Benchmarking Machine Learning and Deep Learning approach for Neuroimaging Data

Ekaterina Antipushina [1] **Ruslan Kalimullin** [1] **Marina Morozova** [1] **Mikhail Knyshenko** [1]

Abstract

One of the key challenges in modern computational neuroscience is the development of robust and reliable methods for analyzing and interpreting large-scale neural data. While machine and deep learning techniques have shown great promise in this area, their effectiveness can vary widely depending on the specific task or dataset being studied. In addition, there is often a lack of consensus among researchers regarding which methods are most appropriate for a given problem, making it difficult to compare results across different studies. To address these issues, a growing number of researchers are working to develop standardized benchmarks and best practices for applying machine learning and other computational techniques to neuroscience research. These efforts are aimed at improving the reproducibility and reliability of scientific findings, as well as facilitating collaboration and data sharing among researchers. Despite these challenges, the potential benefits of applying machine and deep learning to neuroscience research are enormous, including the development of more accurate and personalized diagnostic tools for neurological and psychiatric disorders.

Github repo: https://github.com/utoprey/CogniNet

1. Introduction

In recent years, the field of Functional Magnetic Resonance Imaging (fMRI) research has seen a growing interest in using machine learning and graph theory to analyze brain function. This interest is driven by the increasing availability of large-scale fMRI datasets, as well as the need for more sophisticated analytical methods to extract meaningful information from these datasets. GNNs, in particular, have gained popularity as a powerful tool for analyzing fMRI data due to their ability to model complex relationships between brain regions, while taking into account the underlying graph structure of the brain.

One key advantage of using GNNs for fMRI analysis is their ability to capture both local and global patterns of brain activity. Unlike traditional machine learning algorithms, which typically operate on a flat feature space, GNNs are designed to operate on graph-structured data. This makes them well-suited for analyzing fMRI data, which naturally lends itself to a graph representation, with brain regions serving as nodes and functional connections between them serving as edges.

In this article, we provide an overview of the key concepts underlying GNNs and also application of classic machine learning algorithms to fMRI data analysis. We review recent studies that have used GNNs to identify biomarkers of neurological and psychiatric disorders, predict cognitive performance, and uncover the functional organization of the brain. We also discuss the challenges and limitations of using GNNs for fMRI data analysis, including issues related to data preprocessing, model selection, and interpretability.

Overall, we argue that the integration of fMRI data with machine learning and graph theory has the potential to transform our understanding of the human brain, paving the way for new insights into brain function, disease, and cognition. In this article, we describe the use of machine learning methods and graph neural networks to analyze fMRI data designed by us to identify the most suitable pipeline which possibly can be used for identifying markers of cognitive diseases.

2. Related work

Several recent studies have focused on using machine learning and graph theory approaches to predict cognitive disorders from fMRI data. For example, one study used a graph-based neural network called BrainGB to predict Alzheimer's disease progression from fMRI data (Cui et al., 2022). The authors showed that BrainGB achieved state-of-the-art per-

[1]Skolkovo Institute of Science and Technology, Moscow, Russia. Correspondence to: Ekaterina Antipushina <Ekaterina.Antipushina@skoltech.ru>.

Final Projects of the Machine Learning 2023 Course, Skoltech, Moscow, Russian Federation, 2023.

formance on this task, outperforming several other machine learning algorithms.

Another study used a multi-modal approach to analyze fMRI and other clinical data for predicting Alzheimer's disease (Albright, 2019). The authors combined several types of data, including fMRI, structural MRI, and cerebrospinal fluid biomarkers, to build a machine learning model that could predict Alzheimer's disease with high accuracy. They found that the fMRI data contributed significantly to the accuracy of the model, suggesting that fMRI analysis may be a valuable tool for the early detection of Alzheimer's disease.

In a similar vein, a recent study proposed a novel preprocessing algorithm for fMRI data and used a neural network approach to predict Alzheimer's disease progression (Venugopalan et al., 2021). The authors found that their preprocessing algorithm improved the accuracy of the neural network model in predicting disease progression.

