The Realtime Assessment of Mental Workload by Means of Multiple Bio-Signals

Masterthesis Report

Methodology and Statistics for the Behavioral, Biomedical and Social Sciences

Utrecht University

Bart-Jan Boverhof, 6000142

Thesis Supervisor

Prof.dr.ir. B.P. Veldkamp

Date

January 7, 2021

1 Introduction

The topic of mental workload is a widely studied phenomenon across a variety of different fields, amongst others the field of ergonomics (Young, Brookhuis, Wickens, & Hancock, 2015), human factors (Pretorius & Cilliers, 2007) and neurosciences (Shuggi, Oh, Shewokis, & Gentili, 2017). A widely utilized definition of mental workload, hereafter referred to as simply "workload", is the demand placed upon individuals whilst they carry out a certain task. As rightfully pointed out by de Waard and te Groningen (1996), the aforementioned definition is lacking, for it defines workload solely as a phenomenon external to the individual. Workload needs to be recognized as a person-specific construct. The amount of perceived workload ushered by a given task may vary across individuals due to how well one is able to cope with that particular task (de Waard & te Groningen, 1996).

A commonly employed method for the assessment of person-specific workload is the well established NASA-Task Load Index questionnaire. This questionnaire inquires into the amount of perceived workload, and embodies six different dimensions (Hart, 2006). Such an assessment is usually conducted post-experiment, which might in certain situations be undesirable. Consider for example an experiment in which it is aimed to assess workload of a pilot in flight. An insightful learning approach towards such an experiment would be to measure the degree of perceived workload at different phases of the flight. However, only after the flight is concluded, a measurement in the form of a questionnaire can be administrated. In such a situation, utilizing a post-experiment assessment well after the action took place is prone to generate biases, such as for example the observer-bias, which advocates that participants in an experiment tend to overexaggerate the treatment effect, i.e. the amount of perceived workload, when having to report it post-experiment (Mahtani, Spencer, Brassey, & Heneghan, 2018).

An alternative approach to the assessment of workload is to collect physiological bio-signals during the experiment, and utilize these inputs to predict the degree of experienced workload. Examples of such bio-signals, hereafter referred to as "modalities", include techniques such as electroencephalography, eye-tracking, galvanic skin response, functional near-infrared spectroscopy, etc. The advantage of utilizing multiple bio-signals is that complementary information streams, each stemming from a different modality, can all be interpreted simultaneously (Ramachandram & Taylor, 2017). Doing so poses the potential of yielding a rich and multifaceted prediction of a mental construct such as workload. However most importantly, it is renders possible to train a separate model for each individual, with which can be catered towards the person-specific nature inherent to workload, the importance of which was pointed out by de Waard and te Groningen (1996). This approach, however, comes at the cost of an increase in complexity. This resides in the need to construct a complex framework that inputs the data from each of the utilized modalities, and ultimately outputs a prediction

Three modalities are utilized in the current research. The first of which is the technique of electroencephalography , hereafter referred to as ”EEG”. The second is the technique of galvanic skin response, hereafter referred to as ”GSR”. Lastly, the third technique is called photoplethysmography, hereafter referred to as ”PPG”. The objective of the current study is to construct a framework with which real-time prediction of a mental construct such as workload can be managed, and to which modalities of choice can easily be added in future research endeavors. Consequently, one of two design principles on which the architecture of the framework reclines is the principle of modularity. Modularity refers to the extent to which different modalities can freely be added and/or removed towards the framework, without the necessity of re-architecting and rebuilding it entirely. The second design principle is the principle of generalizability, prescribing that the framework should not merely be utilizable for the classification of workload, but for the classification of other mental constructs as well.

A deep-learning approach towards the construction of a real-time multi-modular framework is adopted. Considering the complexity and sheer size of such a deep-neural network, an important point of attention is prediction speed. The challenge in real-time prediction with deep-learning is often not reaching adequate performance, but rather to attain adequate speed of prediction. Deep neural networks easily constitute thousands of calculations to be made simultaneously, which is even more true for a multi-modular approach such as the current. In order for real-time prediction to work, prediction cannot take too long, for otherwise it is not real-time anymore and the previously delineated benefit of a real-time approach dissipates. As a consequence, multiple networks are considered and contrasted in terms of performance and their ability to predict in real-time. Firstly, for each of the three modalities individually, a single-modular network is constructed. and assessed. Secondly, a multi-modular network is architected combining all three modalities into one. These networks were created in Python, by utilizing the deep-learning toolbox PyTorch (Paszke et al., 2017)

The aim of the current research is to explore the circumstances under which a multi-modular approach by means of deep-learning is capable of real-time classification of workload, whilst still ensuring ample and adequate performance. The objective is to provide a framework with which the previously delineated is possible, whilst satisfying the principles of modularity and generalizability. Ultimately, this line of research pursues the ability to conduct a dynamic experiment for multiple people simultaneously, and of which the state can be altered in real-time.

2 Methods

2.1 Related Work

The following sections provide an overview of preceding research on the most optimal network architecture for each single-modular network. Subsequently, the most feasible architecture for the multi-modular framework in its entirety is explored, with particular attention placed upon the data fusion strategy and a range model optimization techniques.

2.1.1 Electroencephalography (EEG)

EEG constitutes a technique that detects electrical activity in the brain by using electrodes. EEG is a commonly utilized method within the field of workload assessment. An overview of the complete literature on EEG applications with deep-learning was presented by Craik, He, and Contreras-Vidal (2019), who reported a total of 16 % of all available papers to deal with workload specifically. With regards to these EEG applications, it was reported that studies mostly used deep belief networks and convolutional neural networks, hereafter referred to as ”ConvNets”. One of these approaches is advised consequently (Craik et al., 2019).

For EEG classification, Tabar and Halici (2016) proposed a combination of a ConvNet with a stacked auto-encoder network, hereafter referred to as ”SAE network”. The input layer was specified to feed into a convolutional layer with the objective of learning the filters and network parameters. The output of this convolutional layer was subsequently specified to feed into the SAE part of the network, designed to include an input layer, six hidden layers and an output layer. A classification accuracy of 90 % was acquired with this network (Tabar & Halici, 2016).

