Deep Learning for ADHD Detection in fMRI: Comparing CNN and CNN-LSTM Networks with and without Transfer Learning from a CNN-Based Autoencoder

Yasty Sánchez, Fabián Navarro, Carlos Ramírez, Ulises Fonseca
School of Computer Science and Informatics
University of Costa Rica, San José, Costa Rica
yasty.sanchez@ucr.ac.cr, fabian.navarroparraga@ucr.ac.cr, carlos.ramirezmasis@ucr.ac.cr, jose.fonsecahurtado@ucr.ac.cr

***Abstract*—This study investigates the use of convolutional neural networks (CNNs) and CNN-LSTM models for the diagnosis of Attention Deficit Hyperactivity Disorder (ADHD) using fMRI data from the ADHD-200 dataset. We explore the performance of CNN-based models, including the integration of pretrained weights from a CNN autoencoder (CNN-AE) and the addition of LSTM layers to capture temporal dependencies in the data. Our experiments show promising results in distinguishing ADHD participants from typically developing controls (TDC). We found that while pretrained CNN-AE weights consistently improved classification accuracy across both architectures, the CNN with AE achieved the most balanced performance with respect to sensitivity and specificity. Interestingly, the CNN-LSTM with AE yielded the highest accuracy and sensitivity, indicating its potential for effectively identifying ADHD cases. These findings highlight the potential of deep learning, particularly CNNs and CNN-LSTMs with pretrained components, for neuroimaging-based ADHD diagnosis.**

***Index Terms*—ADHD diagnosis, fMRI, convolutional neural networks (CNN), CNN-LSTM, pretrained weights, CNN autoencoder (CNN-AE), deep learning, neuroimaging, temporal dependencies, ADHD-200 dataset.**

I. Introduction

Attention Deficit Hyperactivity Disorder (ADHD) is a common neurodevelopmental disorder, affecting an estimated 5% to 7.2% of children and 2.5% to 6.7% of adults globally [1]. Traditionally, ADHD diagnosis has relied on behavioral assessments, which are often subjective and may lead to variability in outcomes [10]. However, with the advancement of neuroimaging techniques, particularly functional Magnetic Resonance Imaging (fMRI), researchers now have access to detailed insights into brain activity. This has opened the door to identifying potential biological markers that could complement behavioral diagnoses [9]. In recent years, machine learning techniques have shown great potential in analyzing fMRI data to uncover complex patterns related to ADHD, offering a more objective approach to diagnosis [20].

Recent studies demonstrate that AI-based models are effective in analyzing fMRI data for distinguishing ADHD from healthy controls, achieving accuracies between 70% and 85% [9]. Shoeibi et al. [19] explored a convolutional autoencoder (CNN-AE) for feature extraction, reaching 72.7% accuracy. Other studies using convolutional convolutional networks (CNNs) without autoencoders include Zhang et al.'s [22] separated channel attention CNN, which achieved 68.9% accuracy, and Riaz et al.'s [17] DeepFMRI model, which reached 73.1%. Incorporating LSTM (Long-Term Short-Term Memory) networks has shown improvements. Dey et al. [8] combined a capsule network with LSTM, achieving 77% accuracy, while Khullar et al.'s [14] hybrid 2D CNN-LSTM reached 96% accuracy on the ADHD-200 dataset.

While autoencoders show promise for ADHD detection [19], their integration with CNN-LSTM models remains under-researched. This study aims to address this gap by comparing the performance of CNN and hybrid CNN-LSTM models, both with and without pretrained weights from a convolutional autoencoder (CNN-AE), using the ADHD-200 dataset. We evaluate accuracy, sensitivity, and specificity to identify the most effective approach for detecting ADHD from fMRI images without focusing on specific subtypes. This broader perspective aims to deepen the understanding of ADHD detection and offer valuable insights for professionals seeking to enhance diagnostic accuracy.

Our findings demonstrate the significant impact of integrating pretrained CNN-AE weights within both CNN and CNN-LSTM architectures for ADHD detection. Notably, the CNN-LSTM model incorporating pretrained AE weights achieved the highest performance, with an accuracy of 76.73% and a sensitivity of 0.85. This highlights the potential of combining the spatial feature extraction capabilities of CNNs, the temporal modeling of LSTMs, and the dimensionality reduction of autoencoders for effectively identifying ADHD from fMRI data.

This paper is organized as follows: The Background and Related Work section reviews relevant literature, setting the stage for the research. The Methodology section covers essential processes such as data preprocessing, feature extraction, and model classification. In this section, we detail the design of our 2x2 experiment, including the architecture of the CNN-AE, CNN, and CNN-LSTM models, while highlighting con-

stant factors such as the dataset and preprocessing techniques employed. The Results section compares the performance of standard CNNs with hybrid CNN-LSTM networks and summarizes findings. Finally, we present our conclusions and suggest future research directions in the Conclusions section.

II. Background and Related Work

The following subsections explore various state-of-the-art approaches and resources for detecting ADHD using fMRI data analysis. We start with the ADHD-200 Preprocessed Dataset, an essential resource for studying the neural and behavioral aspects of ADHD. Next, we review CNN algorithms and CNN-LSTM algorithms, which have proven effective in classifying ADHD from fMRI data. Finally, we discuss CNN-based autoencoders, which are useful for reconstructing and identifying patterns in fMRI data, as they capture important features for ADHD diagnosis.

