GRAPH-BASED MANIFOLD LEARNING FOR MOTOR IMAGERY CLASSIFICATION USING EEG SIGNALS

Capstone Project Report

MID SEMESTER EVALUATION

Submitted by:

102103484 VANSHIKA MITTAL

102103493 ISHITA SUCHDEVA

102103496 SELINA VARSHNEY

102103510 VANSHIKA NARANG

102103237 AARUSHI KAMBOJ

BE Fourth Year, CoE

CPG No: 160

Under the Mentorship of

Dr. Anurag Tiwari

Assistant Professor

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

Computer Science and Engineering Department
Thapar Institute of Engineering and Technology, Patiala

July 2024

ABSTRACT

In the past few years, the investigation of novel deep-learning methods has dramatically improved the field of EEG-based motor imagery classification for BCI systems which is highly essential for such interfaces. In our proposed approach, we aim at improving motor imagery classification from EEG signals with GNN, finding a manifold in the data dimension with BILSTM and constructing a Deep Belief Attention Network (DBAN). These three effective techniques are brought together with the aim of using in-depth spatial and temporal information as well as increasing dimensionality while at the same time capturing complicated patterns of the EEG data for better classification performance.

Primarily, GNNs, [1] which are exploited to establish spatial connections between EEG data, take into account the graph-based representation of the relationship between the electrode. The key advantage of GNNs is carefully modeling the interactions of EEG channels so that the features crucial for an accurate prediction are extracted. Second, we employ Bi-Directional Long ShortTerm Memory (BI-LSTM) model which is used for multivariable learning helping to model the temporal features of the EEG signals. BiLSTM models are suitable for those situations when it is necessary to capture long-range dependency and subtle temporal patterns of EEG time series. Moreover, DBAN, a novel Deep Belief Attention Network combining the strengths of deep belief networks and attention mechanisms, is also defined. In hierarchical representations of vital EEG features, attention is paid to only the regions of the input data that are of importance. It is attention mechanism that powers the neural network to detect discriminative actions and consequently boosts classification accuracy. The proposed set-up is evaluated by scouring open EEG motor imagery data and it is proven to be more effective than existing methods. Given the presented experimental results, it can be stated that the proposed method is highly effective in the accurate classification of motor imagery tasks, thus the issue of using spatial, temporal, and hierarchical representations in EEG-based BCI systems becomes very urgent. The proposed framework holds promise for advancing the development of robust and efficient EEG-based motor imagery classification systems with potential applications in neurorehabilitation, assistive technology, and brain-controlled interfaces.

DECLARATION

We hereby declare that the design principles and working prototype model of the project entitled "Graph-Based Manifold Learning For Motor Imagery Classification using EEG Signals" is an authentic record of our own work carried out in the Computer Science and Engineering Department, TIET, Patiala, under the guidance of Dr. Anurag Tiwari during 6th semester (2024).

Date: 23 August, 2024

Roll No.	Name	Signature
102103484	Vanshika Mittal	
102103493	Ishita Suchdeva	
102103496	Selina Varshney	
102103510	Vanshika Narang	
102103237	Aarushi Kamboj	

Counter Signed By:

Faculty Mentor:

Anurag Tiwari

Dr. Anuraj Tiwari

Designation: Assistant Professor

CSED,

TIET, Patiala

ACKNOWLEDGEMENT

We would like to express our thanks to our mentor Dr. Anurag Tiwari. He has been of great help in our venture and an indispensable resource of technical knowledge. He is truly an amazing mentor to have.

We are also thankful to Dr, Ashima Singh, Head, Computer Science and Engineering Department, the entire faculty and staff of the Computer Science and Engineering Department, and also our friends who devoted their valuable time and helped us in all possible ways towards successful completion of this project. We thank all those who have contributed either directly or indirectly towards this project.

Lastly, we would also like to thank our families for their unyielding love and encouragement. They always wanted the best for us and we admire their determination and sacrifice.

Date: 23 August, 2024

Roll No.	Name	Signature
102103484	Vanshika Mittal	
102103493	Ishita Suchdeva	
102103496	Selina Varshney	
102103510	Vanshika Narang	
102103237	Aarushi Kamboj	

.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

BILSTM	Bi Directional Long Short Term Memory
EEG	Electroencephalography
GNN	Graph Neural Network
DBAN	Deep Belief Attention Network

TABLE OF CONTENTS

4. Project Overview

The exploration of advanced deep learning techniques has significantly improved EEG-based motor imagery classification, a cornerstone of brain-computer interface (BCI) systems. Our project seeks to further enhance motor imagery classification by leveraging a novel approach that integrates Graph Neural Networks (GNNs), Bi-directional Long Short-Term Memory (BiLSTM) networks, and Deep Belief Attention Networks (DBAN). These three techniques aim to capture and utilize the spatial, temporal, and hierarchical representations within EEG signals to improve classification accuracy. This approach not only tackles the challenge of decoding motor imagery from EEG signals but also addresses issues related to channel selection, feature reduction, and model generalization.

The cornerstone of this approach lies in the GNN, which is employed to establish spatial connections between EEG channels. By considering the relationships among electrodes as a graph, the GNN effectively models the interactions between EEG channels, enabling the extraction of spatial features critical for accurate predictions. The graph-based representation captures complex spatial dependencies, allowing for a more nuanced understanding of the brain's spatial dynamics during motor imagery tasks. The key advantage of the GNN lies in its ability to model these interactions while focusing on the most relevant features, which is essential for improving classification performance.