Overall, these studies highlight the potential of machine learning and graph neural networks approaches for predicting cognitive disorders from fMRI data. By combining fMRI data with other types of clinical and biological data, these approaches may enable earlier and more accurate diagnosis of these disorders, ultimately leading to better treatment and care for affected individuals.

```
BrainGAT(
  (convs): ModuleList(
    (0): Sequential(
      (0): GATConv(200, 32, heads=2)
      (1): Linear(in_features=64, out_features=32, bias=True)
      (2): LeakyReLU(negative_slope=0.2)
    )
    (1): Sequential(
      (0): GATConv(32, 32, heads=2)
      (1): Linear(in_features=64, out_features=32, bias=True)
      (2): LeakyReLU(negative_slope=0.2)
    )
    (2): Sequential(
      (0): GATConv(32, 32, heads=2)
      (1): Linear(in_features=64, out_features=64, bias=True)
      (2): LeakyReLU(negative_slope=0.2)
    )
  )
  (prepool): Sequential(
    (0): Linear(in_features=64, out_features=8, bias=True)
    (1): LeakyReLU(negative_slope=0.2)
    (2): BatchNorm1d(8, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
  )
  (fcn): BasicMLP(
    (net): Sequential(
      (0): Sequential(
        (0): Linear(in_features=1600, out_features=32, bias=True)
        (1): LeakyReLU(negative_slope=0.2)
        (2): Dropout(p=0.6, inplace=True)
      )
      (1): Sequential(
        (0): Linear(in_features=32, out_features=32, bias=True)
        (1): LeakyReLU(negative_slope=0.2)
        (2): Dropout(p=0.6, inplace=True)
      )
      (2): Sequential(
        (0): Linear(in_features=32, out_features=2, bias=True)
      )
    )
  )
)
```

Figure 1. The architecture of the Graph Neural Network model built by us with attention for the analysis of the connectivity graph obtained after processing the connectivity matrices obtained for fMRI datasets ABIDE and COBRE.

3. Algorithms and Models

The main goal of our study was to explore the application of deep learning methods in the analysis of neuroimaging data, with a focus on identifying which baseline methods could be most effective in this context.

3.1. Graph Attention Neural Network architecture

To this end, we constructed a graph neural network with three graph layers, each utilizing attention mechanisms to capture important features in the fMRI data. To enhance the performance of the neural network, we used LeakyReLU activation functions, which have been shown to improve training speed and accuracy in deep learning models (*Figure 1*).

To classify patients into those with mild cognitive disorders and those who are healthy, we used a fully connected neural network in the dense layer of the graph neural network. This allowed us to leverage the full power of deep learning to identify subtle patterns and differences in the fMRI data that may not be immediately apparent to the human eye.

3.2. Multilayer Perceptron architeture

As part of our analysis, we included a comparison between a Graph Neural Network and a Multilayer Perceptron.

For the Multilayer Perceptron (*Figure 2*), we used a three-layer architecture with several techniques to improve its performance. These included a dropout rate of 0.1, batch normalization, and a rectified linear unit (ReLU) activation function. The dropout technique is used to prevent overfitting of the model, while batch normalization helps to speed up the training process and improve the overall accuracy of the model. The ReLU activation function is commonly used in neural networks for its ability to overcome the vanishing gradient problem and improve the speed of convergence during training. By comparing the performance of the Multilayer Perceptron to that of the Graph Neural Network, we were able to determine the effectiveness of each model for our fMRI classification task.

