Pattern Recognition and Deep Learning: Group Project

1st Theo Bouwman
Graduate School of Natural Sciences
Utrecht University
Utrecht, The Netherlands
Student number: 8168067
t.c.bouwman@students.uu.nl

2nd Soheil Bagheri
Graduate School of Natural Sciences
Utrecht University
Utrecht, The Netherlands
Student number:6208908
s.bagheri@students.uu.nl

3rd Ivo Borkus
Graduate School of Natural Sciences
Utrecht University
Utrecht, The Netherlands
Student number: 6746713
i.borkus@students.uu.nl

4th S. Masoud Aghayan
Graduate School of Natural Sciences
Utrecht University
Utrecht, The Netherlands
Student number: 1138510
s.m.aghayan@students.uu.nl

5th Nicola Greco
Graduate School of Natural Sciences
Utrecht University
Utrecht, The Netherlands
Student number: 2327775
n.greco@students.uu.nl

***Abstract*—Magnetoencephalography (MEG) is a crucial tool in neuroscience, detecting brain wave abnormalities with high temporal resolution. Despite its spatially rich data from numerous sensors, MEG result interpretation requires expertise. Convolutional neural networks (CNNs) are introduced to enhance MEG data interpretability. CNNs, adept at spatial information in image tasks, are extended to accommodate MEG's spatial nature, providing a promising avenue for classification tasks.**

This project assesses two classification types: intra-subject and cross-subject. Intra-subject classifies subjects using data from the same set, while cross-subject trains on one set and tests on unseen subjects. The classification task categorizes subjects into rest, story, memory, or motor states. The goal is to explore the optimal CNN architecture for MEG data classification, addressing the complex interpretation challenges. The research aims to bridge the gap between MEG's potent data capture and the need for efficient, interpretable classification methods, potentially revolutionizing the understanding of brain function.

I. Introduction

Magnetoencephalography is a technique used in neuroscience research to detect abnormalities in brain wave activities [1]. MEG makes use of a highly temporal resolution as it captures data milliseconds apart. This allows for investigation of the highly dynamic neuronal activity in the brain. Furthermore, many sensors are used to create a higher spatial resolution of the brain activity. However, since this process captures so much data over a short period of time, interpreting results requires extensive experience from clinicians [2]. Interpretation of MEG data using deep learning approaches has been of great influence on the field. Researchers are exploring many types of architectures to determine the best possible model for classification tasks.

Convolutional neural networks are well equipped to capture spatial information in image classification tasks, which can be extended to the MEG data with its spatial nature. MEG data contains many sensors that act on different parts of the brain, the total combination of these sensors can represent a map of the brain. Therefore, the classification of tasks using MEG data can be accomplished by the use of CNNs.

During this project, we will evaluate 2 classification types, namely intra-subject and cross-subject classification. The intra-subject model is trained and evaluated on the same subjects. The cross-subject is trained on a set of subjects and evaluated on a set of unseen subjects. The classification task is to classify whether the observed subjects are in the rest (the subject is in a resting state), story (the subject performs a mental calculation and language processing task), memory (the subject is performing a memorization task), or motor state (the subject is performing motor tasks, like moving feet or fingers).

II. Related Work

In recent years, various studies have proposed deep learning architectures to handle MEG data, leveraging Convolutional Neural Networks (CNNs) to decipher the intricate neural signals for tasks ranging from clinical diagnostics to brain-computer interfaces. A notable example is a study by Ivan Zubarev et al. where CNNs were designed to conform to a generative model of brain signals, optimizing the interpretability of the neural sources influencing classification decisions [3]. The CNNs employ spatial and temporal filters to extract a compact representation of MEG signal features and use l1-regularized output layers for class sparsity. This approach is robust to inter-individual differences and suitable for pooling data from multiple subjects to apply to new ones.

