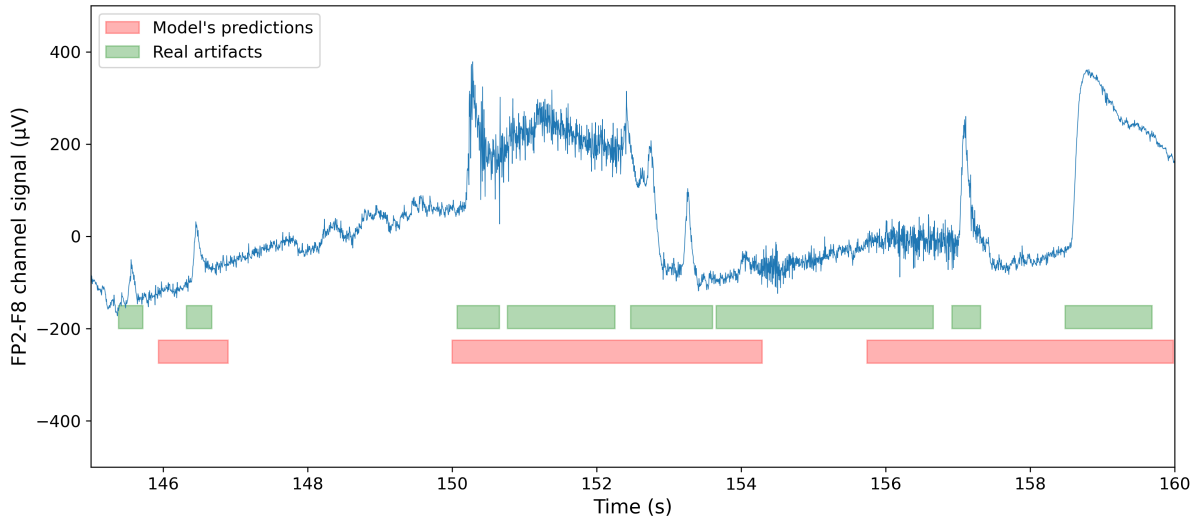

Figure 5: The model's predictions on the previously seen EEG data

As one can clearly see, the model's predictions are more or less acceptable, but they are not great. Notably and namely; it failed to predict the first artifact, it over estimated the duration of the second artifact, and it is not able to properly separate the different instances of artifacts when they are close together.
In this other example, however, although the predictions still aren't great, they are overall better.

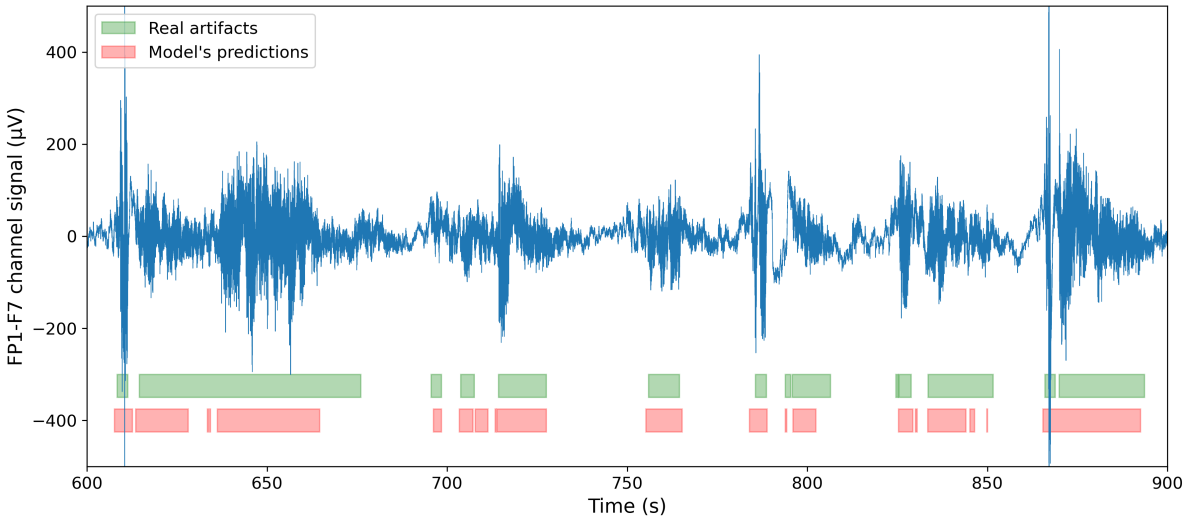

Figure 6: The model's predictions on a segment of another patient's data

It's important to point out the big difference in the artifacts durations between the two figures; in the first one (figure 5), the artifacts in the 15 seconds long segment of data happen to be shorter, with the largest one only being around 2.5 seconds long. Whereas in the second one (figure 6), the artifacts in the 300 seconds long segment of data are generally longer.

5 Discussion

Multiple hypotheses could explain the result achieved, for one, the disproportionate amount of background data to artifact data (which is to be expected in real-life datasets of EEG data); an imbalanced dataset in general could lead to poor performance; that is because the model sees one class of data a lot more frequently than the others.
Another reason could be the skewed distribution of the artifacts' durations. Figure 7 shows the median vs. the mean of the duration of the 6 most common artifact classes. The positive skew indicates that there are a number of artifacts whose duration is a lot longer than the rest. In the case of the `elec` artifact, this could be caused by a faulty electrode. This could also explain the overall better results in figure 6 when compared to figure 5; that is because the model is more accustomed to longer artifacts.