Temporal dependencies within the EEG signals are captured using a BiLSTM network. The BiLSTM is particularly well-suited for this task due to its ability to model long-range dependencies and subtle temporal patterns inherent in EEG data. By processing the data in both forward and backward directions, the BiLSTM network can capture temporal information more comprehensively, thus improving the model's understanding of the temporal dynamics associated with motor imagery tasks. This dual-directional processing allows the network to retain information from both past and future time steps, which is crucial for accurately decoding the complex time-series data present in EEG signals.

In addition to spatial and temporal modeling, the proposed approach includes the development of a Deep Belief Attention Network (DBAN), which combines the strengths of deep belief networks with attention mechanisms. Deep belief networks, composed of multiple layers of Restricted Boltzmann Machines (RBMs), provide hierarchical representations of features, making them well-suited for capturing complex relationships and dependencies in EEG data. The integration of attention mechanisms allows the network to focus on the most relevant regions of the input data, thereby enhancing the model's ability to detect discriminative actions and improving overall classification accuracy. This attention-driven approach is particularly beneficial in identifying the most informative features within the EEG signals, leading to a more precise and reliable classification of motor imagery tasks.

To further improve classification accuracy, the proposed methodology also includes a novel approach to channel selection and feature reduction. Channel selection is performed using both filter and wrapper methods. The filter method selects features based on statistical measures, while the wrapper method treats feature selection as a search problem, evaluating different combinations of features to identify the most relevant subset. Techniques such as forward selection, backward selection, exhaustive feature selection, and recursive feature elimination are utilized to iteratively train the algorithm on a subset of features, thereby reducing noise and redundant signals. This approach not only improves classification accuracy but also enhances the efficiency of the overall model by focusing on the most relevant EEG channels.

Feature reduction is further refined through the use of autoencoders, which learn compressed representations of the data, extracting the underlying structure while eliminating redundant or noisy information. Manifold learning techniques are employed to capture the non-linear structure of the data, which traditional linear approaches like principal component analysis (PCA) might miss. This pipeline architecture for feature reduction ensures that the model retains only the most relevant features, thereby improving its ability to accurately classify motor imagery tasks.

The proposed methodology includes a comprehensive approach to training and evaluation. The model is trained on multiple datasets with varying numbers of EEG electrodes (22, 59, and 118 electrodes), ensuring that it can generalize across different configurations. The use of multiclass data from open BCI and BCI competition datasets provides a robust training ground for the model, enabling it to learn from a wide variety of motor imagery tasks. Additionally, the proposed setup is evaluated against existing methods, demonstrating its superiority in terms of classification accuracy and robustness.

The spatial and temporal methods used in the project are designed to capture the intricate patterns within EEG data that are essential for accurate motor imagery classification. The spatial neural network design incorporates residual connections and dilated convolutions, allowing the network to capture features at different spatial scales while mitigating the vanishing gradient problem. Multi-resolution pathways further enhance the network's ability to capture features at varying scales, providing a more comprehensive representation of the spatial dynamics within the EEG data.

On the temporal side, the BiLSTM network is augmented with skip connections and attention mechanisms to focus on the most informative temporal segments. This approach allows the network to adaptively attend to relevant temporal scales during feature extraction, thereby enhancing its ability to capture the complex temporal dependencies inherent in EEG signals. Residual LSTM blocks are also employed to ease the flow of gradients through the network, further mitigating the vanishing gradient problem and ensuring robust temporal representations.

The fusion of multi-scale features is another key aspect of the proposed methodology. Feature pyramid fusion techniques merge features extracted at different spatial and temporal scales, creating a comprehensive feature representation that captures information across multiple dimensions. Attention mechanisms are employed to dynamically combine multi-scale features based on their relevance to the task, enhancing the discriminative power of the model while reducing the impact of irrelevant information. In cases where multichannel data is used, graph-based fusion techniques are leveraged to integrate features across different channels effectively, further improving the model's ability to capture the spatial relationships between EEG channels.

Regularization and optimization techniques are also incorporated to ensure the model’s performance is robust and generalizable. Gradient clipping is applied to prevent exploding gradients during training, while dropout and batch normalization layers are used to regularize the network and stabilize training. Learning rate scheduling techniques are employed to adaptively adjust the learning rate during training, promoting faster convergence and better generalization.

The proposed framework holds significant promise for advancing the development of robust and efficient EEG-based motor imagery classification systems. With potential applications in neurorehabilitation, assistive technology, and brain-controlled interfaces, this project addresses a critical need in the domain of assistive technology and neurorehabilitation. By decoding neural activity associated with motor imagery, this project seeks to translate mental commands into actionable outputs, empowering individuals with paralysis or severe motor impairments to interact, communicate, and perform daily tasks more independently.

Furthermore, the proposed technology has broader implications beyond assistive technology. It could revolutionize consumer electronics interfaces, enabling control of smart devices and computers through mental commands. In addition, the technology could enhance gaming and virtual reality experiences, neurofeedback therapy, and brain-computer music interfaces, offering new avenues for innovation and improving personal well-being through thought-controlled interfaces and therapies.

In summary, this project aims to improve motor imagery classification from EEG signals through a novel integration of GNNs, BiLSTM, and DBAN, combined with advanced techniques for channel selection, feature reduction, and model training. By capturing and leveraging the spatial, temporal, and hierarchical representations within EEG data, the proposed framework promises to deliver superior classification accuracy and robustness, with far-reaching implications for assistive technology, neurorehabilitation, and beyond.