Beyond the example provided, the literature abounds with various other instances where CNNs have been effectively employed for decoding MEG data [4], [5]. These studies collectively underscore the versatility of CNNs in neuroimaging,

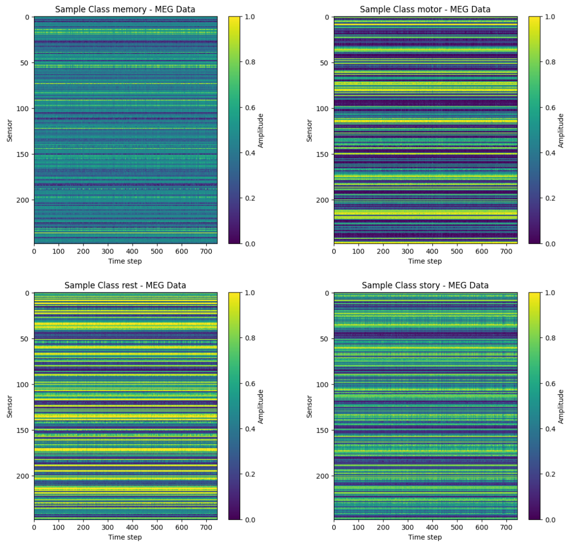

TABLE I
SAMPLES FOR EACH CLASS.

where their capacity to unravel complex, high-dimensional datasets is leveraged to advance our understanding of brain activity and its associated patterns.

The proven efficacy of CNNs in the field, along with the two-dimensional characteristics of MEG data, influenced our choice to utilize a CNN framework for our study.

III. APPROACH

We will explore the data used in this project, which models are tried, and which choices we made here.

A. Data

The data used in this project consists of two folders called Intra and Cross. Intra is used for the intra-subject classification and cross for the cross-subject classification. The training data for the intra-subject classification task has 8 samples per class, so 32 training samples. And it has 2 samples per class as a test set, so 8 test samples.

The training data for the cross-subject classification task has for each class 2 subjects with 8 samples per subject, totaling 64 training samples. As test data for the cross-subject classification task, there are for each class 2 subjects with 2 samples per subject, totaling 16 test samples.

Each sample has 248 magnetometer sensors, which are placed on the brain scalp, and 35624 time steps, resulting in a 248 x 35624 matrix. The recordings have a sample rate of 2034 Hz, therefor every sample corresponds to around 17.5 seconds. Due to computational limitations and the similarity between the time steps, we downsampled them during training and evaluation. From the 35624 time steps, 3 * 248 time steps were randomly selected. Resulting in a 248 x 744 matrix per sample. After downsampling, normalization was done using the MinMaxScaler class from the sci-kit-learn library, and all the sensor values were mapped to [0, 1].

B. Data Preprocessing

Data preprocessing is a critical step in ensuring that the input MEG data is suitable for analysis through deep learning models. The preprocessing pipeline implemented in this study encompasses two primary steps: downsampling and normalization.

Downsampling: The original MEG data consists of a time series with 35,624 time points per each of the 248 sensors. Given the high temporal resolution and the computational limitations, a downsampling procedure was necessary. The process involved randomly selecting 744 columns from the original 35,624 time points, effectively reducing the dimensionality of the data while retaining a representative subset of the time series. This method ensured that the essential temporal characteristics of the data were preserved.

Normalization: Normalization is an indispensable step in the preprocessing of data for neural network models. It helps in scaling the different magnitudes of sensor readings to a uniform range, which is critical for the convergence of the model during training. In this project, the MinMaxScaler from the scikit-learn library was employed. This scaler transformed each feature by scaling it to a given range, here set to [0, 1]. The normalization was applied to the training, validation, and test sets independently to prevent data leakage and ensure that the model could generalize well to unseen data.

The combination of downsampling and normalization formed the crux of the data preprocessing stage, which was instrumental in preparing the high-dimensional MEG data for the subsequent deep learning analysis.

C. Models

The CNN architecture designed for analyzing MEG data incorporates a multi-layered approach to effectively extract and classify complex features. It employs 4 convolutional layers with varying filter sizes and different numbers of channels to process the input data, applying the ReLU activation function for non-linear transformations. To reduce overfitting and enhance generalization, dropout layers with specific rates and batch normalization are integrated. The network also includes max pooling layers for spatial dimension reduction.

Subsequent layers of the network progressively increase the depth and complexity of feature extraction. This design aids in capturing intricate patterns within the data.

The final stages of the network involve flattening the 2D feature maps into a 1D vector, followed by two dense layers with ReLU activation. These layers are crucial for classification, culminating in a softmax output layer that categorizes the data into multiple classes.

Overall, the network's structure, combining convolutional, pooling, dropout, and dense layers, is tailored to effectively process and classify neuroimaging data, utilizing an optimization strategy suited for such complex datasets.

