Residual Neural Networks to Distinguish Craving Smokers, Non-craving Smokers and Non-smokers by their EEG signals

1st Christoph Doell
Dept of Computer and Information Science
University of Konstanz
Konstanz, Germany
christoph.doell@uni-konstanz.de

2nd Sarah E. Donohue*
Dept of Behavioral Neurology
Leibniz Institute for Neurobiology
Magdeburg, Germany
donohue@med.ovgu.de

3rd Christian Borgelt
Dept of Computer and Information Science
University of Konstanz
Konstanz, Germany
christian@borgelt.net

***Abstract*—We investigate the differences in brain signals of craving smokers, non-craving smokers, and non-smokers. To this end, we use data from resting-state EEG measurements to train predictive models to distinguish these three groups. We improve the neural network models applied earlier in two ways: firstly by adding channel-wise convolutional layers, secondly by adding residual connections to the network. We further extend the validation to make it similar to a real world scenario, in which a prediction is based on all data available for this measurement. Finally, we analyze the prediction quality for each measurement individually. Our results demonstrate significant improvements.**

***Index Terms*—Addiction, Smoker, Craving, Residual Neural Network, EEG, Classification**

I. Introduction and related work

The use of drugs, whether legal or illegal, has a profound impact on the brain. Not only do drugs act on neuronal receptors, causing changes in the signaling patterns of these cells, they also induce a sensation of craving when they are gone, due to the effects of withdrawal. Studying these effects is challenging, as it is ethically questionable to administer drugs to humans, and animal models may not reflect the same changes that occur in the human brain [14], [16]. However, the legality of some drugs, such as nicotine, makes it possible to study addiction in humans, and non-invasive measures such as electroencephalography (EEG), provide a time-point-by-time-point measure of the neural signal. Understanding how these EEG signals may differ between non-smokers, sated smokers (non-craving), and craving smokers could help shed new light on the changes of brain function in addiction.

One promising way to examine such differences is to apply recent advancements in machine learning algorithms to accurately classify these data into the categories of craving smokers, non-craving smokers, and non-smokers. That is, as nicotine influences neural functioning, this should manifest in some alteration of signaling that can be measured with EEG. Moreover, the brain-state of craving would be expected to have different (and possibly more salient) patterns than non-craving, due to the stress/arousal present with the desire to smoke.

*This work was partially funded by DFG SFB 779 TP A14N

We are specifically interested in the differences in the resting state, without the presence of a specific task. Resting-state data are typically acquired continuously over a period of time during which the participant has no specific task to do. The idea is to monitor natural fluctuations in activity that are not due to the onset of an image or sound. Any neural activity that is a result of the use of nicotine, and therefore common across the smokers, should be present in the resting state, and not masked by any task. This pattern of activity may be subtle, however, and difficult to extract without knowing what the specific patterns are. As such, neural network models may be able to find patterns and classify data in a way that more traditional cognitive neuroscience-based analyses cannot.

Research into such topics using EEG has typically looked at patterns in various frequency bands to determine if there are differences between groups. Although the findings in this area are rather inconclusive, they do point to the potential for characterizable differences. For example, Brown [1] found reduced alpha and increased rhythmic high frequency when looking at the EEG signals of smokers vs. non-smokers. Rass [17] detected reduced alpha as well, but also observed reduced delta when comparing smokers to non-smokers. Additionally, Knott [10] reports reduced delta and increased beta within smokers as a function of image-induced craving. Together, these studies suggest that there may be differences present, but the definition of the precise patterns that lead to such differences needs further exploration. Neural networks, with their ability to automatically identify and learn important features and with their recent advances like convolutions with skip connections, could be well-suited for this task.

In our previous work [4], we created several models to distinguish two classes and all three classes of the resting-state EEG data. While simple Bayesian models failed, neural networks succeeded in predicting significantly better than guessing. We used two kinds of neural networks: The first processed the data channel-wise, mostly using dense layers. When distinguishing craving from non-smokers, the dense network was successful, with a median prediction accuracy of 63.7%. However, it failed at the three class problem, unable to

perform predictions better than random guessing. The second network used mainly convolutions. For the two-class problem it achieved only 62.8%. However, with all three classes, the convolutional neural network models achieved a median class-balanced accuracy of 37.6%, which is the current baseline. In contrast to the earlier works on frequency bands, these results exhibit predictions significantly better than random guessing. The goal of the present work is therefore to improve the former models that distinguish between EEG data from craving smokers, non-craving smokers, and non-smokers.

II. Difficulties when classifying EEG data

Neural networks are used in many applications, where features in biological data are to be learned, like medical imaging or brain computer interfaces [15]. Here, we emphasize the difficulties to classify EEG data by comparing it to two other prominent tasks, where neural network models were applied with great success: object recognition in pictures [11] and playing the Asian board game Go [19].

Number of data samples. For both of the prominent tasks, plenty of data is available. The Imagenet challenge uses a database of millions of images, in which objects are to be found. Mastering the game of Go can rely on the current model playing against itself in order to generate more training data. Therefore, more training data can be constructed whenever needed and the number of games is, in principle, unlimited.