Research by Schirrmeister et al. (2017) contrasted the performance of several ConvNets against the widely utilized baseline method for EEG classification, filter bank common spatial pattern, hereafter referred to as ”FBCSP”. A deep ConvNet, a shallow ConvNet, a deep-shallow hybrid ConvNet and a residual ConvNet were contrasted with an FBCSP. Both the deep and shallow ConvNets were found to reach at least similar, and in some regards better classification results as compared with the FBCSP baseline approach. Altogether, a deep ConvNet with four convolutional-max-pooling blocks was found to perform best, exhibiting an accuracy of 92.4 % (Schirrmeister et al., 2017).

2.1.2 Galvanic Skin Response (GSR)

GSR is a technique that measures sweat gland on the hands, hereby inferring arousal. GSR activity is known to be strongly correlated with perceived workload, indicating the utility of GSR for workload classification (Shi, Ruiz, Taib, Choi, & Chen, 2007).

Sun, Hong, Li, and Ren (2019) explored the most suitable model design for the classification of several emotional states with GSR. Various models were explored, amongst others a support vector machine, a ConvNet and a long-short-term-memory network, hereafter referred to as "LSTM network". Additionally, the feasibility of a hybrid network, combining both the ConvNet and LSTM approaches, was explored. This aforementioned hybrid model was found to perform best, exhibiting an accuracy of 74% (Sun et al., 2019).

Dolmans, Poel, van 't Klooster, and Veldkamp (2020) took a multi-modular approach to workload classification, and designed a variant based on the previously delineated CovNet-LSTM approach for GSR. The performance of this model was contrasted with a network consisting solely of fully connected dense layers. Conform with findings by Sun et al. (2019), the hybrid model was found to perform best, displaying an accuracy of 82 % (Dolmans et al., 2020). The model architecture as utilized by Dolmans et al. (2020) deployed two convolutional max-pooling blocks and two LSTM layers.

2.1.3 Photoplethysmography (PPG)

PPG is a technique utilized to measure volumetric changes in blood peripheral circulation, from which it is possible to derive heart-rate. Zhang et al. (2018) contrasted several approaches towards workload classification, each utilizing a different modality. PPG was found to perform among the best, explaining why PPG constitutes one of the most widely utilized modalities in workload classification.

Work by Biswas et al. (2019) explored a deep-learning approach towards PPG classification, with the objective to perform both bio-metric identification and obtain heart rate information. Exceptional results were realized with a deep neural network, attaining an average accuracy of 96 % (Biswas et al., 2019). This performance was realized with a ConvNet-LSTM hybrid, incorporating two convolutional max-pooling blocks followed by two LSTM layers.

2.1.4 Multi-modular Fusion Strategy

When conducting a multi-modular deep-learning approach, the information streams stemming from the three modalities are required to be combined, i.e. ”fused”, at a certain point in the network, in order to ultimately result in a single classification of workload. Fusion can be done conforming different strategies. Several strategies as proposed by Ramachandram and Taylor (2017) have been considered.

Early, or data-level, fusion constitutes an approach that fuses data sources before being fed into the network. Techniques that manage this include for example principle-component analysis and factor analysis. Early fusing is usually proven to be quite challenging, residing in the fact that data streams stemming from different modalities often differ in dimensionality and sampling rate. In addition, when taking an early fusion approach, the oversimplified assumption of conditional independence is made implicitly. This assumption is unrealistic in nature, for data stemming from different modalities are expected to be correlated in practice (Ramachandram & Taylor, 2017).

Late, or decision-level, fusion refers to the process of aggregating the decisions of multiple separate networks, each applied towards every modality separately. In case the data sources stemming from the various modalities are either correlated or ultimately differ in their dimensionality, late fusion is a much more feasible approach as contrasted with early fusion (Ramachandram & Taylor, 2017).

Lastly, intermediate fusion is the most widely employed fusion strategy for multi-modular deep-learning problems. Data streams are usually fused by a concatenation layer, joining the outputs of the separately defined network parts of each modality. This results in a single joint deep neural network. Classification is managed based on the outputs from each of the single-modular network parts. Several ”higher-order layers” are usually defined in between the concatenation layer and the ultimate classification. The depth of the fusion, i.e. the specified number of higher-order layers, can be chosen conform to the specific circumstances, posing intermediate fusion to be the most flexible and therefore the most widely adopted fusion strategy (Ramachandram & Taylor, 2017).

Indeed, when consulting the literature, one is forced to conclude that intermediate fusion strategies are the most prevailing for multi-modular deep-learning approaches. When taking an intermediate fusion approach, the higher-order part of the network needs to be designed and established, for which several previous research endeavors are considered. Rastgoo, Nakisa, Maire, Rakotonirainy, and Chandran (2019) utilized a multi-modular

ConvNet approach, and fused the modalities by concatenation, followed with two LTSM layers, two dense layers and a softmax layer. A simpler approach is adopted by Han, Kwak, Oh, and Lee (2020), who utilized an intermediate fusion approach solely consisting of several fully connected dense layers, and ending with a soft-max layer. Lastly, Dolmans et al. (2020) took a relatively deep intermediate fusion approach, consisting of two dense layers, two convolutional layers followed by another two dense layers.

2.1.5 Model Optimization Strategies

The technique of batch normalization was originally proposed by Ioffe and Szegedy (2015), and is often a applied in deep-learning with the objective of stabilizing a network. Especially ConvNets are able to capitalize on this technique. It is beneficial to include a batch normalization layer after each convolutional layer, re-centering and re-scaling the input feeding into subsequent layers. When incorporating a batch normalization layer, it is recommended to do so before the specification of the activation function (Ioffe & Szegedy, 2015). An increase in accuracy for EEG classification was attained by Dolmans et al. (2020) and Schirrmeister et al. (2017) through specifying a batch normalization layer following each convolutional layer. Equally so, the best performing network for PPG as proposed by Biswas et al. (2019) included a batch normalization layer succeeding each convolutional layer.

Pooling layers are commonly employed in ConvNets, usually succeeding a convolutional layer with the purpose of decreasing the dimensionality. The objective of such layers are to merge similar features into one. For a more extensive elaboration on pooling, please consult LeCun, Bengio, and Hinton (2015). In their EEG ConvNets, both Schirrmeister et al. (2017) and Tabar and Halici (2016) specified a max-pooling layer after each convolutional layer. The network as proposed for GSR by Sun et al. (2019) incorporated a max-pooling layer after each but one of the convolutional layers. Lastly, the network as proposed for PPG by (Biswas et al., 2019) specified a max pooling layer after each convolutional layer.