A. ADHD Detection Using fMRI Data Analysis

ADHD is a neurodevelopmental disorder characterized by inattention, hyperactivity, and impulsivity, often linked to abnormalities in brain regions like the prefrontal cortex and basal ganglia [4]. fMRI allows researchers to study brain activity by tracking blood flow changes, providing insights into which areas are active during tasks. fMRI data has revealed altered activation and connectivity in individuals with ADHD in regions associated with attention and impulse control [4]. However, using fMRI for diagnosis is challenging due to the heterogeneity of ADHD [18]. Machine learning offers a promising solution by analyzing complex neuroimaging data to detect subtle patterns and abnormalities. Recent research demonstrates the potential of AI-based models to effectively analyze fMRI data and distinguish ADHD from healthy controls with over 80% accuracy [18].

B. The ADHD-200 Preprocessed Dataset

The 1000 Functional Connectomes Project (FCP) in 2010 aggregated fMRI data globally, highlighting its utility for hypothesis-driven research [2]. Expanding on this, the ADHD-200 Consortium released a large fMRI dataset to aid in ADHD diagnosis and classification research, comprising structural and functional MRI data from 973 participants (491 with ADHD, 582 typically developing controls) across eight research sites [2]. To support machine learning researchers, the ADHD-200 Preprocessed dataset—available on NITRC—was created, offering 4D fMRI volumes with anatomical and functional data, time-series extractions, and phenotypic details such as ADHD subtype, IQ, and medication status for participants aged 7 to 21 [4]. Table I shows a summary of the dataset demographics from NYU, KKI and Peking.

Site	*Age*	*Female*	*Male*	*Control*	*ADHD*	*Total*
NYU	7-18	76	140	98	118	216
KKI	8-13	37	46	61	22	83
PU	8-17	52	142	116	78	194
PU_1	8-17	36	49	61	24	85

Table I: Summary of ADHD-200 Dataset Demographics from Four Key Sites

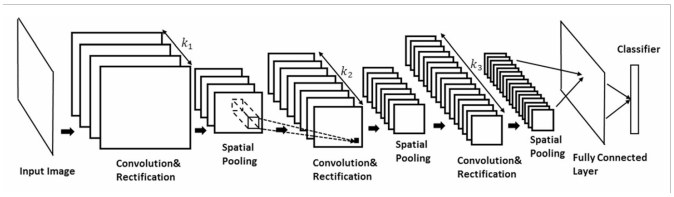

Figure 1: The architecture of a 2D convolutional neural network (CNN) [14]

C. CNN Algorithms for ADHD Detection

Convolutional neural networks (CNNs) are deep learning algorithms widely used for image and video recognition that mimic the way the human brain processes visual information, due to their ability to capture spatial patterns in data [3]. CNNs consist of layers that apply filters to input data, extracting important features across various dimensions. In the context of fMRI data, CNNs are effective at analyzing brain activity by identifying spatial features from different brain regions [14]. Their ability to handle complex input formats, such as 1D, 2D, and 3D data, makes them well-suited for fMRI-based ADHD detection. Typical CNN architectures consist of multiple layers, including convolutional and pooling layers for feature extraction, followed by fully connected layers and a classifier for final output prediction [5]. For example, Figure 1 shows the architecture of a 2D CNN with three convolutional layers, three pooling layers, one fully connected layer and a softmax classifier.

Several studies have applied CNNs to ADHD diagnosis, with promising results. For instance, Zhang et al. [22] developed a Separated Channel Attention CNN that achieved 68.9% accuracy, while Riaz et al. [17] introduced the DeepFMRI model, reaching 73.1% accuracy. CNNs are particularly useful for identifying abnormal patterns in brain connectivity, distinguishing between individuals with ADHD and healthy controls by focusing on spatial features in fMRI data. However, these models often analyze static brain images, limiting their ability to capture temporal dynamics in brain activity.

To overcome this limitation, 3D CNNs have been introduced to analyze entire brain volumes, using three-dimensional filters to detect patterns across the brain [20]. This approach allows for more comprehensive spatial analysis, as demonstrated by Zou et al. [23], who used a 3D CNN model to classify ADHD subjects, outperforming previous methods. Additionally, 4D CNNs have emerged to capture both spatial and temporal patterns in brain activity, providing a more complete analysis of fMRI data. For example, Mao et al. [15] developed a 4D CNN with LSTM integration to analyze time-series fMRI data, achieving 71.3% accuracy in ADHD detection.

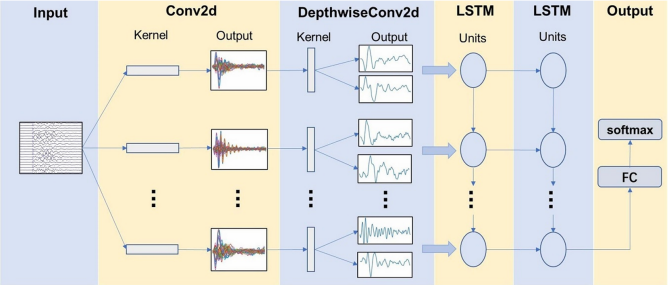

Figure 2: Overall visualization of the CNN-LSTM model architecture [21]

D. CNN-LSTM Algorithms for ADHD Detection

Long Short-Term Memory (LSTM) networks (LSTMs), which are specialized RNNs, address the vanishing gradient problem common in traditional RNNs [3]. Developed by Hochreiter and Schmidhuber [11], LSTMs use gates (input, output, forget, update) and a cell to effectively manage information flow and capture long-term dependencies in sequential data [5, 3]. This enables them to retain information over extended periods [12].