In stark contrast to the favorable situation in these two tasks, we use only 48 measurements in total. This number of measurements is so small for a data analysis task, that we recommend the verification of our results with a bigger data set. We combine several techniques to augment our data in order to achieve results, which are as reliable as possible, given the limited number of samples.

Noise in the data. An EEG electrode not only measures signals stemming from the brain, but also activity from all electrical sources, whether they be externally generated electrical oscillations or subject-generated muscle activity such as breathing, heart beats, or eye movement. As the signal from the brain, that is measured on the scalp, has to travel through the skull and other tissues, by the time it reaches a given electrode, this signal is in the order of μV and has to be amplified before it can be recorded. Other sources of noise, however, may be closer to a given electrode and also larger in size than the brain signal, essentially dominating the data. Moreover, this noise often differs between electrodes, which is a result of their relative positions to the source of the noise.

For the game of Go, there is no noise in the data. The possible positions of the stones on the board are fixed. Even if an error occurs when a move is recorded, this can be fixed in many cases by the information contained in the other moves. Noise in pictures is different and can depend on the camera used, the lighting at the moment when the picture was taken, or the aptitude of the photographer. Still, these problems can (usually) be solved easily by removing bad or noisy images, since humans have the ability to compare pictures to reality.

Knowledge of the task and of important features. Many people have the ability to recognize objects in pictures. Furthermore, we can explain why we think the picture contains an object. For an animal we can describe where we see legs, arms, the head — or other features, which might help with the classification. Therefore we can compare our interpretation of the image to the one performed by the network and can even analyze how the network may be tricked into making a wrong prediction, for example by an adversarial example [7].

For the game of Go, a vast body of expert knowledge exists, so we can validate whether a model plays moves similar to those a human expert would play. And if they are different, but the model still wins, we gain insights into the game and how to play it well, as actually happened for Go.

When looking for differences between craving, non-craving and non-smokers, the primary goal is to gain insights into the functioning of the human brain. There is no expert knowledge, neither about the full task nor about features that may be helpful. Practitioners in the field even argued that this task was impossible to learn. However, we successfully showed in our earlier work that it *is* possible to predict significantly better than random guessing. Therefore there *are* differences between the classes and we want to find them.

In the current work, we make four main improvements:

1) We improve our former results [4] by adding 1D-Convolutions to automatically scale inputs channel-wise.
2) We use the idea of residual networks to add identity mappings [9]. This was originally invented for pictures and we adapt the idea for time series.
3) We extend our testing, making better use of the available data to obtain results similar to a real world application.
4) We extend our analysis by performing a validation on subjects, in order to gain better insights into the data.

III. Residual Neural Networks

Neural Networks are predictive models, which use learning samples to iteratively adapt their parameters. A layered neural network usually consists of an input layer, one or more hidden layers and an output layer. Each layer can be interpreted as a transformation into a new feature space. The number of dimensions and the kind of mapping is determined by the network's designer, while the concrete transformation is automatically learned during the backpropagation process that minimizes the prediction error, see e.g. [12].

Dense layers. In a dense layer each neuron has weighted connections to each neuron in the preceding layer. Dense layers have many parameters and therefore a high computational power, but they are also prone to overfitting the data.

Convolutional layers. Convolutional layers use shared weights and apply them at several local parts of the input [6], thus detecting specific patterns at arbitrary positions in the input. The resulting output shows the pattern's strength of activation for each local part. Scherer [18] applies a Max-pooling operation after the convolutions. This operation aggregates local patterns by taking the maximal salience, discarding the others. The combination of convolution and Max-pooling was

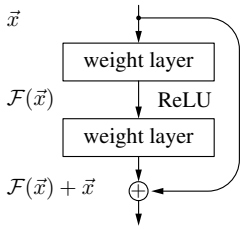

Fig. 1. Residual Block, skipping two weight layers

very successfully applied [11] e.g., if one is only interested in which object is in a picture, but not where this object is.

An alternative to Max-pooling is a strided convolution [20]. The stride specifies for each dimension of the convolution how far the filter is moved before it is applied again. A stride of 2 makes sure that the convolution is only applied at every second possible position in the given dimension.

Residual connections. The approach of residual connections was very successful for images and won 1st places in the ILSVRC and COCO 2015 competitions. The motivating principle is: Consider a layered network N, to which we add another hidden layer, which can learn weights to produce either the identity or some other function, thus creating the network N'. Note that generating the identity requires the same number of neurons in two consecutive layers. Because each solution represented by N can also be represented by N', we expect N' to perform at least as well as N. While this is correct in theory, experiments show [9] that the network struggles to learn the identity in the added layer, which causes worse results. To tackle this problem, we add a *skip connection* which is untrainable and passes on the data unchanged (identity mapping), bypassing one or more trainable layers. The two paths are recombined by adding up their values.

A schematic view of a *residual block* is depicted in Fig. 1. The processing block computes a mapping $\mathcal{F}(\vec{x})$, the skip connection (on the right) passes $\vec{x}$ unchanged to the output of the block, where the mapping and the unchanged input are summed so that the entire block computes $\mathcal{F}(\vec{x}) + \vec{x}$. The skip connection does not change the set of possible outcomes of the residual block, but merely changes the default output. Forwarding the identity function works fine when the number of input and output neurons is equal. When the output is smaller — let's say it is half the size of the input — we can again apply Max-Pooling or strided convolutions. Max-Pooling always keeps the strongest signal, but it varies depending on the signal; strides ignore a part of the signal but always pass the same part of the input to the next layer.