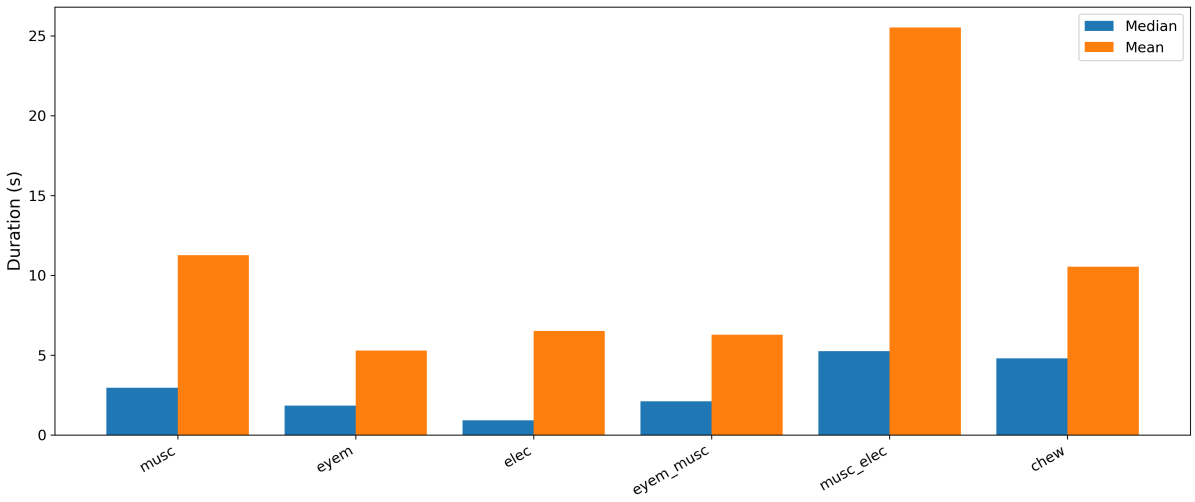

Figure 7: The median vs. the mean of the duration of the 6 most common artifact classes

Other papers that have worked with the TUAR dataset exist, namely: [5], [6], and [7]. The first paper in particular ([5]) is interesting because it compares different models and techniques using a modified version of the F1 score, where instead of defining TP, FP, and FN as numbers based on a given overlap threshold, they instead let them be a decimal value between 0 and 1, where 0 indicates no overlap, and 1 is perfect overlap. Even though this effectively means our results cannot be directly compared, below is a table with the results they shared:

Model	Modified precision	Modified recall	Modified F1 score
SegNet_single	0.682	0.569	0.620
SegNet_all	0.675	0.620	0.646
PatchTST	**0.684**	0.591	0.634
SegPatchT	0.672	0.628	0.649
WaveNet (theirs)	0.678	**0.667**	**0.672**

Table 2: The results shared in this paper [5, p. 3363]

6 Conclusion

With the release of the TUAR dataset, a new approach for detecting EEG artifacts was enabled. We've trained SUMO2, a U-Net based model, and we got acceptable results.

However, before being able to tackle the question of how can one remove the artifacts to improve the accuracy of analysis and diagnosis, we should first explore the impact on performance of an in-depth cleaning of the dataset, or using some rebalancing techniques to address the disproportionate amount of background data to artifact data, or even try training multiple SUMO2 models, one for each artifact class, with the hope that, if it focused on a single type during training, it would be more precise in detecting that type. But, that is left to be explored in future work.

7 Acknowledgements

We are grateful to Mr. Martin Reißel for providing us with computing resources.

References

1. Teplan M et al. Fundamentals of EEG measurement. Measurement science review 2002; 2:1–11
2. Hamid A et al. The Temple University Artifact Corpus: An Annotated Corpus of EEG Artifacts. *2020 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*. IEEE, 2020 Dec :1–4. DOI: 10.1109/spmb50085.2020.9353647. Available from: http://dx.doi.org/10.1109/SPMB50085.2020.9353647
3. Grieger N, Mehrkanoon S, Ritter P, and Bialonski S. From Sleep Staging to Spindle Detection: Evaluating End-to-End Automated Sleep Analysis. 2025. DOI: 10.48550/ARXIV.2505.05371. Available from: https://arxiv.org/abs/2505.05371
4. Ronneberger O, Fischer P, and Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. 2015. DOI: 10.48550/ARXIV.1505.04597. Available from: https://arxiv.org/abs/1505.04597
5. Yu Y, Li Y, Zhou Y, Wang Y, and Wang J. A Learnable and Explainable Wavelet Neural Network for EEG Artifacts Detection and Classification. IEEE Transactions on Neural Systems and Rehabilitation Engineering 2024; 32:3358–68. DOI: 10.1109/tnsre.2024.3452315. Available from: http://dx.doi.org/10.1109/TNSRE.2024.3452315
6. Qendro L, Campbell A, Liò P, and Mascolo C. High Frequency EEG Artifact Detection with Uncertainty via Early Exit Paradigm. 2021. DOI: 10.48550/ARXIV.2107.10746. Available from: https://arxiv.org/abs/2107.10746
7. Roy S. Machine Learning for removing EEG artifacts: Setting the benchmark. 2019. DOI: 10.48550/ARXIV.1903.07825. Available from: https://arxiv.org/abs/1903.07825