Hybrid CNN-LSTM models leverage the strengths of both architectures: CNNs for spatial feature extraction and LSTMs for temporal modeling [15]. This synergy is particularly valuable for fMRI analysis, where capturing spatiotemporal dynamics is crucial. CNN layers extract spatial features from brain images, while LSTM layers analyze the temporal relationships between these features, enabling the model to understand changes in brain activity over time [15]. 3D CNN-LSTM models further enhance this by incorporating 3D convolutional layers to analyze spatial patterns across all three dimensions of fMRI data.

Several studies have explored the use CNN-LSTM models to classify ADHD, showing promising results. For example, Wang et al. [21] used a CNN-LSTM model on EEG data, achieving a high accuracy of 98.23% in classifying ADHD patients from healthy individuals and identifying ADHD subtypes (see Figure 2). Similarly, Dey et al. [8] combined a capsule network with LSTM layers for fMRI data and reached 77% accuracy. In another study, Khullar et al. [14] used a 2D CNN-LSTM model on the ADHD-200 fMRI dataset, achieving an accuracy of 96%. These results highlight how CNN-LSTM models can capture both spatial and temporal patterns in EEG and fMRI data, making them useful tools for ADHD diagnosis.

Despite their significant potential, CNN-LSTM models require careful tuning and considerable computational resources. For example, adjusting learning rates, dropout rates, and batch sizes helps prevent overfitting, where models may memorize noise instead of learning meaningful patterns [14]. Furthermore, the high dimensionality of fMRI data needs robust computational capabilities to ensure effective training and generalization to new, unseen data [16]. However, with the right configuration and sufficient data, CNN-LSTM models can achieve promising results in distinguishing between ADHD and non-ADHD subjects [8][14][21].

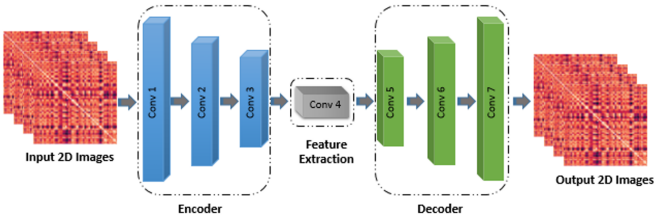

Figure 3: 2D CNN-AE model for diagnosis of SZ from rs-fMRI modality [19]

E. CNN-Based Autoencoders: Reconstruction and Pattern Detection in fMRI

To date, various deep learning models have been developed with a range of training methods, including supervised, semi-supervised, and unsupervised approaches. Autoencoders (AEs), a key type of deep learning model, are trained using unsupervised learning and are commonly applied for feature extraction [13]. A convolutional autoencoder (CNN-AE) is a type of neural network made up of an encoder and a decoder with convolutional layers. These models are designed to compress high-dimensional data, like fMRI data, into a lower-dimensional latent space (encoder), and subsequently reconstruct the original data from this compressed representation (decoder) [5].

CNN-based autoencoders are effective for analyzing fMRI data, particularly in ADHD classification. By applying convolutional filters, these autoencoders capture essential spatial features in the data, reducing noise and dimensionality [19]. This simplification helps identify key patterns associated with ADHD, like specific brain region activity. Studies have shown that combining autoencoders with CNNs improves classification accuracy [5]. For example, Chen et al. [6] demonstrated significant accuracy improvements using an Attention Auto-Encoding Network (Att-AENet) that prioritizes relevant brain connections. This approach achieved a remarkable 98.9% accuracy [7]. Another study by Shoeibi et al. [19] used a CNN-based autoencoder with Interval Type-2 Fuzzy Regression, achieving 72.71% accuracy on the UCLA dataset. Figure 3 shows their 2D CNN-based autoencoder for the diagnosis of schizophrenia.

To summarize, this section has covered the foundational concepts behind CNN, CNN-AE, and CNN-LSTM models, emphasizing how they bring a fresh approach to ADHD detection by capturing spatial and temporal brain activity patterns. By pre-training the CNN-AE and integrating CNN-LSTM, our model aims to achieve improved accuracy in identifying ADHD from fMRI data. Next, we move to the methodology section, where we detail our preprocessing steps, feature extraction process, and classification approach within the proposed architecture.

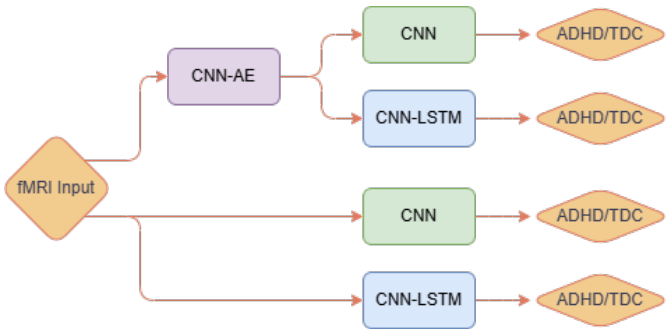

Figure 4: Overview of the experimental workflow

III. Methodology

In this section, we describe the methodology used in our study, which is divided into three parts: an overview of the experimental design, a detailed look at the algorithms implemented, and additional information relevant to data processing and experimental setup, which serve as the experiment's constant factors. Each subsection provides insights into how the models were constructed and optimized to address the specific challenges of our data.