Adaptations: from pictures to EEG signals. For digital images, the data consist of two congenerous spatial dimensions (height and width), which is why 2D-convolutions are applied. A 2D-convolution with a filter size of 1x1, applied on a gray-scale image, performs a linear transformation, which is applied to all pixels. This adjusts the brightness of the image. Although this only adds very few weights to the model, these convolutions improve the performance.

We adapt this idea for EEG data: EEG data could also be represented in two dimensions, time and channel. While the order is clear for time, channels are measured on the scalp, which is why there is no natural linear ordering for them. Therefore, we apply a 1D-Convolution with one filter along the time axis for each channel. This layer adjusts the amplitude of each channel individually to cope with the problem of noise, influencing the signal amplitude of a whole channel. Because the visual inspection after preprocessing did not show strong noise of this kind, we do not expect a strong effect.

IV. Data

Our data stems from a study conducted between 2014 and 2015 at the Leibniz Institute for Neurobiology in conjunction with the Clinic for Neurology at the University of Magdeburg. Written, informed consent was obtained from all participants. The original study [5] used EEG data that were time-locked to the onset of smoking-related and non-smoking-related images. Here, the authors observed that smokers directed their attention away from the smoking-related images (i.e., avoided them), regardless of their state of craving. Craving did, however, have an impact on the overall state of arousal, with subjects showing a larger P1 (an early sensory-related component). After the task-based data had been acquired, the resting-state signal was measured in the same subjects for 9.5 minutes. These are the data we use to investigate resting state differences of craving smokers, non-craving smokers, and non-smokers. The present study includes EEG measurements from 28 smokers and 9 non-smokers. Each non-smoker was measured once, while each smoker was measured twice: once craving (after an abstinence of at least 4 hours) and once non-craving, having smoked directly before the measurement started.

A. EEG Measurement

EEG measures changes in potential (with respect to a reference, in our case, the right mastoid) over time. The temporal resolution is quite high, being on the order of milliseconds, whereas the spatial resolution is rather sparse, with electrodes being both sparsely distributed over the scalp (see Fig. 2 for an example of the layout used here), and with other issues such as volume conduction, which smears out the signal in space. Although EEG presents a great opportunity to record activity that is closer to cellular transmission rates than functional magnetic resonance imaging (fMRI), EEG also has drawbacks in that the electrodes pick up any source of electrical activity, whether neural or external noise, and therefore the signal is not as pure as one might hope (see in depth description, below, for sources of noise). Here, the EEG data were acquired using 32 electrodes at a sampling frequency of 508 Hz and it was bandpass-filtered online from DC to 50 Hz, with low impedances (5 kΩ) maintained for each channel.

B. Preprocessing

EEG electrodes measure signals arising from the brain, but also pick up signals from other sources. As we are interested in the brain signal only, signals from other sources are noise.

This noise has several causes: muscle-related activity such as respiration, heartbeat, or eye movements, sweat that can impede the connection between the scalp and the electrode, often adding a slow-drift to the signal, and electrical interferences at 50 Hz stemming from the alternating current of the power supply. To preprocess the EEG data, we applied a low-pass filter at 30 Hz and a high-pass filter at 0.5 Hz. This removed high frequency noise, including power line interference, some muscle artifacts, slow-drift related movements, respiration and sweat artifacts. Removing physiological artifacts needs to be done semi-automatically, which we did using Independent Component Analysis (ICA). This algorithm creates independent components using the signals measured at the electrodes. We manually selected and removed components containing eye blinks, eye movements and heart beat. The selection of the components was conservative, as the removal of a noise-like component also removes some brain signals, and it is not possible to remove only the noisy parts of a component.

The ICA successfully removed most of the artifacts, but also created high frequency noise. To mitigate this noise, we filtered again, keeping only the signal between 0.5 Hz and 30 Hz. Manually reviewing the resulting signals, we found remaining eye artifacts in channels Fp1 and Fp2 and muscle artifacts in channels T7 and T8 (cf. Fig. 2: the mentioned channels are marked in red). To prevent these artifacts from biasing our results, we excluded these channels from all subjects in subsequent analysis. One smoker had moved so much during both sessions that we were unable to remove this noise and therefore excluded these two measurements completely.

We normalized each channel to have a mean of zero, but we kept all the variances. Different variances may be caused by the brain and should therefore be considered in the training process, or they may be caused by noise in the measurement and should be filtered out. As we cannot be sure about the source, we decided to keep them. The entire preprocessing was performed using the MNE framework [8]. For the experimentation, we used the Keras [2] framework.