5. Need Analysis

- Our project addresses a critical need in the domain of assistive technology and neurorehabilitation. With aging, accidents and rising neurological issues, more individuals face paralysis or motor impairments, and struggle with speech and gestures. By decoding the neural activity associated with motor imagery such as imagining moving their left hand or right leg, this project seeks to translate their mental commands into actionable commands.
- Our project acknowledges the fact that mapping brain signals from one individual to another presents unique challenges, with existing models often failing to deliver accuracy, particularly concerning spatial and temporal data.
- Previous models for channel selection and classification in EEG-based BCIs have been inefficient as they use many EEG channels, which can include noisy and redundant signals. Additionally, manually selecting channels based on neurophysiological knowledge may not always yield optimal results. [2] Our project addresses the inefficiencies in previous models by using filter and wrapper methods for channel selection. These techniques help us identify the most relevant EEG channels for our analysis, filtering out noise and redundant signals, and improving the efficiency and accuracy of our classification process.
- Installation of complex software for EEG-based BCIs can also be challenging and time-consuming, hence there is a need to make BCI software more accessible through user-friendly interfaces, such as websites and mobile apps, which we aim to provide.
- The proposed technology has vast potential beyond assistive technology. It can revolutionize consumer electronics interfaces, enabling control of smart devices and computers through mental commands. BCI could enhance gaming and virtual reality experiences, neurofeedback therapy and brain-computer music interfaces offer additional avenues for innovation. These advances could revolutionize how we interact with computers and boost personal well-being with thought-controlled interfaces and therapies.[3]

6. Research Gaps

- Capturing spatio-temporal features from multi-channel, high-density EEG signals is challenging, as previous work shows that temporal and spatial convolutions are often treated independently, making single-feature extraction methods inadequate.
- The complexity of neuro-mechanisms makes it difficult to define an explicit topological structure in EEG data, posing a challenge for GNN-based methods that rely on predefined graphs. [4]
- EEG data can be highly variable across different patients, making it difficult to create a one-size-fits-all model. This variability can be due to differences in electrode placement, brain anatomy, or the nature of the seizures themselves. [5]
- EEG signals are inherently non-stationary, with their statistical properties changing over time due to dynamic brain activity and various external influences. Additionally, EEG data is frequently contaminated with noise and artifacts from muscle movements, eye blinks, and other external sources. Deep learning models, particularly complex architectures, often require large amounts of clean, stationary data, which is difficult to obtain in practical EEG experiments. This gap highlights the need for more effective methods to address non-stationarity and noise in EEG data to improve model performance and reliability. [6]
- Explainability and Interpretability: While Grad-CAM is mentioned for interpretability, there may be other explainability methods that could be explored to provide a deeper understanding of the model's decision-making process, especially in the context of EEG signals.[7]
- Representation of EEG Signals: EEG is a type of multi-channel time series data, and while it has been treated as a 2D image in previous research using convolutional neural networks, this approach neglects the underlying graph structure that needs to be considered now. The challenge is how to represent the EEG signals in a way that captures the graph structure among the electrodes.
- Unknown Latent Graph Structure: The real latent graph structure among EEG channels is unknown. Although there is spatial positional correlation among electrodes, this does not necessarily reflect functional correlation. The challenge is to learn a weighted complete graph that can capture the functional correlation among EEG signals for classification tasks.[8]

7. Problem Definition and Scope

Individuals with paralysis or severe motor impairments struggle to interact, communicate, and perform daily tasks independently. Empowering them requires an assistive technology capable of decoding their brain signals to enable movement in the targeted body part.

8. Assumptions and Constraints

Table 1: Assumptions of Project

S.NO.	Assumptions
1	Data Quality: The EEG data collected from BCI competitions is assumed to be of high quality, with minimal noise, artifacts, and consistent labeling across datasets
2	Subject Variability: It is assumed that the variability in brain activity across different subjects can be managed effectively through advanced preprocessing and model generalization techniques.
3	Channel Selection: The selected channels (22, 59, and 118) are assumed to provide sufficient spatial resolution for capturing relevant motor imagery signals.
4	Preprocessing Efficacy: The preprocessing methods used (baseline correction, bandpass filtering, smoothing, etc.) are assumed to adequately remove noise and artifacts, allowing for accurate feature extraction.
5	Computational Resources: The project assumes access to sufficient computational resources for training complex models, such as GNNs, BiLSTMs, and DBANs, on large EEG datasets.
6	Transfer Learning Potential: The models are assumed to have the potential for effective transfer learning, allowing them to adapt to new subjects or datasets with minimal retraining.

Table 2: Constraints of Project

S.NO.	Constraints
1	Data Availability: The project relies on publicly available EEG datasets, which may limit the diversity and quantity of data available for training and validation.
2	Inter-Subject Variability: High variability in EEG signals across different subjects may constrain the model's ability to generalize effectively, requiring additional data or more sophisticated normalization techniques.
3	Real-Time Performance: The project may face constraints in achieving real-time performance due to the computational complexity of the proposed models, especially in applications requiring low-latency responses.
4	Model Complexity: The integration of spatial and temporal features through GNNs, BiLSTMs, and DBANs increases model complexity, which may lead to challenges in model interpretability and risk of overfitting.
5	Ethical Considerations: The project is constrained by ethical guidelines, ensuring that all data usage respects privacy and consent agreements.
6	Scalability: The designed system must be scalable for potential deployment in real-world applications, balancing accuracy with computational efficiency.