Importantly, this same neural network design was used for both the Intra-Subject and Cross-Subject datasets. This approach ensures a consistent method for analyzing the different types of data.

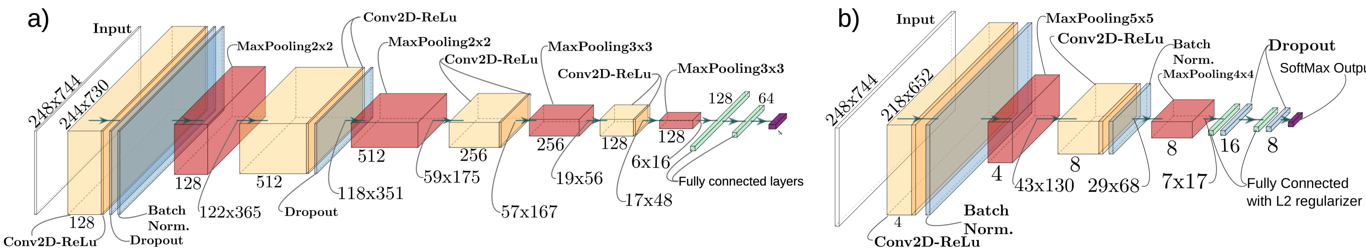

Fig. 1. Architecture of the Models. a) Complex Cross-subject trained model's architecture. b) Intra-subject simplified architecture

To further refine our model and address over-fitting, which was pinpointed by the big differences in train and test accuracy in the Cross-subject classification model, we simplified the architecture and trained new models. The new model, which will be referred to as 'simple' model architecture in contrast to the first 'complex' architecture, includes just two convolution layers, with larger filter sizes and lower number of channels, followed by max pooling and batch normalization layers, and introduced L2 normalization on both fully connected layers at the end of the network. The architecture of the model and its simplified version are visualized in Figure 5 and Figure 6

D. Hyper-parameters

Upon finalizing our model architecture, our primary focus shifted to the optimal selection of hyper-parameters, specifically targeting dropout rate, learning rate, and kernel sizes. The dropout rate, dictating the likelihood of neurons being deactivated during training, and the learning rate, crucial for the Adam optimization algorithm, were pivotal in this process. We experimented with varied kernel sizes across different layers to explore the impact of diverse hyper-parameter and filter size combinations on performance enhancement. For both the intra- and cross-subject models, our hyperparameter tuning encompassed learning rates of 0.001 and 0.0001 and dropout probabilities between 0.35 and 0.50. This approach aimed to ascertain the optimal balance for model generalization. Furthermore, we adopted a strategy of testing various kernel sizes, grouping the first two and the last two layers separately to streamline the process. The optimal combination for the intra-subject model was identified as a dropout rate of 0.35, kernel sizes of (3x6) for the first two layers and (3x9) for the third and fourth, and a learning rate of 0.0001, achieving a perfect validation accuracy of 1.000. The cross-subject model mirrored these parameters, with the exception of a (5x15) kernel size for the first layer and a dropout rate of 0.5, also reaching a validation accuracy of 1.000.

In the simplified architecture framework, we maintained our previous range for dropout and learning rates, while experimenting with larger kernel sizes (31x93 and 15x63) and varying L2 lambda values (0.001 and 0.01). In determining the most effective parameters for the Intra-subject model, the following configuration emerged as superior, delivering optimal validation set accuracy: a dropout rate set at 0.35, kernel sizes configured to (31, 93) for the first layer, and (15, 63) for the second, an L2 regularization lambda of 0.01, and a learning rate of 0.0001. Conversely, the Cross-subject model's optimal parameter set mirrored the Intra-subject model in terms of dropout rate, L2 lambda, and learning rate, but differed in kernel sizes, both set to (15, 63).

It is noteworthy that three out of the four models evaluated share the same hyper-parameters for L2 regularization, dropout rate, and learning rate. This uniformity across both complex and simple architectures, as well as Intra- and Cross-Subject models, indicates a methodological consistency in the hyper-parameter optimization strategy.