V. Experimentation

A. Bootstrapping

Most importantly we use bootstrapping: we split our measurements into snippets of length $2^{10} = 1024$ (approximately two seconds) and sample from them with replacement. The model is trained to perform predictions for these snippets. This has two advantages: it decreases the (time) dimensionality and increases the number of training samples. Unfortunately, though, snippets from the same measurement are not independent of each other [13]. In order to still be able to get reliable results, we make sure that no snippets from the same measurement are used in both the training set and the test set at the same time. Note also that the i.i.d. assumption does not hold when taking samples from the same subject's craving measurement for training and non-craving measurement for testing. This led to results significantly worse than guessing in preliminary experiments. We assume that the model had learned person-specific patterns, which caused incorrect predictions on the test set. Allowing overlapping snippets, we can generate up to $\approx$ 289.000 samples per measurement; without overlap, we obtain 289 samples. Since the latter seemed too few, we decided to use overlapping samples for training and chose to take samples randomly. Note that the parts near the start and the end have a smaller chance of getting selected.

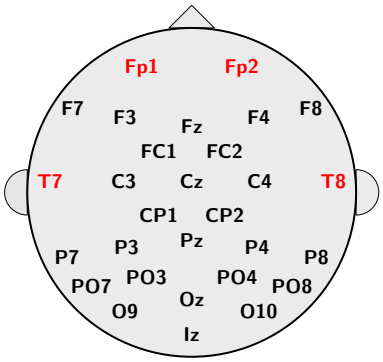

Fig. 2. Overview of the electrode positions

We repeatedly sample snippets to generate batches of size 200. In order to minimize the variance induced by the batch generation, we make sure that the distribution within the batch differs as little as possible from the whole data set.

B. Validation and Training

For the validation, we use random shuffle split cross validation. This method has the advantage that in can be repeated arbitrarily often. We repeat each of our experiments at least 100 times with different random seeds. We use seven of nine measurements for training and the remaining two for testing. That is, with 27 smokers and nine non-smokers overall, a training batch uses 21 smokers and seven non-smokers. As there are two measurements for each smoker, we get 21 samples of craving smokers, 21 samples of non-craving smokers and seven samples of non-smokers — a total of 49. For a batch, we sample four snippets from each measurement plus the remaining four randomly, such that they vary the least. As quality metric, we measure class-balanced prediction accuracy in all of our experiments.

C. Testing

The testing scheme works similar to training. We use the craving and non-craving measurements of six smokers and two measurements of non-smokers as the basis for a batch.

As a first evaluation, we use one batch sampled *like training* for testing. This scheme uses about eight seconds per measurement in the evaluation, which is not realistic as there are 9.5 minutes available. As this evaluation was used earlier [4], we apply it here mainly for comparison.

As a second evaluation, we use the available data more thoroughly: We sample one batch of 200 snippets for each measurement — which corresponds to $\approx$6.7 minutes. In analogy to training, there are two possibilities for the sampling: repeatedly sample from random locations or apply a sliding

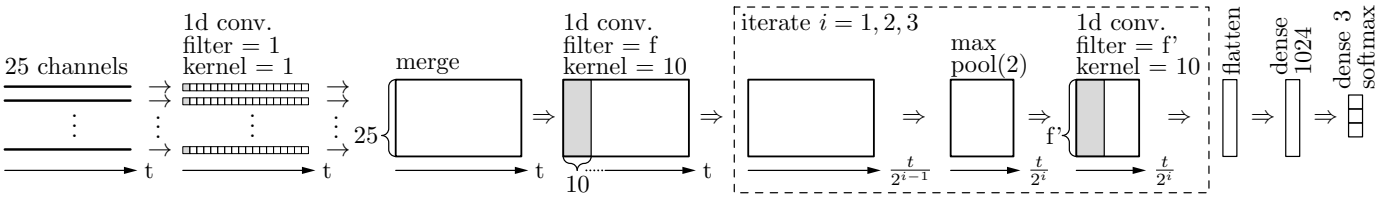

Fig. 3. Network structure for network *(I)* with the extra convolution before the merge

window. A sliding window has the advantage that the snippets cover most of the measurement, unless the windows overlap. A disadvantage is that the procedure is not randomized, which might influence the prediction.

In random sampling with replacement, parts of the measurement may be used multiple times. There is no fixed static choice to influence the results, but we probably cover less of the measurement. We chose to perform both methods to investigate the effect of the static sampling.

When sampling extensively, but still using models working on snippets of two seconds, we get 200 predictions per measurement, which need to be combined into one prediction. If the predictions are particularly *diverse*, that is, if they make their errors for different sample cases, it can be proven that the combined predictor performs at least as well as the worst single predictor. The work of Dietterich [3] explains several techniques to generate ensembles of classifiers.

We aggregate predictions in five different ways. The first is *majority voting*: The model performs a prediction for each snippet, casting a vote for the predicted class, and finally predicting the class with the largest number of votes. A drawback of this approach is that it uses the winner-takes-all principle: The most probable class gets the vote, the others get nothing. In cases when the best two classes have probabilities close to each other, this appears to be inappropriate. We handle this situation by allowing to abstain from voting (*vote with abstain*): We do not count a vote if the two best predictions are close; i.e., if the values differ by less than 0.1.

Another way to avoid the winner-takes-all is to *sum the probabilities* and then predict the class with the highest value. Thus, we reduce the effect of the repeated application of the winner-takes-all by applying it only once at the end.