A. Experimental Design

This study uses a 2x2 experimental design to explore how two factors influence the effectiveness of ADHD classification using fMRI data. As shown in Figure 4, the two factors are:

1) **Use of Pretrained Weights (With CNN-AE vs. Without CNN-AE):** This factor tests whether the inclusion of a pretrained CNN autoencoder (CNN-AE) for data compression improves classification accuracy compared to models trained without it.
2) **Inclusion of LSTM Layers (With LSTM vs. Without LSTM):** This factor examines whether adding LSTM layers to capture temporal information from the fMRI sequences enhances classification accuracy compared to models that rely only on CNN layers.

This 2x2 setup results in four combinations, allowing for an analysis of how each factor individually and in combination affects model performance. Specifically, it enables us to see if either the pretrained CNN-AE or the addition of LSTM layers has a greater impact, or if the combination of both yields the best classification results.

B. Model Architecture and Artifacts

1) Feature Extraction: Our CNN-AE has two main components: an encoder and a decoder (see Table II). The encoder consists of four sequential 3D convolutional layers that progressively reduce the spatial dimensions, starting with 64 filters and scaling up to 512. This design enables the model to learn complex, high-level spatial patterns in brain activity. Each convolutional layer uses a $3 \times 3 \times 3$ kernel and is followed by batch normalization and ReLU activation, stabilizing learning and retaining essential features. The decoder mirrors the encoder, using 3D transpose convolutional layers to reconstruct the data back to its original dimensions. The final output shape, `[4, 1, 61, 73, 61]`, represents four fMRI samples, each with a single channel and spatial dimensions of $61 \times 73 \times 61$, matching the input format.

Component	Layer	Type	Channels
Encoder	1	`Convolutional 3D` `Batch Norm`	64
	2	`Convolutional 3D` `Batch Norm`	128
	3	`Convolutional 3D` `Batch Norm`	256
	4	`Convolutional 3D` `Batch Norm`	512
Decoder	1	`Convolutional Transposed 3D` `Batch Norm`	512
	2	`Convolutional Transposed 3D` `Batch Norm`	256
	3	`Convolutional Transposed 3D` `Batch Norm`	128
	4	`Convolutional Transposed 3D`	64
Activation		ReLU	
Classifier		Sigmoid	

Table II: Architecture of Convolutional Autoencoder

We train the model using Mean Squared Error (MSE) to minimize reconstruction error and support effective denoising and extraction of critical brain patterns. Training is optimized with the Adam optimizer (learning rate = 0.00001, weight decay = 1e-5), with a scheduler reducing the learning rate by 0.1 every 10 epochs. This setup enables the CNN-AE to generate ADHD-relevant features for improved classification in later stages.

2) Classification Models: We explored four CNN-based architectures for ADHD classification:

1) **Pure CNN (Table III):** This model consists of three 3D convolutional layers with increasing filters (64 to 256) to capture spatial features. Each layer uses a $(3 \times 3 \times 3)$ kernel, batch normalization, and ReLU activation. An Adaptive Max Pooling layer reduces dimensions, followed by a fully connected layer (256 neurons) and a dropout layer (probability 0.5) to prevent overfitting.
2) **CNN with AE (Table III):** This architecture utilizes pretrained weights from a CNN-AE (described below) for enhanced feature extraction. It features three 3D convolutional layers with decreasing filters (512 to 64) to refine the pretrained features. The remaining structure mirrors the pure CNN, including the pooling, fully connected, and dropout layers.
3) **Pure CNN-LSTM (Table III):** This model integrates convolutional layers with LSTM units to capture both spatial and temporal features. Three 3D convolutional layers with increasing filters (64 to 256) extract spatial features. An Adaptive Max Pooling layer reduces dimen-

sions while preserving temporal information (sequence length of 60). A two-layer LSTM (hidden size 64) then learns temporal dependencies. This is followed by a fully connected layer and a dropout layer.

4) **CNN-LSTM with AE (Table III):** This architecture combines the CNN-LSTM structure with pretrained weights from the CNN-AE. It uses three 3D convolutional layers with decreasing filters (512 to 64) to refine pretrained features. The remaining structure, including the pooling layer, LSTM, fully connected layer, and dropout layer, is identical to the pure CNN-LSTM.

Model	Layer	Parameters
Pure CNN	`Convolutional 3D`	64
	`Batch Norm`	–
	`Convolutional 3D`	128
	`Batch Norm`	–
	`Convolutional 3D`	256
	`Batch Norm`	–
	`Adaptive Pooling`	$10 \times 10 \times 10$
	`Linear`	256
	`Dropout`	0.5
	`Linear`	2
CNN with AE	`Convolutional 3D`	256
	`Batch Norm`	–
	`Convolutional 3D`	128
	`Batch Norm`	–
	`Convolutional 3D`	64
	`Batch Norm`	–
	`Adaptive Pooling`	$10 \times 10 \times 10$
	`Linear`	256
	`Dropout`	0.5
	`Linear`	2
Pure CNN-LSTM	`Convolutional 3D`	64
	`Batch Norm`	–
	`Convolutional 3D`	128
	`Batch Norm`	–
	`Convolutional 3D`	256
	`Batch Norm`	–
	`Adaptive Pooling`	$10 \times 10 \times 60$
	`LSTM`	in = $256 \times 10 \times 10$, sequence = 60
	`Linear`	256
	`Dropout`	0.5
	`Linear`	2
CNN-LSTM with AE	`Convolutional 3D`	256
	`Batch Norm`	–
	`Convolutional 3D`	128
	`Batch Norm`	–
	`Convolutional 3D`	64
	`Batch Norm`	–
	`Adaptive Pooling`	$10 \times 10 \times 60$
	`LSTM`	in = $64 \times 10 \times 10$, sequence = 60
	`Linear`	256
	`Dropout`	0.5
	`Linear`	2
Activation	ReLU	–