Hyper-parameter optimization, hereafter referred to as "HPO", is a technique that can be utilized to optimize hyper-parameters such as learning rate and dropout. Substantial advancements in deep-learning have been attained by utilizing HPO, especially for ConvNets (Bergstra & Bengio, 2012). The Optuna toolbox provides a method for creating a parameter search space, from which values for the hyper-parameters can be sampled and optimization can be performed (Akiba, Sano, Yanase, Ohta, & Koyama, 2019).

2.2 Data

The subsequent section will provide an overview of the utilized data. Attention is primarily placed upon the experimental setup, the description of the respondents, the utilized devices for data collection and the synchronization process.

2.2.1 Experimental Setup

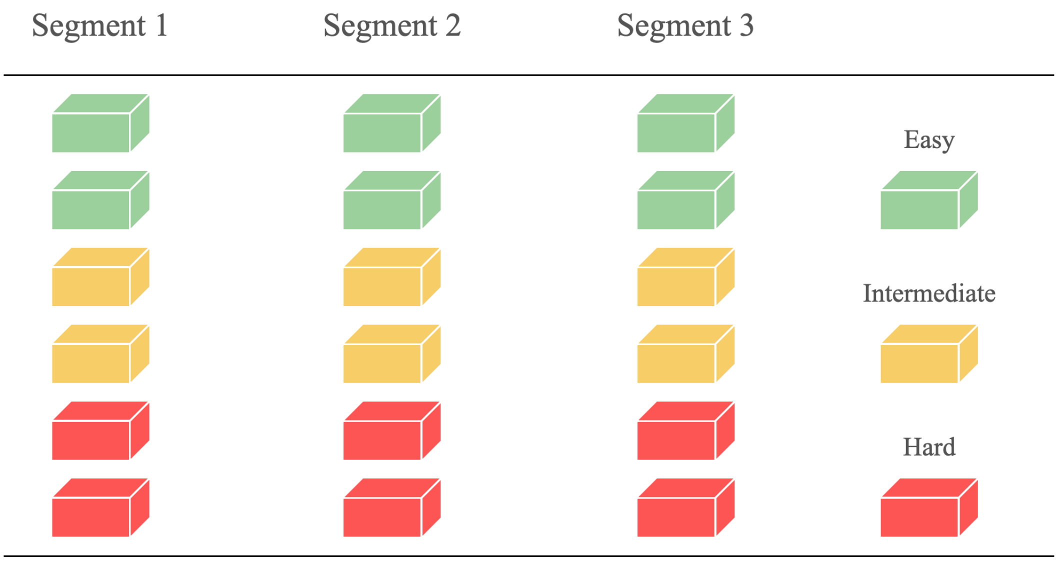
The experimental setting in which the data collection took place is the open-source spaceship video-game Empty Epsilon, in which respondents were asked to carry out tasks on a virtual spaceship (Daid & Nallath, 2016). This experiment was instituted by the Brain Computer Interfaces testbed, hosted by the University of Twente and carried out in cooperation with Thales group Hengelo. The experiment constituted three different segments, in each of which the respondent had to carry out tasks. These tasks were designed to evoke varying degrees of workload. Each segment consisted of six small sessions lasting roughly five to ten minutes. These sessions varied in difficulty, including two easy, two intermediate and two hard sessions per segment. Within each segment, the order in which the sessions were presented was randomized. The order in which the three main segments were administrated was not randomized. Between every three sessions, respondents were given a short two minute break. A schematic overview of the experimental structure is depicted as Figure 1.

After each of the 18 sessions, respondents were asked to fill in the TLX questionnaire, compromising of six questions each. This resulted in 18 filled in questionnaires per respondent. Each questionnaire inquired upon the degree to which the respondent experienced workload during the preceding session. These measurements have been utilized to label the data for network training purposes.

The first segment emulated a scenario in which hostile spaceships approached the respondent's spaceship. The respondent was required to quickly react by defusing the hostiles in order to survive. The increment in difficulty caused the process of defusing hostile spaceships to become more challenging, hereby aiming to cause an increase in workload. The second segment emulated a scenario in which the respondent had to navigate their spaceship trough space, gathering as many way-points as possible. Obstacles

Figure 1

Experimental Setup



around which the respondent had to navigate carefully, as well as hostile spaceships the respondent had to decimate, were introduced in the higher difficulty sessions. The third and final segment emulated a machine room, in which respondents had to control the power based on randomly generated requests. In the increased difficulty settings, variables that could overheat the spaceship were introduced, demanding the respondent to multi-task and aiming to increase workload as a consequence.

2.2.2 Respondents

In total, 38 respondents have participated in the study. Demographic information of the respondents is set forth in Table 1. The respondents are students recruited from the University of Twente. Recruitment has been conducted by means of Sona, which is a cloud-based participant management system. A requirement for participation was that respondents didn't have any constraints that might interfere with the utilized sensors, such as for example a pacemaker. This was assessed by means of a short demographic questionnaire prior to the experiment. Additionally, respondents were made aware of informed consent prior to the experiment in order to ensure completely voluntary participation. Respondents were able to draw back from the experiment at any time.

Table 1

Demographic Overview Respondents

	Female	Male
10-19	7	0
20-29	13	4
30-39	1	1
40-49	0	8
50-59	0	0
60-69	0	4
Total	21	17

2.2.3 Devices and Sampling Rate

The Shimmer3 GSR+ sensor was used for both PPG and GSR measurements. This device was worn on the wrist. Signals were communicated wirelessly. An ear-clip was utilized for measuring PPG, and converting this to estimate heart rate. Skin conductivity, or GSR, was monitored by two electrodes attached to the fingers (Shimmer-research, n.d.). EEG measurement was conducted with the Muse 2, which is a multi-sensor headband that provides feedback on brain activity (Muse-incorperation, n.d.). The Shimmer3 GSR+ is able to read and output data signals on a sampling rate of 256 Hz, whereas the Muse 2 is able to sample at a maximum rate of 220 Hz.

As was set forth in the introduction, real-time classification requires a swiftly classifying network. A higher sampling rate equals more data traveling through the network, decelerating classification speed. As a consequence, it is highly beneficial to input data on the minimum required sampling rate with which key features can be detected consistently.