9. Standards

- IEEE Standard for Software Test Documentation:
  This standard defines the format and content of software test documentation, ensuring a comprehensive and standardized approach to documenting testing processes. Additionally, we have utilized the IEEE referencing stylesheet provided by our institute to maintain consistency and adherence to academic standards in documenting and presenting our project work.
- BCI standards:
  ISO/TS 80601-2-26: Particular Requirements for Basic Safety and Essential Performance of Electroencephalographs
  This standard primarily focuses on EEG devices used for data collection. Although it is hardware-centric, it provides valuable guidelines for ensuring the quality and integrity of EEG data, which can inform best practices even when not using specific hardware.

10. Objectives

- Develop a Multimodal Deep Learning Framework: Design and implement a novel multimodal deep learning framework that integrates EEG signals with spatiotemporal graph neural networks and improved deep belief networks for decoding motor imagery tasks.

- Enhance Decoding Accuracy: Improve the accuracy and reliability of motor imagery decoding from EEG signals by leveraging the spatial and temporal correlations within the data using advanced neural network architectures and multimodal integration techniques.

- Investigate Feature Learning and Representation: Explore the capabilities of deep belief networks for learning hierarchical representations of EEG features and investigate how these learned representations can enhance the discrimination of motor imagery tasks.

- Evaluate Generalization and Transfer Learning: Assess the generalization performance of the proposed framework across different motor imagery tasks, subjects, and experimental conditions, and investigate the potential for transfer learning to adapt the model to new datasets or tasks.

- Validate Against Existing Methods: Benchmark the proposed framework against state-of-the-art methods for motor imagery decoding, including traditional machine learning approaches and other deep learning architectures, to demonstrate its effectiveness and superiority in terms of decoding accuracy and robustness.

11. Methodology

1. Data Collection and Pre-processing

- Collect and preprocess multichannel EEG data to remove noise and artifacts.
- Segment the data into epochs and label them according to motor imagery tasks.

2. Multiscale-Feature Extraction

- Spatial Neural Network Design: Implement residual connections, dilated convolutions, and multi-resolution pathways to capture spatial features at different scales.
- Temporal Neural Network Design: Use LSTM layers with skip connections, attention mechanisms, and residual LSTM blocks to extract robust temporal features.
- Multi-Scale Fusion: Fuse features across scales using feature pyramid fusion, attentional fusion, and graph-based techniques for comprehensive representation.
- Regularization and Optimization: Apply gradient clipping, dropout, batch normalization, and learning rate scheduling to stabilize training and improve generalization.

3. Temporal Attention Mechanisms

- Integrate attention mechanisms to focus on important temporal segments and enhance the model's discriminative power.
- Combine LSTM layers with graph convolutional networks (GCN) to capture both temporal and spatial dependencies, fusing their outputs for better classification.

4. Training and Optimization

- Split data into training, validation, and testing sets.
- Train the model using optimization techniques, tune hyperparameters, and regularize to prevent overfitting.
- Use Grad-CAM to visualize important EEG regions and evaluate the model's performance on various metrics.

12. Project Outcomes and Deliverables

- The classification framework will be capable of accurately identifying and classifying different motor imagery tasks from EEG signals.
- A software design or prototype implementing the classification framework will be produced, providing a user-friendly interface.
- The project will involve the development of new algorithms, techniques, and methodology for processing EEG signals. These algorithms will be novel adaptations and extensions of existing methods to increase accuracy, efficiency, and interpretability.

13. Novelty of Work

1. Novel algorithm for channel selection

- Channel selection will be performed using both filter and wrapper methods.
- In filter method, features are selected based on statistics measures.
- In wrapper method, selection of features is done by considering it as a search problem, in which different combinations are made, evaluated, and compared with other combinations.
- Some techniques utilized are forward selection, backward selection, exhaustive feature selection and recursive feature elimination.

2. Pipeline architecture for features reduction

- Autoencoders learn the compressed representation, extract the underlying structure, and eliminate redundant or noisy information.
- Lower order embedding methods to be employed.
- Manifold learning has been proved to be effective when the data is non-linear which cannot be captured by linear approaches like principal component analysis. [9]

3. Customized deep belief networks/variants

- Deep Belief Networks are made up of numerous layers of Restricted Boltzmann Machines (RBMs), allowing for a hierarchical representation of features.
- According to previous studies, it has shown to be successful at capturing complex linkages and dependencies in EEG data for discriminative feature representations.
- Hence, DBN is well suited due to its adaptability to Non-Stationary Signals.
- In case of any problems, the networks will be further customized and new variants will be developed. [10]

4. Train model on multiple datasets

- Multiple datasets with various EEG electrodes (22, 59, and 118 electrodes) will be utilized for training. A multiclass data will be used. [11][12]

5. Website/App design

- A user friendly and easy to access web application will be deployed.