Table III: Architectures of Classification Models

All four models use cross-entropy loss with class weights of $[0.8004, 1.3323]$ to balance the dataset. We use the Adam optimizer with a learning rate of 0.00001 and a scheduler that reduces the rate by a factor of 0.1 every 10 epochs, improving classification performance by effectively capturing both spatial and temporal features.

C. Constant Factors

1) Dataset: The ADHD-200 Preprocessed Dataset from DPARSF serves as the primary data source for this study. This dataset is publicly available and comprises fMRI scans from children diagnosed with ADHD and typically developing control (TDC) subjects. The inclusion of both functional and structural MRI data enables a comprehensive analysis for identifying patterns associated with ADHD.

2) Environment: All models are implemented using PyTorch, a popular deep learning framework known for its flexibility and dynamic computation graph capabilities. This choice allows for efficient model building, training, and evaluation. The models are run using Windows Subsystem for Linux (WSL) on a system equipped with a 4GB GeForce GTX GPU and an Intel Core i7 9th generation processor.

3) Data Preprocessing: fMRI data preprocessing is performed using a custom `FMRI_Dataset` class. This process involves loading fMRI volumes using NiBabel and converting them to NumPy arrays. Optional Gaussian smoothing with a configurable `smoothing_sigma` parameter is applied to reduce high-frequency noise and enhance data consistency. Data augmentation techniques, including random Gaussian noise addition, random rotations along spatial axes, and intensity shifts, are employed to increase data diversity and model robustness. Normalization through standardization (mean subtraction and division by standard deviation) is then performed to reduce inter-subject variability. Finally, the data is converted to PyTorch tensors and padded to ensure consistent dimensions of `[4, 1, 61, 73, 71]`.

4) Testing: After preprocessing, the data is divided into training, validation, and test sets using stratified sampling to ensure balanced class distributions across subsets. Typically, 70% of the data is allocated for training, 15% for validation, and 15% for testing. This split ensures that the models can be effectively evaluated and generalized to unseen data.

5) Evaluation Metrics: Model performance is assessed using accuracy (overall correctness by the proportion of correctly predicted cases), sensitivity (model's ability to detect ADHD cases), specificity (model's ability to detect non-ADHD cases), and AUC-ROC (model's ability to distinguish between ADHD and non-ADHD across different thresholds). These metrics provide a comprehensive view of model performance, balancing overall accuracy with its effectiveness in distinguishing ADHD from control cases.

IV. Results

The performance of the models was evaluated using accuracy, loss, sensitivity (recall), specificity, and AUC metrics, as summarized below. These metrics provide a comprehensive assessment of each model's ability to classify ADHD from

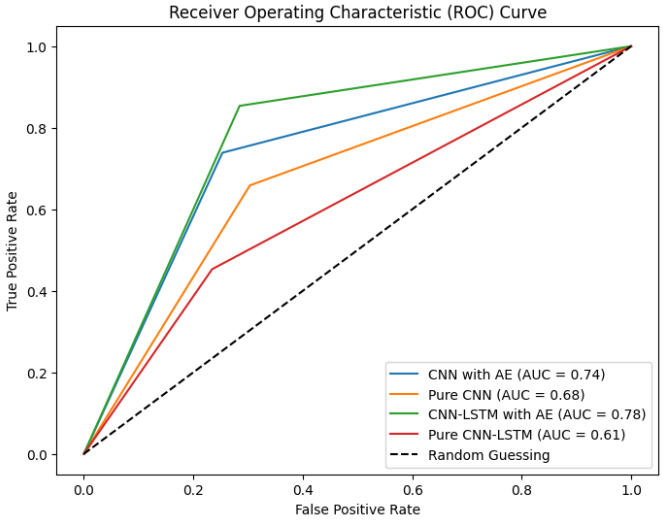

Figure 5: Comparative ROC Curve

fMRI data. Table IV summarizes the performance of each model.