3.3. Metrics for classification

To evaluate the performance of our models, we employed standard classification metrics, namely F-score, ROC, and accuracy. These metrics were selected as they are widely used in the field of machine learning and are appropriate for comparing the performance of both classical and deep learning methods in classification tasks. The F-score is a metric that balances precision and recall, making it useful in scenarios where there is an uneven distribution between the positive and negative classes.

```
MLP(
  (layers): Sequential(
    (0): Flatten(start_dim=1, end_dim=-1)
    (1): Linear(in_features=13456, out_features=1024, bias=True)
    (2): BatchNorm1d(1024, eps=1e-05, momentum=0.1, affine=True, track_running_stats=True)
    (3): ReLU()
    (4): Dropout(p=0.1, inplace=False)
    (5): Linear(in_features=1024, out_features=512, bias=True)
    (6): ReLU()
    (7): Dropout(p=0.1, inplace=False)
    (8): Linear(in_features=512, out_features=2, bias=True)
  )
)
```

Figure 2. The Multilayer Perceptron architecture we used to analyze fMRI data.

$$F = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{1}$$

The ROC-AUC is the true positive rate against the false positive rate and is useful in situations where the cost of false positives and false negatives are different.

Finally, accuracy is a common metric that measures the overall correctness of the classification results.

$$Accuracy = \frac{TP * TN}{TP + FP + TN + FN} \tag{2}$$

By using these metrics, we were able to effectively compare and assess the performance of our models on the given datasets.

4. Experiments and Results

Github repo: https://github.com/utoprey/CogniNet

To address the challenge of analyzing fMRI data, we developed a pipeline that incorporated both classical machine learning models and deep learning algorithms.

4.1. Datasets explanation

The ABIDE and COBRE datasets are both collections of functional magnetic resonance imaging (fMRI) data.

The ABIDE dataset, which stands for the Autism Brain Imaging Data Exchange, contains fMRI data from individuals with autism spectrum disorder (ASD) and typically developing individuals. The data includes resting-state fMRI scans as well as task-based fMRI scans, and was collected from 17 different sites across the United States. In current work, we work not with images, but with 163 connectivity matrices (shape for one matrix is 116x116) with 79 patients with schizophrenia and 73 healthy controls.

The COBRE dataset, which stands for the Center for Biomedical Research Excellence, contains fMRI data from individuals with schizophrenia and healthy controls. The data includes resting-state fMRI scans as well as task-based fMRI scans, and was collected from a single site in New Mexico. The utilized dataset contain 1035 connectivity matrices (shape for one matrix is 200x200) with 530 with Autism spectrum disorder (ASD) and 505 healthy controls.

Both datasets are widely used in the field of neuroscience to develop and test methods for analyzing fMRI data, particularly in the context of predicting and diagnosing various cognitive disorders.

A special files for train, validation and test splitting were provided with datasets (`abide_splits.json` and `cobre_splits_new.json` in GitHub repository).

Links for ABIDE `https://drive.google.com/drive/folders/1Xw-x9Zpfe-gEM0OLzDr35dlSzUVp2c2j` and COBRE `https://github.com/utoprey/CogniNet/tree/main/cobre` datasets.

4.2. fMRI data prepossessing

Preprocessing for Classical Machine Learning approach:

1. Matrix concatenation: transformations of all data in the dataset into one feature matrix. 163 matrices of dimension 116x116, in the case of the COBRA dataset are converted to a matrix of 163x6700 and 1035 matrices of dimension 200x200 in the case of the ABIDE dataset are converted to a matrix of 1035x19902.

2. Adding targets for classification: to train the classifier with a target, we had diagnoses made by the test subjects. For the COBRE dataset: schizophrenia struct = 1, no known disorder = 0. For the ABIDE dataset: autism = 1, healthy = 0.

Preprocessing for Deep Learning approach:

1. Graph construction: The connectivity between ROIs is represented by an adjacency matrix, which is used to construct a graph. There are different ways to define the adjacency matrix, including using correlation, partial correlation, or mutual information measures.

2. Graph data structure: Finally, the fMRI data is organized into a graph data structure, which can be fed into a GNN for further analysis and modeling.

The Pytorch Geometric library provides tools to support this preprocessing pipeline, including data loaders for loading fMRI data, graph construction functions, and GNN models that can be trained on the constructed graphs.

4.3. Classical Machine Learning

Our pipeline consisted of a set of popular machine learning methods, including Support Vector Machines (SVM), Logistic Regression (LogReg), Random Forest, XGBoost, and SVM with Principal Component Analysis (PCA).