2.1 Literature Survey

2.1.1 Theory Associated With Problem Area

- Motor Imagery and EEG Signals: Motor imagery (MI) refers to the cognitive process of imagining a movement without actual execution. It is a common practice in neurorehabilitation and brain-computer interfaces (BCIs) due to its potential to reflect brain activity associated with specific motor tasks. EEG signals, which record electrical activity in the brain, are particularly valuable in detecting motor imagery as they provide real-time, non-invasive data on brain activity.
- Graph-Based Neural Networks for EEG Analysis: Graph Neural Networks (GNNs) have emerged as a powerful tool for analyzing EEG data by representing the brain's spatial and temporal characteristics as graphs. Here's a breakdown of the key theories and methods associated with this approach:
- Graph Convolutional Networks (GCNs): GCNs are designed to process data represented as graphs. For EEG analysis, GCNs can model the spatial dependencies between different brain regions by treating each electrode as a node and the connections between them as edges. The Multi-Scale Adaptive Spatial-Temporal Graph Convolutional Network (MAST-GCN), for instance, leverages this by integrating various convolutional layers to capture diverse temporal and spatial features, improving the classification of motor imagery tasks.
- Adaptive Spatiotemporal Graph Convolutional Networks (ASTGCN): ASTGCN adaptively assesses node importance and extracts time-domain features to enhance classification performance. This method is particularly effective for motor imagery classification as it adapts to the dynamic nature of EEG data and improves accuracy by better capturing the temporal evolution of brain activity.
- Dynamic and Multivariate Analysis: Dynamic Causal Analysis: Techniques like Multivariate Nonparametric Dynamical Granger Causality (mndGC) explore causality and interactions between different brain regions. This method helps in understanding the directed network changes in EEG signals, crucial for distinguishing between different motor imagery states.

- Channel-Weighted and Feature Fusion Networks: The Point-wise Dynamic Multi-Graph Convolution Network (dMGCN) and Channel-Weighted Transformer Feature Fusion Network (CWTFFNet) integrate multiple graph convolutions and feature extraction mechanisms to improve seizure prediction and motor imagery classification. These frameworks focus on optimizing the extraction of relevant features and balancing sensitivity and false prediction rates.
- Spatio-Temporal and Edge-Aware Methods: Spatio-Temporal Graph Convolutional Networks (ST-GCNs): These networks classify EEG signals by treating them as frames of a graph, thus capturing the complex relationships among different channels over time. The Edge-aware Spatio-Temporal Graph Convolutional Network further enhances this by considering the importance of different edges, providing a more nuanced understanding of brain connectivity during motor imagery tasks.
- Performance Metrics and Evaluation: The performance of these models is often evaluated using metrics like accuracy and the Area Under the Curve (AUC). For example, models like ASTGCN and ST-GCN have demonstrated superior performance with higher accuracy and AUC scores compared to traditional methods. This performance indicates their effectiveness in capturing and classifying complex EEG patterns associated with motor imagery.

2.1.2 Existing Systems and Solutions

Various systems and methodologies have been developed to classify and analyze EEG signals, particularly in the context of motor imagery. These approaches range from traditional signal processing techniques to cutting-edge deep learning frameworks. Below is a general overview of these systems based on recent advancements:

Graph-Based Convolutional Networks for EEG Analysis

- **Tools and Technology**: These systems typically use advanced EEG equipment, deep learning frameworks like PyTorch, and powerful hardware including GPUs. The frameworks often incorporate graph convolutional layers, adaptive spatial-temporal processing, and multi-scale time convolutional layers to handle the complexity of EEG data.

- **Findings**: Such frameworks significantly outperform traditional models by integrating dynamic topological information, capturing diverse time-domain features, and enabling spatial-temporal integration. They are particularly effective in enhancing accuracy, F-Score, and sensitivity metrics, with a strong focus on channels with high EEG signal correlation.

Adaptive Spatiotemporal Networks for Motor Imagery Classification

- **Tools and Technology**: These systems employ advanced hardware and software, including systems with numerous scalp electrodes and deep learning libraries. Techniques like adaptive node importance assessment and time-domain feature extraction are commonly used.
- **Findings**: These models achieve high accuracy rates, often outperforming traditional neural networks. They utilize cross-validation techniques to ensure robustness and display low accuracy deviation, highlighting their stability and reliability in motor imagery classification.

Feature Extraction and Fusion Networks

- **Tools and Technology**: Utilization of comprehensive EEG datasets, coupled with multi-branch feature extractors and dynamic graph convolution networks, is common. These systems often incorporate feature fusion networks to enhance the accuracy of classification tasks.
- **Findings**: These frameworks achieve high accuracy and sensitivity, particularly in challenging tasks like seizure prediction. They optimize post-processing parameters to balance between high sensitivity and low false prediction rates, improving overall performance in EEG analysis.

Dynamic Causal Analysis in EEG Data

- **Tools and Technology**: Advanced causal analysis tools are employed to understand the directional flow of information in the brain during motor imagery. Techniques include dynamic causal models that are nonparametric and noise-resistant.
- **Findings**: These techniques excel in capturing instantaneous changes in brain networks, offering deeper insights into the different stages of motor imagery. They extend the exploration of brain connectivity and causality, contributing to more precise analysis and classification.

Spatio-Temporal Analysis of EEG Data

- **Tools and Technology**: These systems leverage spatio-temporal graph convolutional networks and EEG datasets from large repositories. The focus is on uncovering latent graph structures within the EEG data that represent functional brain connectivity.
- **Findings**: These models show significant improvements in classification accuracy compared to conventional convolutional networks. They provide valuable insights into the brain's functional connectivity, aiding in the understanding of motor imagery tasks.

Analysis of Existing Solutions The existing systems for EEG-based motor imagery classification have shown significant advancements in terms of performance and accuracy, primarily due to the application of advanced neural network models and dynamic analysis techniques. Key contributions include:

- **Advanced Neural Networks**: Leveraging graph-based neural networks enables the capture of complex spatial-temporal relationships in EEG data, leading to enhanced classification accuracy and robustness.
- **Dynamic Causal Analysis**: Techniques focusing on dynamic causality have deepened the understanding of directed network changes in the brain, offering insights into motor imagery and brain connectivity.
- **Feature Fusion and Optimization**: Advanced frameworks have optimized feature extraction and post-processing, leading to improved prediction accuracy and minimized false prediction rates.