Model	Accuracy	Sensitivity	Specificity	AUC
CNN with AE	74.38%	0.74	0.75	0.74
Pure CNN	68.24%	0.66	0.70	0.68
CNN-LSTM with AE	76.73%	0.85	0.72	0.78
Pure CNN-LSTM	64.84%	0.45	0.77	0.61

Table IV: Performance Metrics for the Models on the ADHD-200 Dataset

The results indicate that the use of pretrained CNN-AE weights improves classification performance across both CNN and CNN-LSTM architectures. This is reflected in the AUC scores (see Figure 5), with the pretrained models exhibiting notably higher values. The CNN-LSTM with AE achieved the best results, with an accuracy of 76.73%, a sensitivity of 0.85, and an AUC of 0.78, highlighting its strong ability to detect ADHD cases. This makes it particularly effective in scenarios where minimizing false negatives is critical, even if it means a slightly higher rate of false positives (specificity of 0.72). The CNN with AE also performed well, achieving an accuracy of 74.38%, a sensitivity of 0.74, and a balanced specificity of 0.75, as evidenced by its AUC of 0.74. This suggests it is effective at correctly identifying both ADHD and non-ADHD cases. In contrast, the pure CNN, with an AUC of 0.68, demonstrated only moderate performance, with an accuracy of 68.24% and a relatively balanced sensitivity (0.66) and specificity (0.70). This weaker performance underscores the importance of pretrained AE weights for effective feature extraction. Finally, the pure CNN-LSTM exhibited the weakest performance, with the lowest accuracy (64.84%), sensitivity (0.45), and AUC (0.61). While it had the highest specificity (0.77), its inability to identify ADHD cases effectively highlights the challenges of relying solely on temporal modeling without robust feature extraction.

These results emphasize the value of pretrained AE weights,

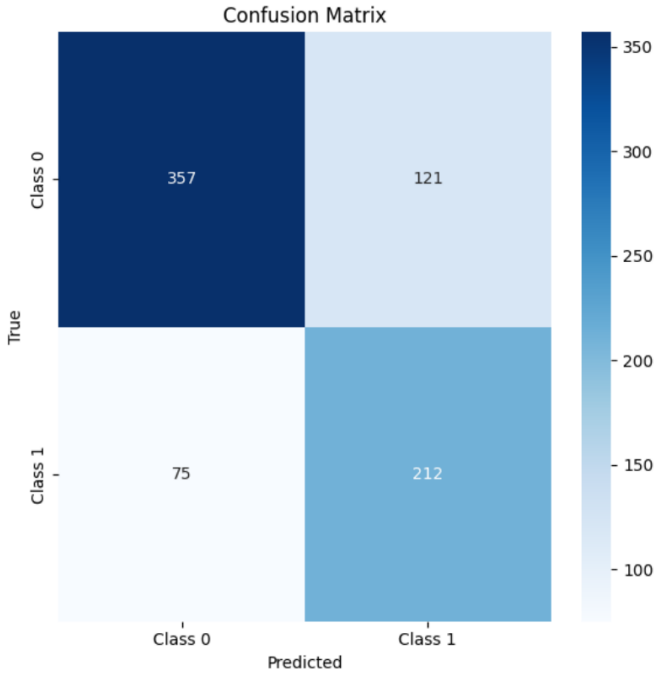

Figure 6: Confusion Matrix of the CNN with Pretrained Weights from AE

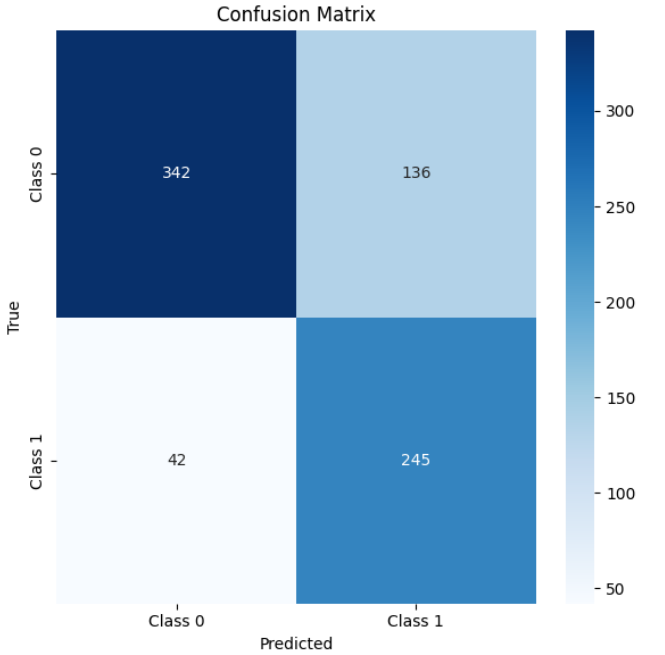

Figure 7: Confusion Matrix of the CNN-LSTM with Pretrained Weights from AE

as both CNN and CNN-LSTM models with AE significantly outperformed their pure counterparts. Figures 6 to 9 provide confusion matrices for each model, offering further insights into their strengths and weaknesses.

Comparing the CNN (see Figure 6) and CNN-LSTM (see Figure 7) models (both with AE pretraining), the CNN shows