Snippets with similar probabilities for the highest classes are generally hard to predict. They could cause a tendency of the summed probabilities predictor towards random guessing. We reduce this effect by weighting the sum such that high probabilities are higher weighted. Therefore, we square the probabilities before we sum them up (*squared sum*).

In principle, any convex function can be applied to increase the weight of the predictions with high probabilities. The most extreme form takes only the one prediction with the highest probability, which we call *max aggregation*.

D. Used network structures

We perform experiments with three basis networks *(I)*, *(II)* and *(III)* to investigate if residual blocks improve predictability, and further, if strided convolutions or max pooling performs better. Basis network *(I)* uses no residual connections and is used for comparison. It uses three times alternating 1D-convolution and max-pooling layers as feature detectors, followed by one fully connected (dense) layer and a final dense layer with softmax activation performing the predictions. The convolutional layers use 32, 64 and 128 filters each of size 10, the pooling factor is always two. The softmax layer uses three neurons – one for each class. Basis network *(II)* is similar to *(I)*, but each convolutional layer is replaced a residual block with two convolution layers per residual block. Filters and pooling factors are the same. Basis network *(III)* uses residual blocks as well. As explained in Section III, as alternative for max-pooling we apply convolutions with stride 2, here.

For each of the basis networks, we use three variations: In the *original* variation of the network, the dense layer before the softmax has 1024 neurons. It is used as comparison for the *extra conv.* variation, where a channel-wise 1D convolution with one filter and filter size of one is used. The *extra conv.* variation of network *(I)* is depicted in Fig. 3 . The third variant uses *extra conv.* and a *smaller Dense* layer with 128 neurons.

VI. Results

A. Model comparison

The results of our experiments are shown in Table I. We performed the sampling 'like training' to compare the results with earlier work [4] on the same data set. Starting with model *(I)*, we see the resulting median (0.3326) is only at the level of guessing. Compared to earlier work's 37.6%, it clearly performs worse, but also uses less filters in its convolutions. When adding the channel-wise convolutions, we reach 39.10% — although the underlying network has fewer parameters. Even when reducing the size of the last layer, we still achieve 37.10%. Although the visual inspection after preprocessing did not show channel-wise noise, the channel-wise convolution causes significantly increased predictions, for model *(I)* even when the smaller dense layer is used.

Model *(II)* uses residual blocks, and reduces the number of neurons by a factor of two after each of the three residual blocks, using max pooling layers. Without channel-wise convolutions, it performs better than the bigger model *(III)* and reaches 37.50%. But it benefits less from the channel-wise convolution, and achieves only 38.95% with it. With the smaller dense layer for the prediction, it only yields 36.77%.

Model *(III)* uses residual connections with strides and achieves 39.14%, even without the channel-wise convolutions. With the convolutions it becomes even better: 39.88% and

TABLE I
OVERVIEW OF MODEL AGGREGATIONS

			original		extra conv.		extra conv. + smaller Dense	
Model	**Sampling**	***Aggregation***	***Median***	***Mean***	***Median***	***Mean***	***Median***	***Mean***
(I)	like training	None	0.3326	0.3532	0.3910	0.3853	0.3710	0.3790
(I)	repeated random	majority vote	0.3333	0.3738	0.4444	0.4266	0.3888	0.4038
(I)	repeated random	vote with abstain	0.3333	0.3738	0.4444	0.4216	0.3888	0.4050
(I)	repeated random	max	0.3888	0.3877	0.3888	0.4161	0.3888	0.3755
(I)	repeated random	sum	0.3333	0.3711	0.4444	0.4244	0.3888	0.4055
(I)	repeated random	squared sum	0.3333	0.3755	0.4444	0.4300	0.3888	0.4061
(I)	shifted window	majority vote	0.3333	0.3683	0.4444	0.4294	0.3888	0.4100
(I)	shifted window	vote with abstain	0.3333	0.3688	0.4444	0.4333	0.3888	0.4072
(I)	shifted window	max	0.3333	0.3716	0.4444	0.4222	0.3888	0.4027
(I)	shifted window	sum	0.3333	0.3722	0.4444	0.4322	0.3888	0.4077
(I)	shifted window	squared sum	0.3333	0.3705	0.4444	0.4361	0.3888	0.4083
(II)	like training	None	0.3750	0.3869	0.3895	0.3876	0.3677	0.3714
(II)	repeated random	majority vote	0.3888	0.4105	0.3888	0.4161	0.3888	0.3950
(II)	repeated random	vote with abstain	0.3888	0.4116	0.3888	0.4172	0.3888	0.3927
(II)	repeated random	max	0.3888	0.4066	0.4444	0.4272	0.3888	0.3994
(II)	repeated random	sum	0.4166	0.4150	0.3888	0.4222	0.3888	0.3938
(II)	repeated random	squared sum	0.4166	0.4177	0.3888	0.4188	0.3888	0.3927
(II)	shifted window	majority vote	0.4444	0.4138	0.3888	0.4227	0.3888	0.3977
(II)	shifted window	vote with abstain	0.4444	0.4200	0.4444	0.4205	0.3888	0.3927
(II)	shifted window	max	0.4444	0.4050	0.3888	0.4216	0.3888	0.4016
(II)	shifted window	sum	0.4444	0.4161	0.4444	0.4316	0.3888	0.3972
(II)	shifted window	squared sum	0.4444	0.4222	0.4444	0.4300	0.3888	0.3988
(III)	like training	None	0.3914	0.3945	0.3988	0.4006	0.3867	0.3895
(III)	repeated random	majority vote	0.4444	0.4316	0.4444	0.4355	0.4166	0.4338
(III)	repeated random	vote with abstain	0.4444	0.4305	0.4444	0.4361	0.3888	0.4355
(III)	repeated random	max	0.4444	0.4150	0.4444	0.4238	0.3888	0.3938
(III)	repeated random	sum	0.4444	0.4338	0.4444	0.4377	0.3888	0.4350
(III)	repeated random	squared sum	0.4444	0.4327	0.4444	0.4338	0.3888	0.4311
(III)	shifted window	majority vote	0.4444	0.4366	0.4444	0.4438	0.4444	0.4366
(III)	shifted window	vote with abstain	0.4444	0.4366	0.4444	0.4411	0.4166	0.4388
(III)	shifted window	max	0.4444	0.4122	0.4444	0.4322	0.3888	0.4111
(III)	shifted window	sum	0.4444	0.4433	0.4444	0.4477	0.4444	0.4394
(III)	shifted window	squared sum	0.4444	0.4433	0.4444	0.4455	0.4166	0.4383