2.1.3 Research Findings for Existing Literature

TABLE 3: Research Findings

S. No.	Roll Number	Name	Paper Title	Tools/ Technology	Findings	Citation
1	102103496	Selina Varshney	MAST-GCN: Multi-Scale Adaptive Spatial-Temporal Graph Convolutional Network for EEG-Based Depression Recognition	Tools: 1.EEG equipment, PyTorch, RTX 3080 Ti GPU 2.SGD optimizer. Frameworks: 1.MAST-GCN, Adaptive Graph Convolution (AGC) 2.Multi-Scale Time Convolutional Layer (MS-TCL) 3.3D Graph Convolution (G3D). Datasets: Two datasets, one from MPHCE and another from Shenzhen People’s Hospital have been used.	1.The study highlights MAST-GCN framework, using Adaptive Graph Convolution for dynamic topologies, Multi-Scale Time Convolution for diverse time-domain features, and 3D Graph Convolution for integrating spatial-temporal information, outperforms RNN, CNN, and other traditional techniques achieving 80.13% accuracy , 76.48% F-Score, and 77.94% sensitivity with a good balance. 2.It was also recognized that High-correlation EEG channels in the prefrontal and frontal lobes, along the right hemisphere, are vital for depression recognition.	L. Haifeng, Z. You, Y. Guo, X. Hu. "MAST-GCN: Multi-Scale Adaptive Spatial-Temporal Graph Convolutional Network for EEG-Based Depression Recognition.", in IEEE Transactions on Affective Computing, 2024.

2	102103484	Vanshika Mittal	Adaptive Spatiotemporal Graph Convolutional Networks for Motor Imagery Classification	Tools: 1.Neuroscan system with 64 Ag/AgCl scalp electrodes, Hardware including 32G RAM, 3.60 GHz Intel Core I7 CPU, NVIDIA GPU (Titan Xp), and Software including PyTorch and deep learning libraries. Frameworks: 1.Graph Neural Network (GNN) 2.Adaptive Graph Convolutional Layer (AGCL), 3.Convolutional Neural Network (CNN). Dataset: EEG signals from twenty-five healthy subjects performing motor imagery tasks.	1.The study proposes an Adaptive Spatiotemporal Graph Convolutional Network (ASTGCN), with 10-fold cross validation for MI-EEG classification, which unlike traditional GCNs, adaptively assesses node importance and extracts time-domain features to enhance classification performance. 2.ASTGCN achieves an average accuracy of 90.6%, outperforming CNN-SAE at 74.9% and EEGNet at 84.9%. ASTGCN shows a 30.0% improvement over CNN-SAE and a 6.7% improvement over EEGNet. 3.Additionally, ASTGCN's accuracy standard deviation is 3.4, which is lower than CNN-SAE's 5.7 and EEGNet's 5.0, highlighting its robustness.	S. Biao, H. Zhang, Z. Wu, Y. Zhang, T. Li. "Adaptive spatiotemporal graph convolutional networks for motor imagery classification." in IEEE Signal Processing Letters, 2021, pp.219-223.

3	102103510	Vanshika Narang	Dynamic Multi-Graph Convolution-Based Channel-Weighted Transformer Feature Fusion Network for Epileptic Seizure Prediction	Tools & Technology: 1.Deep Learning Framework: MB-dMGC-CWTFFNet 2.Multi-Branch Feature Extractor 3.Point-wise Dynamic Multi-Graph Convolution Network (dMGCN) 4.Channel-Weighted Transformer Feature Fusion Network (CWTFFNet) 5.GitHub repository for code availability Datasets: 1.CHB-MIT Scalp EEG Dataset 2.Xuanwu Intracranial EEG Dataset	1.The study proposes the MB-dMGC-CWTFFNet, a deep learning framework designed to predict epileptic seizures using EEG data. 2.The framework achieves outstanding accuracies on two datasets: an AUC of 0.935 on the CHB-MIT dataset and 0.984 on the Xuanwu dataset, with corresponding sensitivities of 97.8% and 100%, and low false prediction rates per hour of 0.059 and 0.079, respectively. These results indicate a substantial improvement over existing methods. 3. The study discusses the optimization of post-processing parameters, such as the moving average filter length and threshold, to achieve a balance between sensitivity and false prediction rate.	Wang, Y., Cui, W., Yu, T., Li, X., Liao, X., & Li, Y. (2023). Dynamic multi-graph convolution based channel-weighted transformer feature fusion network for epileptic seizure prediction. IEEE Transactions on Neural Systems and Rehabilitation Engineering.

4	102103237	Aarushi Kamboj	Classify EEG and Reveal Latent Graph Structure with Spatio-Temporal Graph Convolutional Neural Network	1.EEG 2.Spatio-Temporal Graph Convolutional Neural Network (ST-GCN) 3.Graph Convolutional Neural Network (GCN) 4.Edge-aware Spatio-Temporal Graph Convolutional Network 5.Information Aggregation based Method 6.Complete Graph and Learnable Weight Matrix 7.Activation Function (ReLU) 8.Batch Normalization 9.Web-based Interactive Tool for Graph Visualization Datasets: EEG Motor Movement/Imagery Dataset from PhysioNet	1. The study proposes a Spatio-Temporal Graph Convolutional Neural Network (ST-GCN) for classifying EEG signals, treating them as frames of a graph to consider the complex relationships among channels 2. The ST-GCN model achieves an AUC of 0.8378, outperforming baseline models like Deep ConvNet (AUC of 0.8071) and Shallow ConvNet (AUC of 0.7967). The model also reveals a meaningful latent graph structure of EEG data, confirmed by neuroscientists, providing insights into the brain's functional connectivity during imagined hand movements.	Li, X., Qian, B., Wei, J., Li, A., Liu, X., & Zheng, Q. (2019, November). Classify EEG and reveal latent graph structure with spatio-temporal graph convolutional neural network. In 2019 IEEE International Conference on Data Mining (ICDM) (pp. 389-398). IEEE.