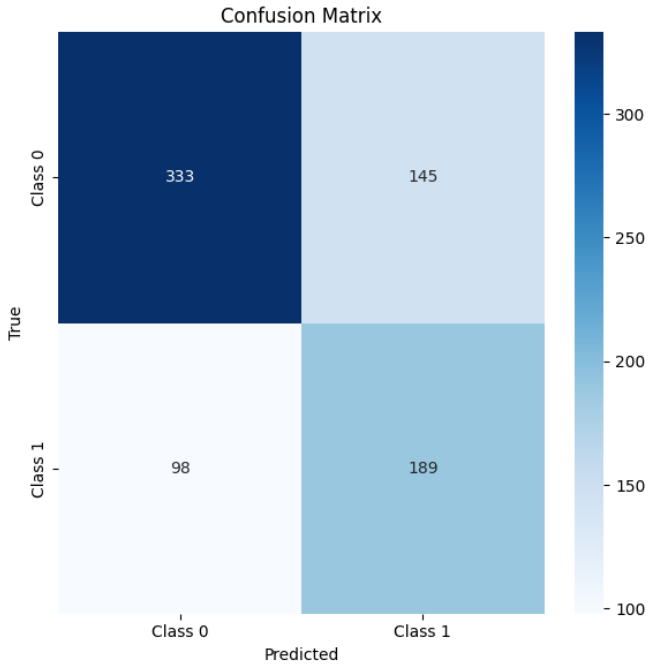

Figure 8: Confusion Matrix of the Pure CNN

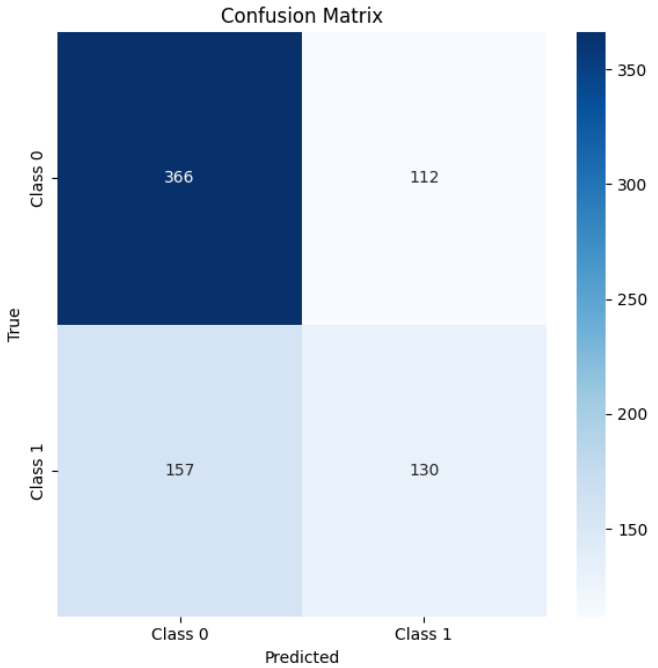

Figure 9: Confusion Matrix of the Pure CNN-LSTM

slightly higher specificity (72.59% vs 71.55% in identifying non-ADHD), while the CNN-LSTM exhibits slightly better sensitivity (85.37% vs 86.06% in detecting ADHD). However, the CNN-LSTM has slightly higher false positive (28.45% vs 27.41%) and false negative (14.63% vs 13.94%) rates. Though both models perform similarly overall, the CNN's ability to effectively leverage spatial features may be slightly more advantageous for this task.

Looking at the pure CNN (see Figure 8) and pure CNN-LSTM (see Figure 9) models, the pure CNN demonstrates a better ability to correctly identify ADHD cases (65.85% vs 45.3%), but it also incorrectly labels more healthy individuals as having ADHD (30.33% vs 23.43%). The pure CNN-LSTM is indeed more cautious in diagnosing ADHD, with a lower false positive rate (23.43% vs 30.33%), but this comes at the cost of missing a significant number of actual ADHD cases, as evidenced by its higher false negative rate (54.7% vs 34.15%).

When comparing all models (see Figure 5), a clear trend emerges: incorporating pretrained components significantly boosts performance. However, while both pretrained models perform well, they have different strengths. The CNN-LSTM with AE achieves higher sensitivity, making it ideal for applications where minimizing false negatives is crucial, such as screening for potential ADHD cases. However, this comes at the cost of a slightly higher false positive rate, potentially indicating some degree of overfitting due to the increased complexity of the model and its potential to capture noise or irrelevant temporal patterns in the data. Careful hyperparameter tuning and regularization techniques could help mitigate this issue. On the other hand, the CNN with AE offers a more balanced performance between sensitivity and specificity, making it suitable for scenarios requiring accurate identification of both ADHD and non-ADHD individuals, such as in a clinical diagnostic setting.

Interestingly, adding temporal complexity with LSTMs did not yield substantial gains beyond increased sensitivity. This could be because the temporal information in fMRI is subtle and complex, or because the dataset used in this study may not have captured enough relevant temporal variations. Nonetheless, this trend suggests that spatial features, effectively captured by the CNN with AE, may be more critical for ADHD classification from fMRI data. This notion is further supported by the poor performance of models lacking pretraining with CNN layers, highlighting the crucial role of robust feature extraction in fMRI analysis.

V. Conclusions

This study explored various CNN architectures, including models with and without pretrained weights, and their integration with LSTM networks for ADHD classification from fMRI data. The results highlight the benefits of transfer learning, as models with pretrained weights demonstrated faster convergence and higher accuracy. Incorporating LSTM layers further enhanced the ability to capture temporal dependencies, improving classification performance.

While spatial features showed greater relevance in ADHD classification, this study's exploration of temporal information was limited by computational constraints. Future work should optimize LSTM configurations, including the number of layers and sequence lengths, to better analyze temporal patterns.

These findings offer valuable insights for advancing fMRI-based diagnostic models. Future efforts could focus on reduc-

ing false positives in CNN-LSTM architectures through hyperparameter tuning and exploring strategies such as attention mechanisms or multimodal data integration to further enhance diagnostic accuracy.