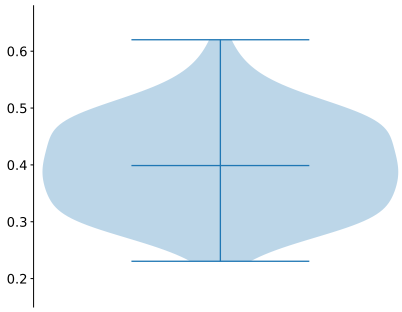

Fig. 4. Models (III), predictions as trained

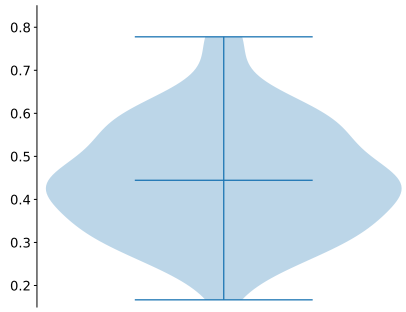

Fig. 5. Models (III), predictions for shifted window

finally even 40.06% when considering the mean. As this is the best model, we depict the distribution of the results in a violin plot in Fig. 4. The picture shows a symmetrical distribution between weighted accuracies of 23% and 63% without any outliers. It is not only the model which performs best on average, but also seems to be robust in its predictions.

Summarizing, we see that the bigger models perform better. The channel-wise convolution adds only very few extra parameters, but causes much better predictions. In all cases, the mean value was higher, in most cases the median value was increased. The filters added before the dense layer seem to cope with the smaller dense layer, which is plausible as the number of weights between these layers is the product of the two layer's sizes. We were able to improve our former baseline from 37.6% to nearly 40% by using residual models with strided convolutions.

B. Comparing aggregation and sampling methods

The two sampling methods show similar results. In most cases, they are even identical for the median, while for the mean they differ by less than 0.3% on average. Shifted windows seem to perform a bit better, especially for models *(II)* and *(III)*. Reasons might be lucky starting positions or a slightly better generalization due to a higher coverage. The