Fujita and Suzuki (2019) explored the required sampling rate for PPG feature detection. The extent to which important features were detected was contrasted for several sampling rates. A sampling rate of 60 Hz was found to be the absolute minimum required sampling rate for extracting all commonly utilized features in a stable manner (Fujita & Suzuki, 2019). Utilizing a slightly higher sampling rate is the safer option, however.

The Shimmer3 GSR+ manufacturers recommend a sampling rate of 100 Hz for PPG (Shimmer-research, n.d.). Hence, a sampling rate of 100 Hz was specified for the PPG modality.

The required sampling rate for the GSR modality is substantially lower as compared with both PPG and EEG measurement. In fact, the Shimmer3 GSR+ manufacturers recommend a sampling rate ranging between 0.03 and 5 Hz (Shimmer-research, n.d.). A sampling rate of 5 Hz was specified as a consequence.

For the EEG modality, different features require a widely different sampling rate in order to be detected. Frequency bands for traditionally considered EEG features reside on about 0.5-4 Hz for the delta feature, and at most on about 16-24 Hz for the beta feature. The gamma feature recently gained in popularity, and resides on a frequency band ranging up to 80 Hz (Weiergraeber, Papazoglou, Broich, & Mueller, 2016). In order to be able to detect an EEG feature residing on a 80 Hz frequency band, a substantially higher sampling rate is required to record the signal without aliasing. The required sampling rate can be determined by means of the Nyquist criterion for practical EEG sampling, defined as Equation 1,

$$f_{samp} > 2.5 * f_{max} \tag{1}$$

where f_{samp} reflects the required sampling rate and f_{max} reflects the frequency band around which the feature to be detected resides (Srinivasan, Tucker, & Murias, 1998). Hence, in order to be able to detect the gamma feature stably, a sampling rate of $2.5*80 = 200$ was specified for the EEG modality. A summary of the specified sampling rates per modality are depicted in Table 2.

2.2.4 Synchronization

Data streams stemming from the different modalities were required to be properly synchronized. This was accomplished by means of an application called Lab-Streaming Layer, hereafter referred to as ”LSL”. The data streams stemming from the different modalities were all streamed to LSL during the experiment. LSL properly synchronized these data streams, such that they are parallel. Subsequently, all data was recorded into a single file per participant (Kothe, Medine, & Grivich, 2018).

Table 2

Sampling Rate per Modality

	Specified sampling rate (Hz)
Electroencephalogram (EEG)	200
Galvanic Skin Response (GSR)	5
Photoplethysmography (PPG)	100

2.3 Framework Architecture

The upcoming section opens with the description of the architecture of the three single-modular networks. Subsequently, the multi-modular network architecture is elaborated upon. Lastly, several variations made on this multi-modular architecture are discussed.

2.3.1 Single-modular Network: Architectures

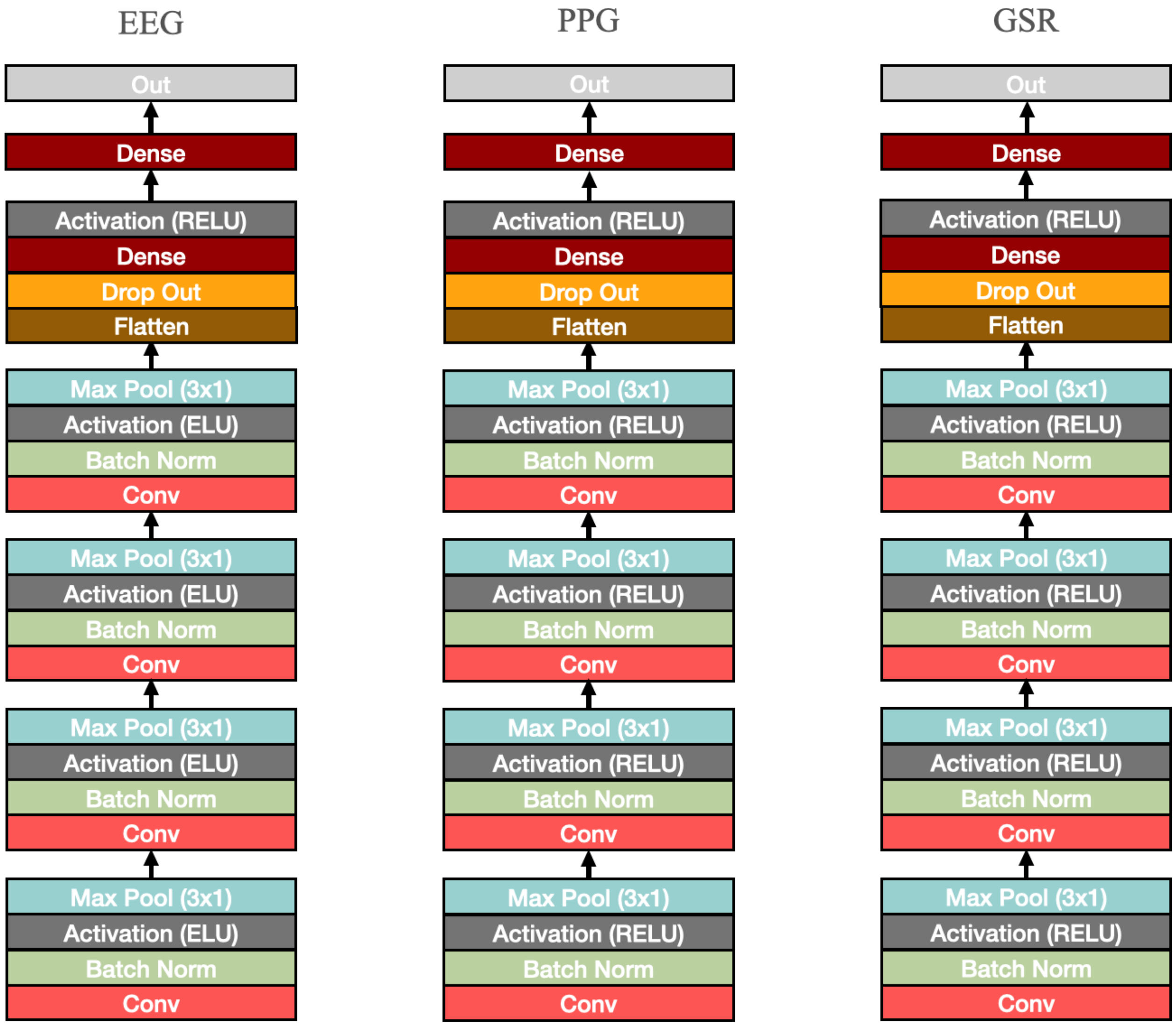
The appropriate architecture for each of the single-modular networks is determined by combining insights from the literature. The three single-modular network architectures are depicted as Figure 2.