5	102103493	Ishita Suchdeva	Constructing Time-Varying Directed EEG Network by Multivariate Nonparametric Dynamical Granger Causality	1.Multivariate Nonparametric Dynamical Granger Causality (mndGC) 2.Adaptive Directed Transfer Function (ADTF)	1. The study proposes a Spatio-Temporal Graph mndGC outperforms ADTF in noise resistance and capturing instantaneous directed network changes in EEG data. 2.mndGC reveals distinguishable network characteristics between left- and right-hand motor imagery (MI) during different MI stages. 3.The study extends nonparametric causality exploration and provides practical suggestions for time-varying directed EEG network analysis.	C. Yi, Y. Qiu, W. Chen, C. Chen, Y. Wang, P. Li, P. Xu, X. Zhang, L. Jiang, D. Yao, F. Li, L. Yang. "Constructing Time-Varying Directed EEG Network by Multivariate Nonparametric Dynamical Granger Causality." IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 30, 2022.

2.1.4 Problem Identified

1. Limited Generalization Across Diverse EEG Datasets

- **Issue:** Most studies, such as those using MAST-GCN and MB-dMGC-CWTFFNet, focus on specific datasets (e.g., datasets from MPHCE, Shenzhen People's Hospital, CHB-MIT, and Xuanwu). This can limit the generalizability of the findings to other EEG datasets or different subject groups.
- **Impact:** The lack of cross-dataset validation raises concerns about the robustness and applicability of the models in real-world scenarios where EEG data may vary significantly across different populations and recording conditions.

2. High Computational Complexity and Resource Requirements

- **Issue:** Techniques like Adaptive Spatiotemporal Graph Convolutional Networks (ASTGCN) and MB-dMGC-CWTFFNet rely on complex deep learning architectures that require substantial computational resources, such as high-end GPUs and large memory capacities.
- **Impact:** The high computational cost can be a barrier to deploying these models in resource-constrained environments, limiting their practical application in real-time or portable EEG systems.

3. Difficulty in Interpreting Model Decisions

- **Issue:** Advanced models such as MAST-GCN and ST-GCN, while highly accurate, often act as "black boxes," making it difficult to interpret the decisions made by these models, especially when they involve complex graph structures and multi-scale convolutional layers.
- **Impact:** The lack of interpretability can hinder the acceptance and trust of these models in clinical settings, where understanding the underlying decision-making process is crucial for diagnosis and treatment planning.

4. Limited Exploration of Temporal Dynamics

- **Issue:** Although models like mndGC focus on capturing time-varying directed networks, there is still a need for more sophisticated approaches to fully understand and leverage the temporal dynamics of EEG data in motor imagery tasks.

- **Impact:** Inadequate modeling of temporal aspects may lead to suboptimal performance in tasks that heavily depend on time-related features, such as continuous motor imagery classification or real-time seizure prediction.

5. Balancing Sensitivity and False Prediction Rates

- **Issue:** In the context of seizure prediction, frameworks like MB-dMGC-CWTFFNet face challenges in optimizing the trade-off between sensitivity and false prediction rates. While high sensitivity is crucial, a high false prediction rate can lead to unnecessary alarms and stress for patients.
- **Impact:** The inability to maintain a low false prediction rate without compromising sensitivity limits the practical utility of these models in clinical settings, where false positives can have significant consequences.

6. Dependence on Specific EEG Channels and Spatial Configurations

- **Issue:** Several models, including MAST-GCN and ST-GCN, emphasize the importance of specific EEG channels, particularly in the prefrontal and frontal lobes. However, the reliance on these specific channels may not generalize well across different tasks or subjects, leading to variability in performance.
- **Impact:** This dependence on particular spatial configurations may reduce the flexibility of the models, making them less effective in scenarios where the optimal channel configuration is unknown or varies across subjects.

7. Challenges in Integrating Multi-Modal Data

- **Issue:** While the current literature focuses primarily on EEG data, there is a growing need to integrate multi-modal data (e.g., EEG, EMG, and fMRI) to improve the accuracy and robustness of motor imagery classification models.
- **Impact:** The lack of multi-modal integration limits the potential of these models to capture a more comprehensive understanding of brain activity, which could enhance performance in complex tasks requiring higher levels of cognitive and motor function analysis.

2.1.5 Survey of Tools and Technologies Used

1. EEG Equipment and Data Acquisition Tools

EEG Equipment:

- Neuroscan System with Ag/AgCl Scalp Electrodes: Used in studies for high-resolution EEG data collection. This equipment is crucial for capturing detailed brain activity, particularly during motor imagery tasks.
- Other EEG Equipment: Specific details about other EEG acquisition setups are mentioned, particularly in studies like MAST-GCN, where different datasets from medical institutions were utilized.

2. Computing Hardware

High-Performance GPUs:

- NVIDIA GPU (Titan Xp, RTX 3080 Ti): These high-end GPUs are critical for training deep learning models like MAST-GCN, ASTGCN, and MB-dMGC-CWTFFNet. They provide the necessary computational power to handle large datasets and complex neural networks.
- General Computing Hardware:
- 32GB RAM and 3.60 GHz Intel Core i7 CPU: This setup is commonly used alongside GPUs to ensure smooth processing and model training without memory bottlenecks.