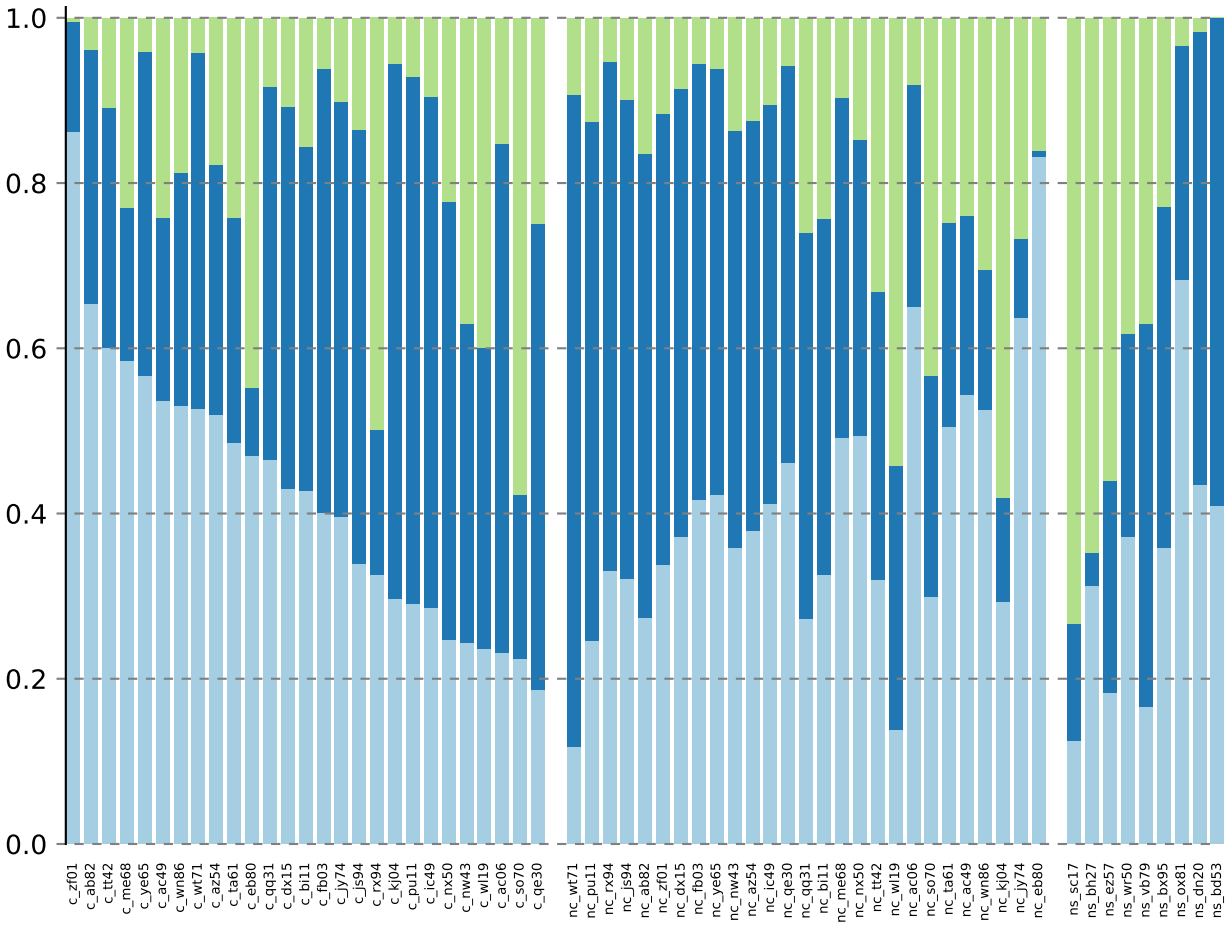

Fig. 6. Bar chart showing the average predicted probabilities for each measurement and all three classes for model *(III)* using the *extra conv*. Light blue represents the predictions of craving, dark blue means non-craving and green stands for non-smoker. The actual class is given by the measurement's prefix.

max aggregation seems to be least robust. Though yielding the highest mean for model *(I)* without channel-wise convolutions, it exhibited the worst performance with them. As robustness is important we advise against max aggregation.

Sum and squared sum show no clear trend. The assumption that squaring may be beneficial is not supported by our results. Findings are similar for majority vote and vote with abstain: Abstaining seems to occur infrequently, hence the results hardly differ and surprisingly there is no clear trend visible.

Comparing the 'real world test' (aggregate over snippets) to the 'single snippet test' shows the expected results: Using more measurements and aggregating the results is clearly better than just using one batch of samples. Furthermore, we find the expected general trend: The better the model if it is tested like it was trained, the better is the performance of the aggregated predictions. The medians increase to a new record of 44.44% for many models — surprisingly even for some experiments of model *(III)* where the smaller dense layer is used. A violin plot for the best model with aggregation is shown in Figure 5. Its predictions even have a mean (44.77%) exceeding the median (44.44%). Compared to the base models' predictions (Fig. 4), the poorest performance is worse by 5% but the best performance is better by more than 15%.

C. Patient-wise evaluation

The model estimates the probability of belonging to each class for a given snippet of data. In the standard evaluation, we do not use the probabilities, but predict the class with the highest probability. In contrast to this, we now consider the probabilities directly. Fig. 6 shows the average probabilities predicted for each measurement and each class. Each bar represents one measurement. They are grouped, craving *c* on the left, followed by non-craving *nc* and non-smoker *ns*, as indicated by the prefix. Within each group, subjects are ordered by average prediction quality. The leftmost bar shows the best-predictable measurement from the craving class. When craving, subject *zf01* is correctly predicted with about 85%, incorrectly considered to be non-craving with about 15%, and mistakenly considered to be non-smoker with less than 1%. When *zf01* is non-craving our correct prediction rate is more than 50%, which makes *zf01* the best predictable smoker. The worst predictable craving subject *c_qe30* was mistakenly considered to be non-craving with more than 50%, while craving has an average probability of less than 20%. The non-smokers *ox81*, *dn20* and *bd53* are misclassified in most of the cases, and only predicted to be non-smokers with less than 6%.

Subjects *so70* and *wl19* look like non-smokers to the model when craving and non-craving. *ac06* looks like being non-craving when actually craving and vice versa. One might guess that the measurements were incorrectly labeled, but the labels are correct. Thus we consider this person to be an outlier. *eb80* seems to confuse the model. When craving it is often thought to be non-smoker, when non-craving the model guesses non-craving with less than 2%; it is mostly misclassified to be craving. This is the only subject for which we can conjecture as to why it may behave differently: it is the only left-handed subject in the dataset.