The utilized network for the EEG modality was a ConvNet as proposed by Schirrmeister et al. (2017). The network was designed to include four convolutional blocks, each constituting a convolutional layer, followed by a batch normalization layer. The Exponential Linear Unit, hereafter referred to as ”ELU”, function was utilized as activation function. Each convolutional block was closed with a max pooling layer of stride three.

The utilized network for the GSR modality was a LSTM ConvNet hybrid, inspired from the work of Sun et al. (2019) and Dolmans et al. (2020). The network was designed to include two convolutional blocks, each constituting a convolutional layer, followed by a batch normalization layer, the activation layer and closed with a max-pooling layer of stride four. The Rectified Linear Unit, hereafter referred to as ”ReLU”, function was utilized as activation function. Following these two convolutional blocks were two LTSM layers.

Figure 2

The Three Single-modular Network Architectures



Lastly, the utilized network for the PPG modality was inspired from the network as proposed by Biswas et al. (2019). The network opens with two convolutional blocks, each consisting of a convolutional layer, a batch normalization layer, the activation layer and closed with a max pooling layer of stride four. The utilized activation function was the ReLU. Following these convolutional blocks were two LTSM layers.

2.3.2 Multi-modular Network: Architecture

The network architecture utilized for the multi-modular approach was determined by a combination of the single-modular networks, as derived from the literature. The previously delineated design principles (i.e. the principles of modularity and generalizability) were taken into account when doing so. A visual representation of the multi-modular network is depicted as Figure 3.

An intermediate fusion strategy is adopted due to its highly flexible nature as compared with other fusion strategies. The architecture for the single-modular parts of the network remained unchanged. In order to fuse the single-modular parts of the network, each of these is closed with one fully connected dense layer before feeding into the head network. This is done in order to flatten all inputs towards a lower dimensional space, such that concatenation was possible. The head network consists of four dense layers. These layers are alternated with a batch normalization and max-pooling layer with the objective of stabilization.

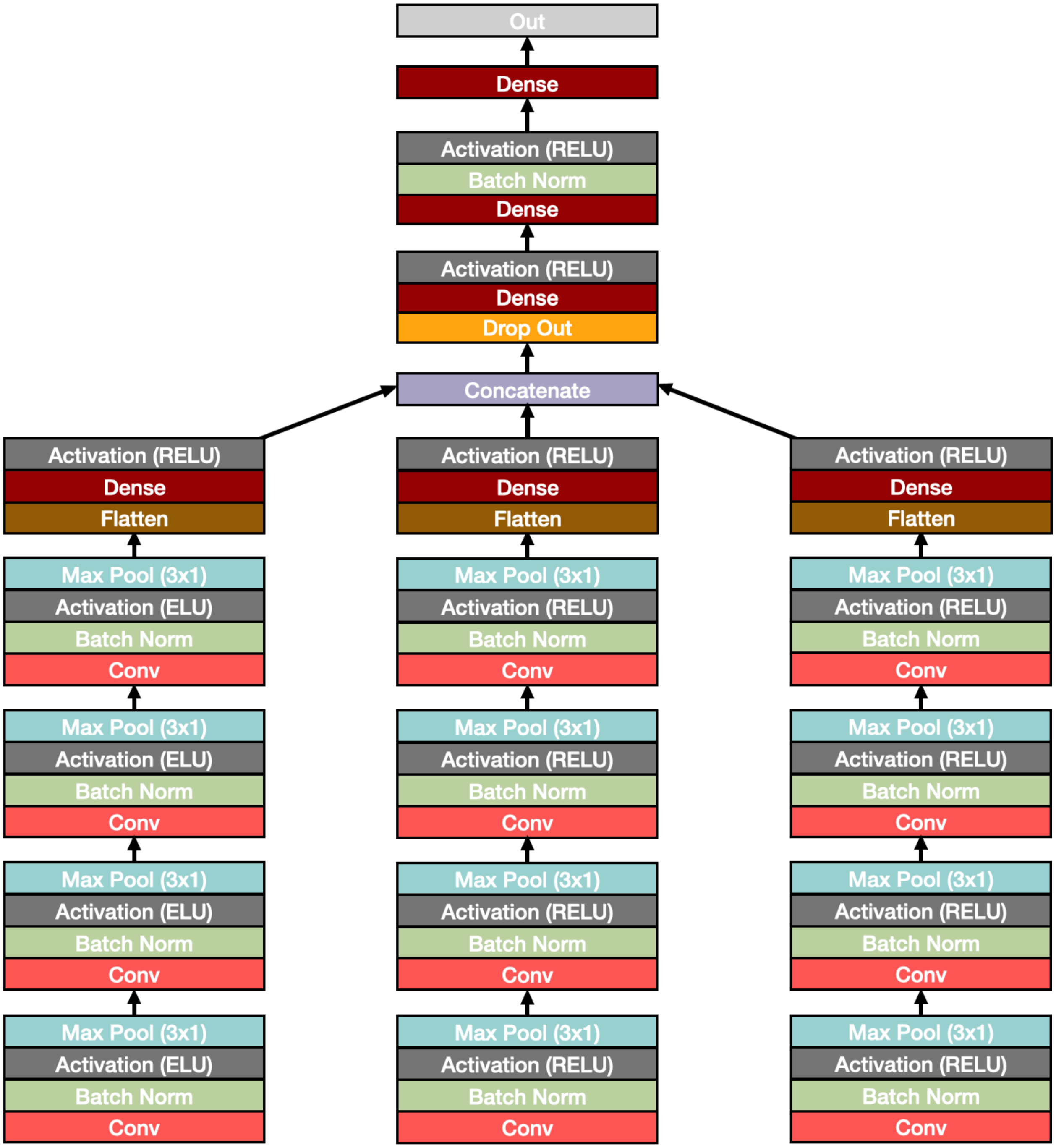
2.3.3 Multi-modular Network: Variations

Speed is a potential bottleneck for a multi-modular approach that is destined to classify in real-time. The previously delineated network is substantially complex in nature, hence posing the risk of not being able to classify fast enough. Therefore, several variations on this network architecture have been designed. These variations are not made by altering the network architecture, for deviating from the validated architecture could be detrimental with regards to performance. Hence, three variations with regards to size of the network as depicted in Figure 3 have been considered. The goal was to propose a network that is fast enough for real-time classification, whilst maintaining the highest amount of accuracy as possible.