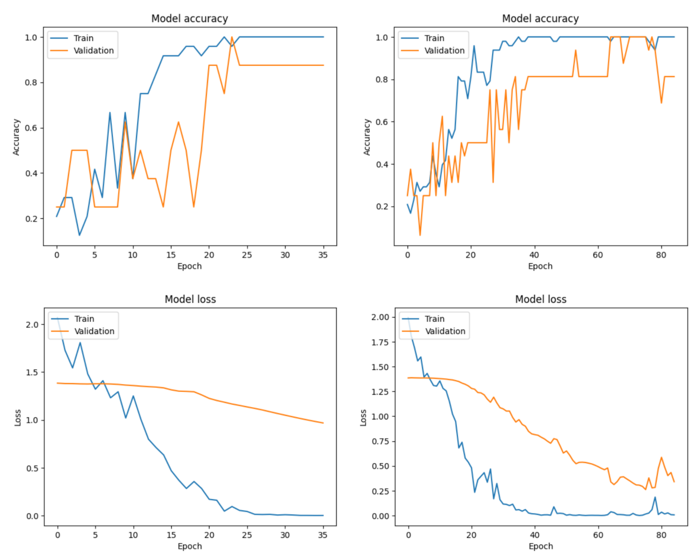

Fig. 2. Accuracy (top) and Loss (bottom) during training on intra (left) and cross (rigth) samples of the complex model

IV. Analysis of Training and Testing Accuracy Discrepancy

Potential Reasons for Discrepancy: The observed discrepancy between training and testing accuracies points to overfitting, where the model excels on training data but underperforms on unseen data. Possible causes include:

- **Model Complexity:** An overly complex model may memorize the training data, including noise, failing to generalize to new data.
- **Data Diversity:** Since the training data lacks variety, the model may not perform well on different testing data.

Model	Kernel Size 1	Kernel Size 2	Drop Out Rate	Learning Rate	L2 Lambda	Validation Accuracy	Test Accuracy
Cross - Complex	(5,15)	(3,9)	0.5	0.0001	na	1	0.4375 - 0.5625 - 0.75
Intra - Complex	(3,6)	(3,9)	0.35	0.0001	na	1	0.875
Cross - Simple	(15,63)	(15,63)	0.35	0.0001	0.01	1	0.50 - 0.25 - 0.3125
Intra - Simple	(31,93)	(15,63)	0.35	0.0001	0.01	1	1

TABLE II

SUMMARY OF THE BEST HYPERPARAMETERS AND THEIR CORRESPONDING ACCURACIES

Data Balance: Our analysis confirmed that the dataset is balanced, negating the need for techniques like SMOTE. The distribution of classes is as follows:

Class	Intra-Subject	Cross-Subject
0 (memory)	8	16
1 (motor)	8	16
2 (rest)	8	16
3 (story)	8	16

TABLE III

CLASS DISTRIBUTION IN THE DATASET

Proposed Solutions and Alternatives: To improve model generalization, we propose:

- **Regularization:** Applying L1/L2 regularization or dropout to limit the model's complexity.
- **Simplifying the Model:** Reducing the model's layers and neurons to prevent overfitting and focus on general patterns.
- **Ensemble Methods:** Using multiple models to average out biases and improve generalization.

Justification for Alternatives: The proposed strategies aim to mitigate overfitting while maintaining the model's ability to capture essential features. Ensemble methods provide an aggregated approach, combining the strengths of various models for enhanced performance and consistency across both training and testing datasets.

V. RESULTS

The evaluation of the initial, more complex CNN architecture yielded insightful results. The Intra-Subject model demonstrated a robust performance, achieving a test accuracy of 87.5%. This high level of accuracy is attributable to the model's training and testing on data from the same subject, allowing it to learn subject-specific features effectively. Conversely, the Cross-Subject model faced greater challenges, reflected in its lower accuracy rates across the three test sets: 43.75% for the first, 56.25% for the second, and 75.00% for the third. A notable observation from the confusion matrices is the difficulty both models encountered in correctly classifying class 0 (memory task), with the Cross-Subject model particularly struggling in this aspect.