Table 1. The best hyperparameters of ML algorithms for ABIDE dataset

MODELS	HYPERPARAMETERS
LOGREG	CLASS WEIGHT: NONE, PENALTY: L2, C: 10
RANDOM FOREST	NUMBER OF ESTIMATORS: 100, MAX DEPTH: 3
SVM	C: 10, KERNEL: RBF
SVM+ PCA	**C: 0.01, KERNEL: LINEAR, CLASS WEIGHT: BALANCED, PROBABILITY: TRUE, COMPONENTS: 500**
XGBOOST	MAX DEPTH: 10, SCALE POSITIVE WEIGHTS: 1.0, α: 0

Table 2. ABIDE Classic Machine Learning methods results

MODELS	F1 SCORE	ACCURACY	ROC-AUC
LOGREG	66± 1.3	66± 1.3	7 66± 1.3
RANDOM FOREST	65± 3.7	67± 3.0	67± 3.1
SVM	67± 1.2	67± 1.3	67± 1.3
SVM + PCA	**69± 1.7**	**68± 1.5**	**68± 1.6**
XGBOOST	65± 0.7	63± 1.0	63± 1.1

To evaluate the performance of each method, we used a range of metrics, including F-score, Receiver Operating Characteristic (ROC) curve analysis, and Accuracy. By leveraging a combination of these metrics, we were able to assess the performance of each method for fMRI data, such as classification task.

When we found the best hyperparameters (*Table 1*) of chosen models and applied our pipeline to the dataset, we found that SVM+PCA achieved the best metrics among all the listed methods (*Table 2*). Specifically, SVM+PCA obtained high scores across all the metrics, indicating that the use of PCA can enhance the performance of SVM in analyzing fMRI data.

However, when we trained our models on the COBRE dataset, we found that SVM (*Table 3*) outperformed all other methods (*Table 4*). These results suggest that SVM is a strong candidate for fMRI analysis when studying neurological disorders. Additionally, these findings demonstrate that the choice of machine learning algorithm may depend on the specific dataset and the task at hand.

Table 3. The best hyperparameters of ML algorithms for COBRE dataset

MODELS	HYPERPARAMETERS
LOGREG	CLASS WEIGHT: NONE, PENALTY: L2, C: 0.1, SOLVER: LIBLINEAR
RANDOM FOREST	NUMBER OF ESTIMATORS: 50, MAX DEPTH: 3
SVM	**C: 0.01, KERNEL: LINEAR, PROBABILITY: TRUE**
SVM+ PCA	C: 10, KERNEL: LINEAR, CLASS WEIGHT: BALANCED PROBABILITY: TRUE, COMPONENTS: 70
XGBOOST	MAX DEPTH: 10, η: 0.1, SCALE POSITIVE WEIGHTS: 1.0, α: 0, OBJECTIVE: 'BINARY:LOGISTIC'

Table 4. COBRE Classic Machine Learning methods results

MODELS	F1 SCORE	ACCURACY	ROC-AUC
LOGREG	73± 5.6	76± 4.5	76± 4.9
RANDOM FOREST	65± 7.3	69± 6.4	69± 6.5
SVM	**76± 5.8**	**78± 3.9**	**79± 4.7**
SVM + PCA	71± 8.7	76± 6.2	75± 7.2
XGBOOST	67± 4.4	72± 2.9	71± 3.3

Overall, our pipeline provides a comprehensive platform for evaluating the performance of various machine learning methods for fMRI analysis. Using a combination of classical machine learning methods, as well as various evaluation indicators, our pipeline can help researchers choose the most appropriate method for their specific fMRI analysis task.

4.4. Deep Learning (Graph Neural Network)

In addition to classical machine learning methods, we also explored the use of deep learning algorithms for fMRI analysis. Specifically, we implemented graph neural networks, which have been shown to be effective in processing fMRI data.

To implement the graph neural networks, we selected a graph neural layer with attention. We compared the performance of the graph neural networks to a baseline 3-layer MultiLayer Perceptron (MLP).