Network size can be understood as the amount of specified filters for convolutional

Figure 3

The Multi-modular Network Architecture



layers, and the amount of specified neurons for all other utilized layers. A decrease in the specified amount of filters and neurons constitutes a decrease in network size, and consequently a decrease in the amount of required calculations. Doing so was expected to bring about an increase in speed. An overview of all three multi-modular network variations, and the amount of specified neurons/filters per layer, is provided in Table 3. Network 1 is referred to as the full network. The size of this network was determined by consulting the literature. The sizes of each of the sub-parts, i.e. the EEG, PPG and GSR sub-parts, were roughly adopted from the research that proposed these designs initially. The size of Network 2 constitutes of 75 % of the size of the full network. Lastly, Network 3 constitutes of 50 % of the size of the full network.

Table 3

Model Variations Based on Size

	EEG	GSR	PPG	Head
Network 1	Conv1: 25	Conv1: 128	Conv1: 128	Dense: 712
	Conv2: 50	Conv2: 128	Conv2: 128	Dense: 356
	Conv3: 100	LSTM1: 256	LSTM1: 256	Dense: 178
	Conv4: 200	LSTM1: 256	LSTM2: 256	
	Dense: 200	Dense: 256	Dense: 256	
Network 2	Conv1: 18	Conv1: 96	Conv1: 96	Dense: 534
	Conv2: 34	Conv2: 96	Conv2: 96	Dense: 267
	Conv3: 75	LSTM1: 192	LSTM1: 192	Dense: 134
	Conv4: 150	LSTM1: 192	LSTM1: 192	
	Dense: 150	Dense: 192	Dense: 192	
Network 3	Conv1: 13	Conv1: 64	Conv1: 64	Dense: 356
	Conv2: 25	Conv2: 64	Conv2: 64	Dense: 178
	Conv3: 50	LSTM1: 128	LSTM1: 128	Dense: 89
	Conv4: 100	LSTM1: 128	LSTM2: 128	
	Dense: 100	Dense: 128	Dense: 128	

Note: For all convolutional layers the depicted number reflects the amount of utilized filters, whereas for LTSM layers it reflects the amount of nodes.

2.4 Model Evaluation

The performance of the networks has been assessed and contrasted by means of several performance metrics. When assessing performance in deep/machine-learning, it is of importance to realize that each performance metric may favor one approach over the other, simply and solely due to the mathematical nature in which both the metric and the model are defined (Gunawardana & Shani, 2009). For this reason, we deemed it of importance to not merely evaluate performance based on one or two metrics. The adopted approach is rather to consider a conglomerate of different metrics, and evaluate performance based on the whole span of those. The utilized metrics constitute six well known and widely deployed metrics in the field of deep/machine-learning. All six metrics are constructed from the confusion matrix. The theoretical format of the confusion matrix is depicted as Figure 4.

Figure 4

Confusion Matrix

	True Positive	True Negative
Predicted Positive	a	b
Predicted Negative	c	d

The six metrics include accuracy, sensitivity, specificity, positive predicted value hereafter referred to as ”PPV, negative predicted value hereafter referred to as ”NPV” and F1. Table 4 depicts the mathematical composition of these performance metrics, by partly referring to confusion matrix depicted as Figure 4. In substantive terms, accuracy can simply be understood as the proportion of correct classifications. Sensitivity is understood as the proportion true positives, whereas specificity is understood as the proportion of true negatives. PPV reflects the proportion of predicted positives that are correctly predicted as positive. NPV on the other hand reflects the proportion of predicted negatives that are correctly predicted to be negative. Lastly, F1-score is a function of both sensitivity and PPV, seeking balance between the latter two.

For each of the three single-modular networks, as well as each of the three multi-modular network variants, these six metrics have been constructed and reported. The

Table 4

Utilized Performance Metrics

Accuracy:	$\frac{a+d}{a+b+c+d}$
Sensitivity:	$\frac{a}{a+c}$
Specificity:	$\frac{d}{b+d}$
Positive Predicted Value (PPV):	$\frac{a}{a+b}$
Negative Predicted Value (NPV):	$\frac{d}{c+d}$
F1-score:	$\frac{2*Sensitivity*PPV}{Sensitivity+PPV}$

network that performed best across the range of these metrics was considered to be the superior performing network. In case of contradictory interpretations, a reference was made back to the mathematical definition and substantive interpretation of the previously delineated performance metrics. To summarize, with the described assessment approach it was aimed to gain insight into multiple dimension of network performance, resulting in a broad and complete assessment of performance altogether.