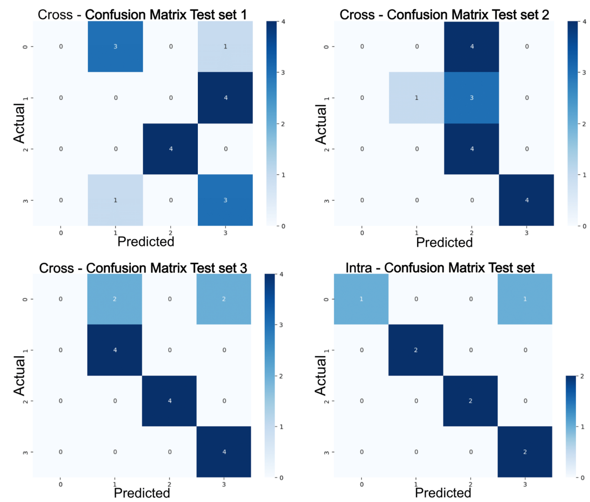

Fig. 3. Confusion Matrices for complex architecture

In the context of the simplified architecture, the Intra-Subject model excelled, remarkably achieving a 100% accuracy rate on its test set, indicating its proficiency in classifying all test samples correctly. However, the simplified architecture did not enhance the Cross-Subject model's performance, with accuracies of 50.00%, 25.00%, and 31.25% on the three test sets, respectively. This suggests that while the reduced complexity was beneficial for the Intra-Subject model, it might have been too simplistic for the Cross-Subject model, potentially leading to inability to generalize features from the training data to other subjects.

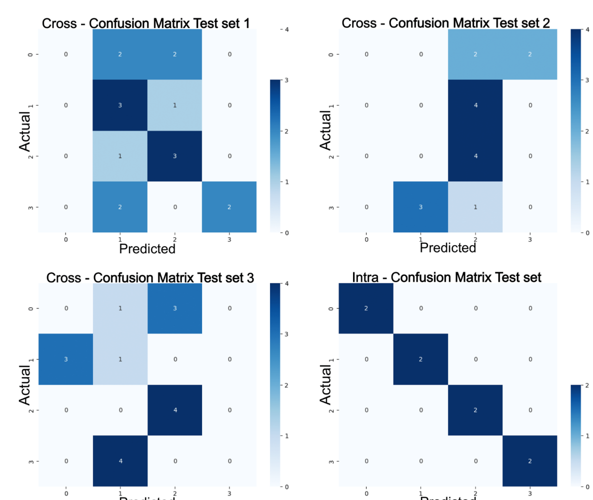

Fig. 4. Confusion Matrices for simplified architecture

VI. CONCLUSION

In this work, we investigated the use of deep learning architectures for processing Magnetoencephalography (MEG) data, with an emphasis on subject task classification. We investigate both cross- and intra-subject classification, which offer different training and generalization challenges. The investigation included data preprocessing, model architecture design, hyper-parameter tuning, and outcome evaluation.

Our exploration into hyper-parameter optimization revealed a delicate balance, especially concerning dropout rates, learning rates, and kernel sizes. The fine-tuning process resulted in optimal configurations that achieved remarkable validation accuracies for both Intra-Subject and Cross-Subject models. However, the observed discrepancy between training and testing accuracies, particularly in the Cross-Subject model, raised concerns about overfitting.

We experimented with various model configurations and introduced a simplified architecture to address this problem. The Cross-Subject model did not perform as well as the Intra-Subject model, which achieved perfect accuracy thanks to the simplified architecture. This disparity brought to light the difficulty in determining a model complexity that successfully strikes a balance between subject-to-subject generalization.

The significance of treating overfitting with regularization and model simplification was underlined by the examination of possible causes for the discrepancy in accuracy between training and testing. The proposed solutions, including ensemble methods and data augmentation, aim to enhance the model's ability to generalize while maintaining robustness.

In conclusion, this research contributes valuable insights into the complex task of classifying MEG data using deep learning architectures. The results highlight the necessity of carefully considering model complexity and hyper-parameter tuning in order to achieve the right balance between generalization and accuracy.

REFERENCES

[1] Singh, Sanjay P. "Magnetoencephalography: basic principles." Annals of Indian Academy of Neurology 17.Suppl 1 (2014): S107.

[2] Laohathai, Christopher, et al. "Practical fundamentals of clinical MEG interpretation in epilepsy." Frontiers in Neurology 12 (2021): 722986.

[3] Zubarev I, Zetter R, Halme HL, Parkkonen L. Adaptive neural network classifier for decoding MEG signals. Neuroimage. 2019 Aug 15;197:425-434. doi: 10.1016/j.neuroimage.2019.04.068. Epub 2019 May 4. PMID: 31059799; PMCID: PMC6609925.