Table 5. ABIDE Neural Network results

MODESL	F1 SCORE	ACCURACY	ROC-AUC
BGBGAT	67± 1.5	65± 1.5	68± 1.7
MLP	**71± 1.2**	**67± 1.7**	**71± 1.5**

Table 6. COBRE Neural Network results

MODESL	F1 SCORE	ACCURACY	ROC-AUC
BGBGAT	79± 4.5	67± 3.6	72± 4.2
MLP	**79± 1.8**	**78± 2.4**	**79± 3.8**

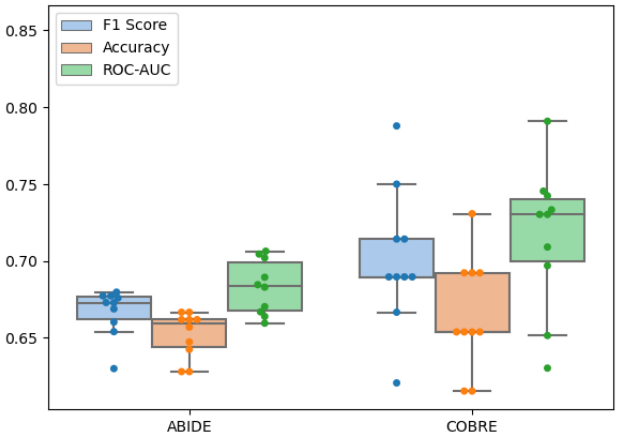

Figure 3. GATConv performance boxplot, for 60 epochs, 10 reinitializations best metric value during training

However, our results showed that the graph networks were too powerful for the small biomedical datasets (*Figure 3*) used in our study, such as ABIDE (*Table 5*) and COBRE (*Table 6*). This can be seen in the neural network accuracy metrics, which were significantly higher than those achieved by the MLP. However, when we compared the metrics of the MLP to the graph neural networks, we found that the MLP was not inferior to the graph methods (*Figure 4*).

These findings suggest that graph neural networks may be more suitable for larger datasets or for more complex fMRI analysis tasks. Additionally, the use of a baseline MLP can still provide a reasonable level of performance in fMRI analysis, particularly when working with small datasets.

Overall, our study highlights the potential of both classical machine learning and deep learning methods for fMRI analysis. By carefully selecting the appropriate method and evaluating its performance using a range of metrics, researchers can gain valuable insights into the functional organization of the brain and improve our understanding of cognitive disorders.

5. Conclusion

In conclusion, our study aimed to compare the performance of classical machine learning methods and deep learning algorithms in analyzing neuroimaging data. Our findings show that SVM with PCA dimension reduction performs

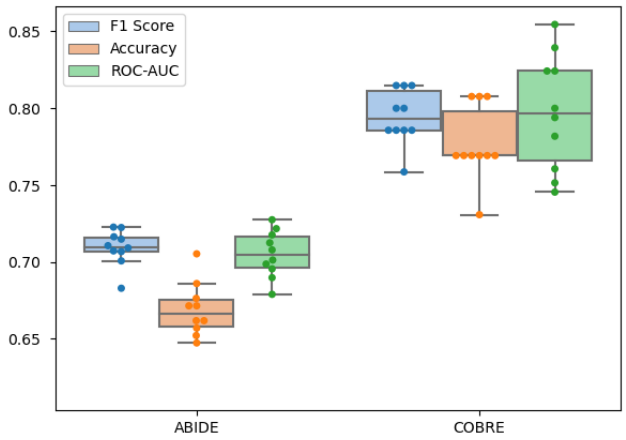

Figure 4. MLP performance boxplot, for 60 epochs, 10 reinitializations best metric value during training

the best among classical machine learning models, while graph neural networks, which are commonly seen as the optimal method for fMRI analysis, did not show significant improvements compared to the Multilayer Perceptron architecture in our experiments. Interestingly, we also found that graph neural networks are too powerful for small biomedical datasets such as ABIDE and COBRE. Our results suggest that careful consideration should be given when selecting a baseline model for neuroimaging data analysis and that the optimal method might depend on the specific dataset characteristics.