3 Results

Figure 5

Mean Absolute Error (MAE) per modality

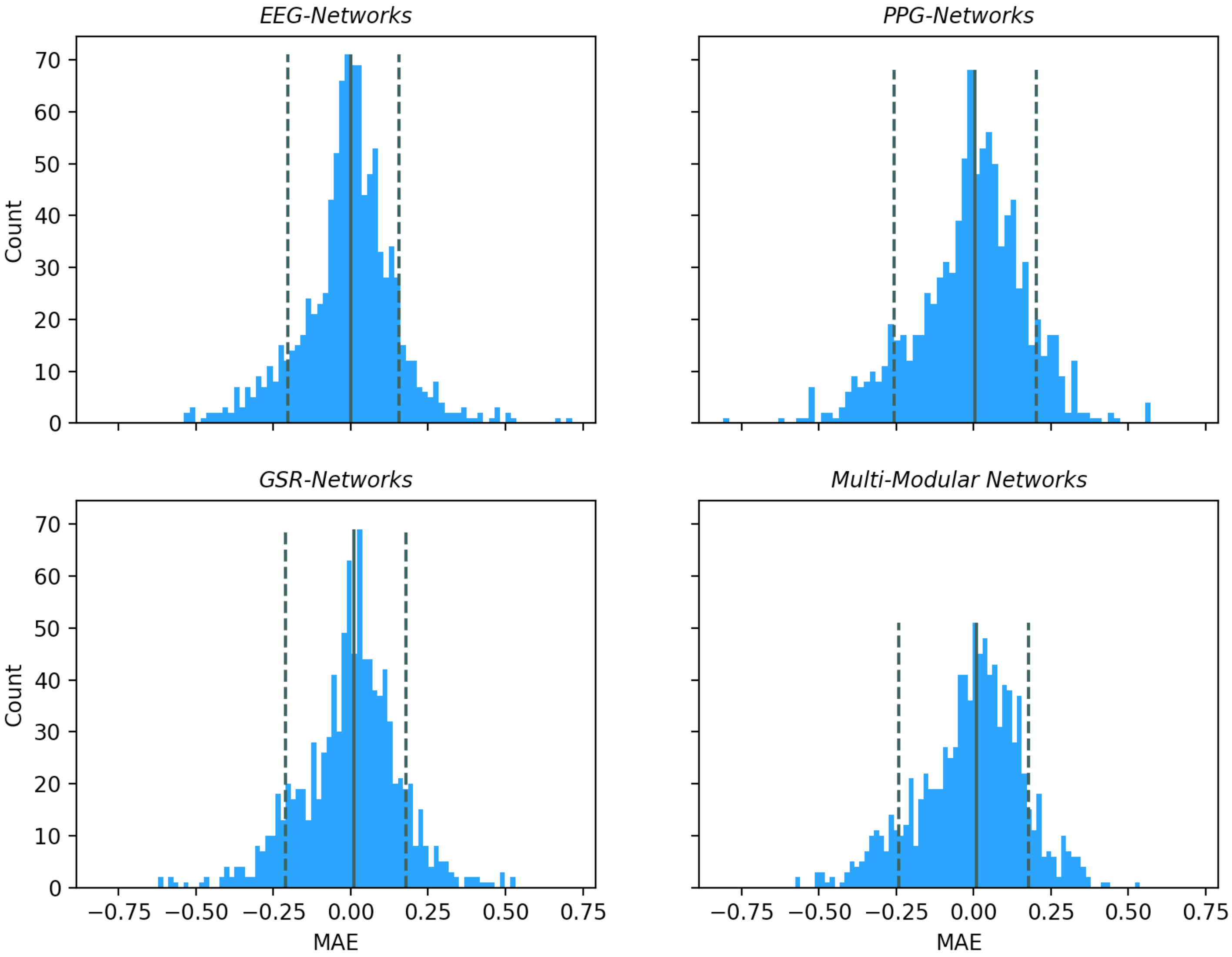
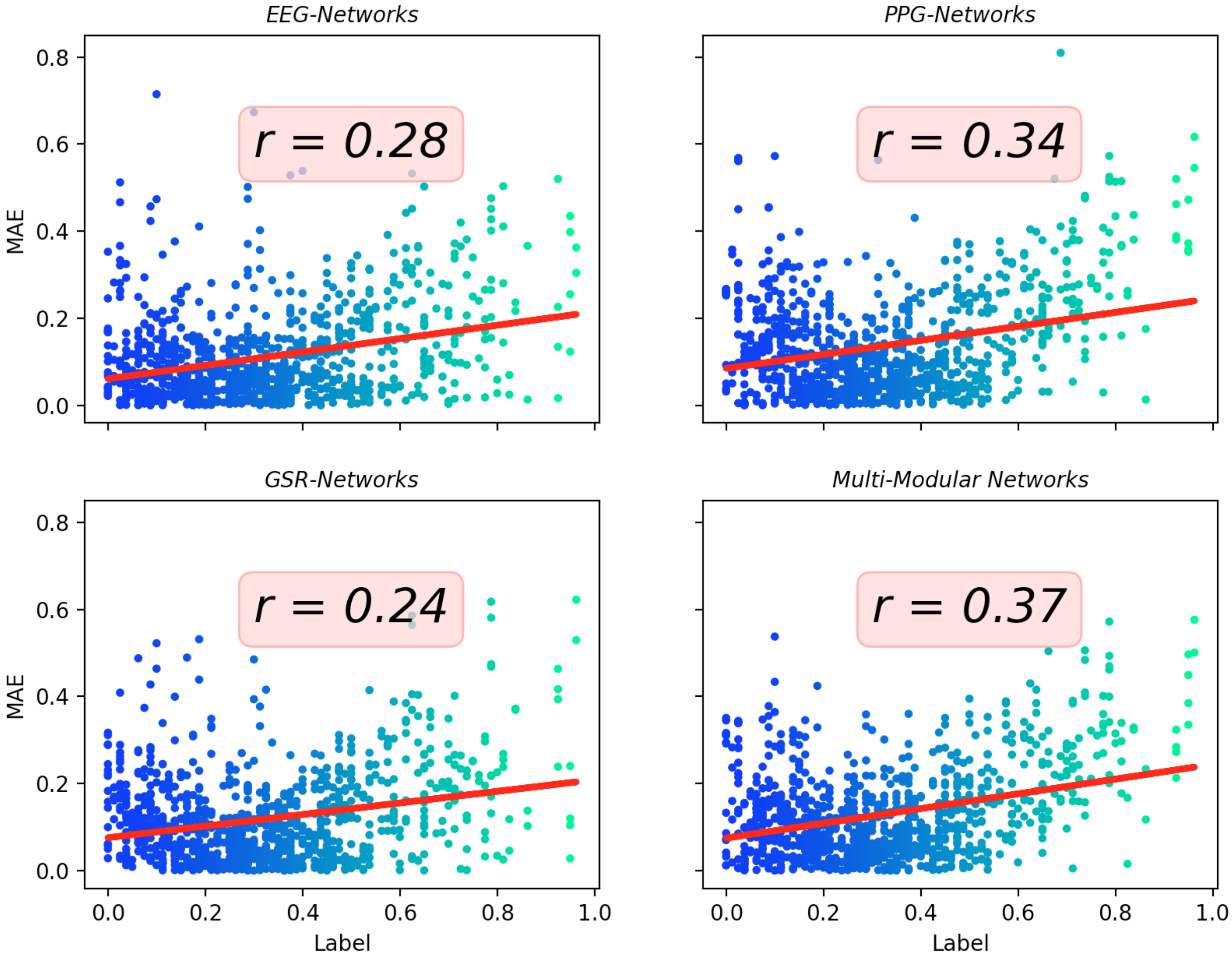


Figure 6

Mean Absolute Error (MAE) against label size



4 Limitations

- More HPO could have been done, to get even better performing models but we had limited computatioal power.