[4] Aoe, J., Fukuma, R., Yanagisawa, T. et al. Automatic diagnosis of neurological diseases using MEG signals with a deep neural network. Sci Rep 9, 5057 (2019). https://doi.org/10.1038/s41598-019-41500-x

[5] Giovannetti, Antonio, et al. "Deep-MEG: spatiotemporal CNN features and multiband ensemble classification for predicting the early signs of Alzheimer's disease with magnetoencephalography." Neural Computing and Applications 33.21 (2021): 14651-14667.https://doi.org/10.1007/s00521-021-06105-4

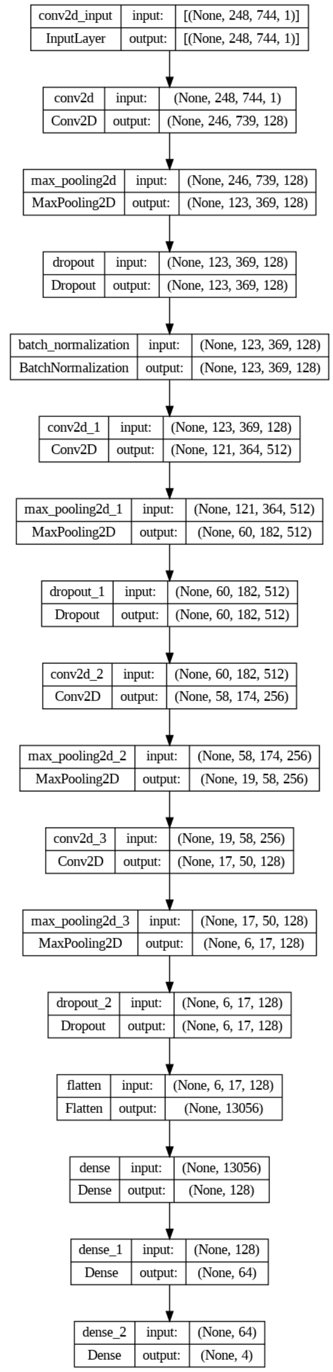

Fig. 5. Model Architecture with 7286724 trainable parameters

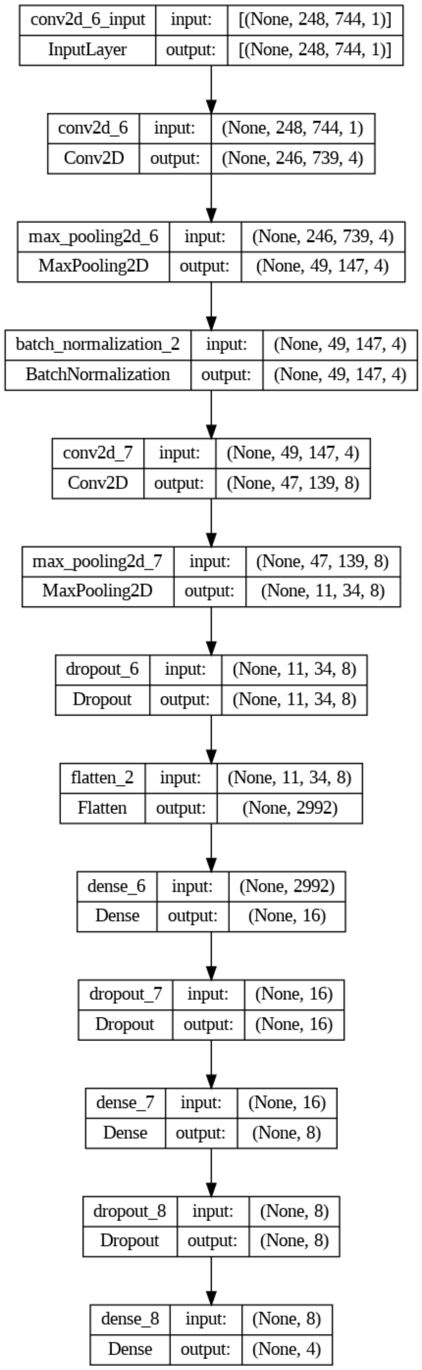

Fig. 6. Simplified Model Architecture with 49016 trainable parameters