Overall, we see that the predictions show a high variance between subjects and measurements. We hypothesize that our model still considers many person-specific or noise patterns and only few which are actually related to being smoker or being craving. However, this hypothesis can not be clarified with the current data. To cope with this problem, we suggest the acquisition and analysis of many more measurements. This would also open the opportunity to investigate whether the handedness plays a role for prediction models.

VII. Conclusion and Future Work

In this work, we created an improved neural network model to distinguish non-smokers, craving smokers and non-craving smokers by their EEG signals. This task is especially difficult: it is unknown which features are important, only few measurements are available, and the data has high dimensionality. Asked practitioners even claimed it was impossible to solve.

Yet, we were able to significantly improve over our earlier models, which already performed better than random guessing, by adding channel-wise 1D-convolutions and residual connections, which leads to class-balanced prediction accuracies of nearly 40%. Given that we use less than 50 measurements overall, this is an amazing result. Compared to previous work, we improved performance by 2.3% percentage points.

Furthermore, we extended our evaluation to make it more realistic and performed predictions on ensembles of 200 snippets per measurement. Sampling them randomly performs slightly worse than a moving window approach. The predictions were aggregated by five different aggregation schemes which performed similarly, only max prediction was less robust. The aggregated predictions reached 44.4% mean and median accuracy, a significant and highly promising result.

Finally, we analyzed our best-performing model and checked the performance for each measurement. The variance over measurements is fairly high, which indicates that our model still learns patterns specific for a subject or a measurement. Several subjects were especially hard to predict. One of the worst predictable smokers, subject `eb80` has an average correct probability of less than 2% when craving. As this subject is the only one in our data set which is left-handed, we hypothesize that handedness has a non-negligible effect. To obtain more reliable results and to further investigate the relevance of handedness, we suggest to collect many more measurements.

In future work, we will visualize the learned features to gain insights about the distinguishing neural patterns. We hope that this visualization enables us also to see what makes some of the measurements so unpredictable. Additionally, we will take into consideration data from questionnaires obtained for each subject on the relative level of nicotine addiction, and the amount of subjective craving prior to the measurement.

References

[1] Barbara B. Brown. Some characteristic EEG differences between heavy smoker and non-smoker subjects. *Neuropsychologia*, 6(4):381–388, 1968.

[2] François Chollet et al. Keras, 2015. https://github.com/fchollet/keras.

[3] Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[4] Christoph Doell, Sarah Donohue, Cedrik Pätz, and Christian Borgelt. Training neural networks to distinguish craving smokers, non-craving smokers, and non-smokers. *Symposium on Intelligent Data Analysis (IDA)*, 2018, in press.

[5] Sarah E Donohue, Marty G Woldorff, Jens-Max Hopf, Joseph A Harris, Hans-Jochen Heinze, and Mircea A Schoenfeld. An electrophysiological dissociation of craving and stimulus-dependent attentional capture in smokers. *Cognitive, Affective, & Behavioral Neuroscience*, 16(6):1114–1126, 2016.

[6] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *arXiv:1603.07285*, 2016.

[7] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv:1412.6572*, 2014.

[8] Alexandre Gramfort, Martin Luessi, Eric Larson, Denis A Engemann, Daniel Strohmeier, Christian Brodbeck, Roman Goj, Mainak Jas, Teon Brooks, Lauri Parkkonen, et al. MEG and EEG data analysis with MNE-Python. *Frontiers in Neuroscience*, 7, 2013.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European Conference on Computer Vision*, pages 630–645. Springer, 2016.

[10] Verner Knott, Meaghan Cosgrove, Crystal Villeneuve, Derek Fisher, Anne Millar, and Judy McIntosh. EEG correlates of imagery-induced cigarette craving in male and female smokers. *Addictive Behaviors*, 33(4):616–621, 2008.

[11] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[12] Rudolf Kruse, Christian Borgelt, Christian Braune, Sanaz Mostaghim, and Matthias Steinbrecher. *Computational intelligence: a methodological introduction*. Springer, 2016.

[13] Jean-Yves Le Boudec. *Performance Evaluation of Computer and Communication Systems*. EPFL Press, Lausanne, 2010.

[14] Chuck Lorre and Bill Prady. The Big Bang Theory, season 4, episode 21. Columbia Broadcasting System (CBS), New York, NY, USA, 2011.

[15] Mufti Mahmud, Mohammed Shamim Kaiser, Amir Hussain, and Stefano Vassanelli. Applications of deep learning and reinforcement learning to biological data. *IEEE transactions on neural networks and learning systems*, 29(6):2063–2079, 2018.

[16] David Premack. Human and animal cognition: Continuity and discontinuity. *Proceedings of the National Academy of Sciences*, 104(35):13861–13867, 2007.

[17] Olga Rass, Woo Young Ahn, and Brian F. O'Donnell. Resting-state EEG, impulsiveness, and personality in daily and nondaily smokers. *Clinical Neurophysiology*, 127(1):409–418, 2016.

[18] Dominik Scherer, Andreas Müller, and Sven Behnke. Evaluation of pooling operations in convolutional architectures for object recognition. In *Artificial Neural Networks–ICANN 2010*, pages 92–101. Springer, 2010.

[19] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[20] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. *arXiv:1412.6806*, 2014.