- This approach costs a lot of computational power to both train and do HPO. Especially since person-specific models are fitted. This upholds mostly for the multi-modal network.

- Due to the sheer size of the trained model and the fact that models are trained person specific, storage sizes of the models are huge. This could be unpractical. This upholds mostly for the multi-modal network.

- Possibly different window sizes per modality could lead to better results. * Insert some of the sources here that suggest different window sizes per modality* . This was beyond the scope of this project.

- How generizable is this to other situations for the same person (since things such as PPG are context dependent (like for example regarding how one ate).

- Long runtimes, even when testing. This upholds only for the multi network. The reason is likely the sheer size of these networks, which for some persons are blown up due to device lagging.

References

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th acm sigkdd international conference on knowledge discovery & data mining* (pp. 2623–2631).

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, *13*(1), 281–305.

Biswas, D., Everson, L., Liu, M., Panwar, M., Verhoef, B.-E., Patki, S., … others (2019). Cornet: Deep learning framework for ppg-based heart rate estimation and biometric identification in ambulant environment. *IEEE transactions on biomedical circuits and systems*, *13*(2), 282–291.

Craik, A., He, Y., & Contreras-Vidal, J. L. (2019). Deep learning for electroencephalogram (eeg) classification tasks: a review. *Journal of neural engineering*, *16*(3), 031001.

Daid, & Nallath. (2016). *Empty epsilon multiplayer spaceship bridge simulation.* URL: https://github.com/daid/EmptyEpsilon. GitHub.

de Waard, D., & te Groningen, R. (1996). *The measurement of drivers' mental workload.* Groningen University, Traffic Research Center Netherlands.

Dolmans, T., Poel, M., van 't Klooster, J.-W., & Veldkamp, B. (2020). Percieved mental workload detection using intermediate fusion multi-modal networks. manuscript submitted for publication.

Fujita, D., & Suzuki, A. (2019). Evaluation of the possible use of ppg waveform features measured at low sampling rate. *IEEE Access*, *7*, 58361–58367.

Gunawardana, A., & Shani, G. (2009). A survey of accuracy evaluation metrics of recommendation tasks. *Journal of Machine Learning Research*, *10*(12).

Han, S.-Y., Kwak, N.-S., Oh, T., & Lee, S.-W. (2020). Classification of pilots' mental states using a multimodal deep learning network. *Biocybernetics and Biomedical Engineering*, *40*(1), 324–336.

Hart, S. G. (2006). Nasa-task load index (nasa-tlx); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting* (Vol. 50, pp. 904–908).

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*.

Kothe, C., Medine, D., & Grivich, M. (2018). Lab streaming layer (2014). *URL:*

https://github. com/sccn/labstreaminglayer (visited on 02/01/2019).

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, *521*(7553), 436–444.

Mahtani, K., Spencer, E. A., Brassey, J., & Heneghan, C. (2018). Catalogue of bias: observer bias. *BMJ Evidence-Based Medicine*, *23*(1), 23.

Muse-incorperation. (n.d.). *Muse 2 brainwave activity headband.* Retrieved from: https://choosemuse.com/muse-2/.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., … Lerer, A. (2017). Automatic differentiation in pytorch.

Pretorius, A., & Cilliers, P. (2007). Development of a mental workload index: A systems approach. *Ergonomics*, *50*(9), 1503–1515.

Ramachandram, D., & Taylor, G. W. (2017). Deep multimodal learning: A survey on recent advances and trends. *IEEE Signal Processing Magazine*, *34*(6), 96–108.

Rastgoo, M. N., Nakisa, B., Maire, F., Rakotonirainy, A., & Chandran, V. (2019). Automatic driver stress level classification using multimodal deep learning. *Expert Systems with Applications*, *138*, 112793.

Schirrmeister, R. T., Springenberg, J. T., Fiederer, L. D. J., Glasstetter, M., Eggensperger, K., Tangermann, M., … Ball, T. (2017). Deep learning with convolutional neural networks for eeg decoding and visualization. *Human brain mapping*, *38*(11), 5391–5420.

Shi, Y., Ruiz, N., Taib, R., Choi, E., & Chen, F. (2007). Galvanic skin response (gsr) as an index of cognitive load. In *Chi'07 extended abstracts on human factors in computing systems* (pp. 2651–2656).

Shimmer-research. (n.d.). *Shimmer3 gsr unit.* Retrieved from: https://www.shimmersensing.com/products/shimmer3-wireless-gsr-sensor.

Shuggi, I. M., Oh, H., Shewokis, P. A., & Gentili, R. J. (2017). Mental workload and motor performance dynamics during practice of reaching movements under various levels of task difficulty. *Neuroscience*, *360*, 166–179.

Srinivasan, R., Tucker, D. M., & Murias, M. (1998). Estimating the spatial nyquist of the human eeg. *Behavior Research Methods, Instruments, & Computers*, *30*(1), 8–19.

Sun, X., Hong, T., Li, C., & Ren, F. (2019). Hybrid spatiotemporal models for sentiment classification via galvanic skin response. *Neurocomputing*, *358*, 385–400.

Tabar, Y. R., & Halici, U. (2016). A novel deep learning approach for classification of eeg motor imagery signals. *Journal of neural engineering*, *14*(1), 016003.

Weiergraeber, M., Papazoglou, A., Broich, K., & Mueller, R. (2016). Sampling rate, signal bandwidth and related pitfalls in eeg analysis. *Journal of neuroscience methods*, *268*, 53–55.

Young, M. S., Brookhuis, K. A., Wickens, C. D., & Hancock, P. A. (2015). State of science: mental workload in ergonomics. *Ergonomics*, *58*(1), 1–17.

Zhang, X., Lyu, Y., Hu, X., Hu, Z., Shi, Y., & Yin, H. (2018). Evaluating photoplethysmogram as a real-time cognitive load assessment during game playing. *International Journal of Human–Computer Interaction*, *34*(8), 695–706.