REFERENCES

[1] Elie Abdelnour, Madeline O Jansen, and Jessica A Gold. "ADHD diagnostic trends: Increased recognition or overdiagnosis?" en. In: *Mo. Med.* 119.5 (Sept. 2022), pp. 467–473.

[2] ADHD-200 Consortium. "The ADHD-200 consortium: A model to advance the translational potential of neuroimaging in clinical neuroscience". en. In: *Front. Syst. Neurosci.* 6 (Sept. 2012), p. 62.

[3] Ali Agga et al. "CNN-LSTM: An efficient hybrid deep learning architecture for predicting short-term photovoltaic power production". en. In: *Electric Power Syst. Res.* 208.107908 (July 2022), p. 107908.

[4] Pierre Bellec et al. "The Neuro Bureau ADHD-200 Preprocessed repository". en. In: *Neuroimage* 144 (Jan. 2017), pp. 275–286.

[5] Yoshua Bengio. *Deep Learning*. Adaptive Computation and Machine Learning series. London, England: MIT Press, Nov. 2016.

[6] Nan Chen and Yun Jiao. "Deep Learning of Automatic Encoder Based on Attention for ADHD Classification of Brain MRI". In: *2023 7th International Conference on Biomedical Engineering and Applications (ICBEA)*. 2023, pp. 11–14. DOI: 10.1109/ICBEA58866.2023.00010.

[7] Ying Chen et al. "ADHD classification combining biomarker detection with attention auto-encoding neural network". In: *Biomedical Signal Processing and Control* 84 (2023), p. 104733. ISSN: 1746-8094. DOI: https://doi.org/10.1016/j.bspc.2023.104733. URL: https://www.sciencedirect.com/science/article/pii/S1746809423001660.

[8] Arunav Dey et al. "Discrimination of Attention Deficit Hyperactivity Disorder Using Capsule Networks and LSTM Networks on fMRI Data". In: *Engineering Applications of Neural Networks*. Ed. by Lazaros Iliadis et al. Cham: Springer Nature Switzerland, 2023, pp. 291–302. ISBN: 978-3-031-34204-2.

[9] Fatemeh Dehghani Firouzabadi et al. "Neuroimaging in attention-deficit/hyperactivity disorder: Recent advances". en. In: *AJR Am. J. Roentgenol.* 218.2 (Feb. 2022), pp. 321–332.

[10] C Thomas Gualtieri and Lynda G Johnson. "ADHD: Is objective diagnosis possible?" en. In: *Psychiatry (Edgmont)* 2.11 (Nov. 2005), pp. 44–53.

[11] S Hochreiter. "Long Short-term Memory". In: *Neural Computation MIT-Press* (1997).

[12] Shantani Kannan, Kannan Subbaram, and Md Faiyazuddin. "Artificial intelligence in vaccine development: Significance and challenges ahead". en. In: *A Handbook of Artificial Intelligence in Drug Delivery*. Elsevier, 2023, pp. 467–486.

[13] Fahime Khozeimeh et al. "Combining a convolutional neural network with autoencoders to predict the survival chance of COVID-19 patients". en. In: *Sci. Rep.* 11.1 (July 2021), p. 15343.

[14] Vikas Khullar et al. "Deep learning-based binary classification of ADHD using resting state MR images". en. In: *Augment. Hum. Res.* 6.1 (Dec. 2021).

[15] Zhenyu Mao et al. "Spatio-temporal deep learning method for ADHD fMRI classification". en. In: *Inf. Sci. (Ny)* 499 (Oct. 2019), pp. 1–11.

[16] Abdul Qayyum et al. "Correction to: An efficient 1DCNN–LSTM deep learning model for assessment and classification of fMRI-based autism spectrum disorder". In: *Innovative Data Communication Technologies and Application*. Singapore: Springer Nature Singapore, 2022, pp. C1–C1.

[17] Atif Riaz et al. "DeepFMRI: End-to-end deep learning for functional connectivity and classification of ADHD using fMRI". en. In: *Neuroscience Methods* 335.108506 (Apr. 2020), p. 108506.

[18] Katya Rubia. "Cognitive neuroscience of Attention Deficit Hyperactivity Disorder (ADHD) and its clinical translation". en. In: *Front. Hum. Neurosci.* 12 (Mar. 2018), p. 100.

[19] Afshin Shoeibi et al. "Automatic diagnosis of schizophrenia and attention deficit hyperactivity disorder in rs-fMRI modality using convolutional autoencoder model and interval type-2 fuzzy regression". en. In: *Cogn. Neurodyn.* 17.6 (Dec. 2023), pp. 1501–1523.

[20] Gurcan Taspinar and Nalan Ozkurt. "A review of ADHD detection studies with machine learning methods using rsfMRI data". en. In: *NMR Biomed.* 37.8 (Aug. 2024), e5138.

[21] Cheng Wang et al. "Towards high-accuracy classifying attention-deficit/hyperactivity disorders using CNN-LSTM model". en. In: *J. Neural Eng.* 19.4 (July 2022), p. 046015.

[22] Tao Zhang et al. "Separated channel attention convolutional neural network (SC-CNN-attention) to identify ADHD in multi-site rs-fMRI dataset". en. In: *Entropy (Basel)* 22.8 (Aug. 2020), p. 893.

[23] Liang Zou et al. "3D CNN based automatic diagnosis of attention deficit hyperactivity disorder using functional and structural MRI". In: *IEEE Access* 5 (2017), pp. 23626–23636.