References

Albright, J. Forecasting the progression of alzheimer's disease using neural networks and a novel preprocessing algorithm. *Alzheimer's Dementia: Translational Research Clinical Interventions*, 5:483–491, 2019. ISSN 2352-8737. doi: https://doi.org/10.1016/j.trci.2019.07.001.

Cui, H., Dai, W., Zhu, Y., Li, X., He, L., and Yang, C. Interpretable graph neural networks for connectome-based brain disorder analysis. June 2022. doi: 10.48550/ARXIV.2207.00813.

Venugopalan, J., Tong, L., Hassanzadeh, H. R., and Wang, M. D. Multimodal deep learning models for early detection of alzheimer's disease stage. *Scientific Reports*, 11(1):3254, Feb 2021. ISSN 2045-2322. doi: 10.1038/s41598-020-74399-w.

A. Team member's contributions

Explicitly stated contributions of each team member to the final project.

Ekaterina Antipushina (25% of work)

- Write a code for the implementation of classical ML approaches on 2 datasets
- Preparing Sections 4 of this report
- Preparing the GitHub Repo

Marina Morozova (25% of work)

- Write code for implementing GAT, and MLP for 2 datasets
- Preparing Sections 4 of this report
- Preparing the GitHub Repo

Ruslan Kalimullin (25% of work)

- Preparing Sections 2 and 3 of this report
- Write code for implementing GAT
- Preparing the GitHub Repo

Mikhail Knyshenko (25% of work)

- Related scientific articles review
- Preparing Sections 1 and 5 of this report
- Review of this report
- fMRI data preprocessing
- Preparing the GitHub Repo

B. Reproducibility checklist

Answer the questions of following reproducibility checklist. If necessary, you may leave a comment.

1. A ready code was used in this project, e.g. for replication project the code from the corresponding paper was used.

   ☐ Yes.
   ☑ No.
   ☐ Not applicable.

   Students' comment: None

2. A clear description of the mathematical setting, algorithm, and/or model is included in the report.

   ☑ Yes.
   ☐ No.
   ☐ Not applicable.

   Students' comment: None

3. A link to a downloadable source code, with specification of all dependencies, including external libraries is included in the report.

   ☑ Yes.
   ☐ No.
   ☐ Not applicable.

   Students' comment: None

4. A complete description of the data collection process, including sample size, is included in the report.

   ☑ Yes.
   ☐ No.
   ☐ Not applicable.

   Students' comment: None

5. A link to a downloadable version of the dataset or simulation environment is included in the report.

   ☑ Yes.
   ☐ No.
   ☐ Not applicable.

   Students' comment: None

6. An explanation of any data that were excluded, description of any pre-processing step are included in the report.

   ☑ Yes.
   ☐ No.
   ☐ Not applicable.

   Students' comment: None

7. An explanation of how samples were allocated for training, validation and testing is included in the report.

   ☑ Yes.
   ☐ No.
   ☐ Not applicable.

   Students' comment: None

8. The range of hyper-parameters considered, method to select the best hyper-parameter configuration, and specification of all hyper-parameters used to generate results are included in the report.

   ☐ Yes.
   ☑ No.
   ☐ Not applicable.

   Students' comment: This information may be found in .ipynb notebooks (CogniNet/ML part/)

9. The exact number of evaluation runs is included.

   ☑ Yes.
   ☐ No.
   ☐ Not applicable.

   Students' comment: None

10. A description of how experiments have been conducted is included.

    ☑ Yes.
    ☐ No.
    ☐ Not applicable.

    Students' comment: None

11. A clear definition of the specific measure or statistics used to report results is included in the report.

    ☐ Yes.
    ☑ No.
    ☐ Not applicable.

    Students' comment: We use just classical metrics of classification model evaluation.

12. Clearly defined error bars are included in the report.

    ☐ Yes.
    ☑ No.
    ☐ Not applicable.

    Students' comment: None

13. A description of the computing infrastructure used is included in the report.

- ☐ Yes.
- ☑ No.
- ☐ Not applicable.

Students' comment: None