3. Deep Learning Frameworks and Libraries

Frameworks:

- PyTorch: A versatile deep learning framework used in various studies, including those employing MAST-GCN and ASTGCN. PyTorch is favored for its flexibility, dynamic computation graphs, and strong support for GPU acceleration.
- Other Frameworks: Specific to certain models, such as the MB-dMGC-CWTFFNet, which utilizes custom neural network layers like Multi-Branch Feature Extractors and Point-wise Dynamic Multi-Graph Convolution Networks.

Libraries:

- SciPy: Used for signal processing tasks such as noise reduction and filtering in studies that involve preprocessing EEG data before feeding it into neural networks.
- MNE-Python: Specialized in EEG data processing and visualization, allowing researchers to handle complex EEG datasets efficiently.

4. Specialized Neural Network Architectures

Graph Neural Networks (GNNs):

- MAST-GCN (Multi-Scale Adaptive Spatial-Temporal Graph Convolutional Network): Designed to handle the spatial and temporal dynamics of EEG data by leveraging graph-based representations.
- ASTGCN (Adaptive Spatiotemporal Graph Convolutional Network): Focuses on adaptively assessing node importance and time-domain feature extraction for motor imagery classification.
- Multi-Branch and Transformer-Based Networks: MB-dMGC-CWTFFNet: A sophisticated deep learning framework that combines multi-branch feature extraction with transformer-based fusion for enhanced prediction accuracy, particularly in seizure prediction.

5. Signal Processing Techniques

- Adaptive Graph Convolution (AGC): Employed in MAST-GCN for dynamically adjusting the topology of graphs representing EEG data. This helps in better capturing the evolving spatial relationships in the brain's electrical activity.
- Smoothing and Filtering Techniques:
    1. Savitzky-Golay Filter: Applied for smoothing EEG signals to remove noise while preserving important features. This is particularly important in preprocessing steps to improve model accuracy.
    2. Baseline Removal and Normalization: Essential preprocessing steps to standardize the EEG data, making it suitable for feeding into neural networks.

6. Datasets

EEG Datasets:

- CHB-MIT Scalp EEG Dataset: Widely used for epileptic seizure prediction, offering a comprehensive set of EEG recordings from various patients.
- Xuanwu Intracranial EEG Dataset: Another important dataset used for seizure prediction, providing high-quality intracranial EEG data.
- Motor Imagery Datasets (e.g., from PhysioNet): Commonly used for motor imagery classification tasks, providing data that is critical for developing and testing EEG-based BCI models.

7. Optimization Techniques

Optimizers:

- SGD (Stochastic Gradient Descent) Optimizer: Utilized in several studies for optimizing neural network training, known for its simplicity and effectiveness in large-scale learning tasks.

Post-Processing Optimization:

- Moving Average Filter Length and Threshold: Specific to seizure prediction models like MB-dMGC-CWTFFNet, where fine-tuning these parameters helps balance sensitivity and false prediction rates.

2.2 Software Requirement Specification

2.2.1 Introduction

2.2.1.1 Purpose

The purpose of this project is to advance assistive technology and neurorehabilitation by developing a robust system for decoding motor imagery from EEG signals.
With the increasing prevalence of neurological conditions and motor disabilities due to aging, accidents, or diseases, there is a growing need for innovative solutions to support affected individuals. This project addresses this need by providing a means to translate mental commands into actionable inputs, enabling users to control devices or express themselves more intuitively, thereby improving their quality of life.

2.2.1.1 Intended Audience and Reading Suggestions

1. **Researchers and Academics in Neuroscience and Biomedical Engineering**:
   Individuals focused on brain-computer interfaces (BCIs), neurorehabilitation, and EEG signal processing, can refer to Chapter 1.10 Novelty of Work, Chapter 2(Requirement Analysis), Chapter 3(Methodology) and Chapter 4 (Diagrams) for better understanding.

2. **Paralyzed and Older Individuals, and Their Caregivers**:
   Individuals living with paralysis or age-related motor impairments, and their caregivers, who are interested in understanding how emerging technologies can enhance mobility and communication can refer to Chapter 1 (Introduction) for an overview, and Chapter 5 (Conclusions and Future Scope) to understand future developments.

3. **Engineers and Developers in Assistive Technology**:
   Professionals developing assistive devices, prosthetics, or neurotechnology, who are interested in applying advanced neural network techniques to improve the accuracy and usability of these technologies, can review Chapter 4 (Design Specifications) for diagrams and technical details on system architecture and design. Chapter 2 (Requirement Analysis) provides an overview of tools, technologies, and performance requirements for implementation.

4. **Clinicians and Rehabilitation Specialists**:
   Medical professionals, particularly neurologists and physiotherapists, who treat patients with motor impairments and are interested in integrating BCIs into therapeutic practices, can review Chapter 1 (Introduction) and Chapter 5 (Conclusions and Future Scope).

5. **Policymakers and Stakeholders in Healthcare Technology**:
   Decision-makers involved in funding, developing, and regulating healthcare technologies, especially those that impact the lives of paralyzed and older individuals, can get an overview from Chapter 1 (Introduction) while Chapter 5 (Conclusions and Future Scope) offers insights into the broader implications for healthcare policy and technology development.

2.2.1.3 Project Scope

The proposed technology has vast potential beyond assistive technology. It can revolutionize consumer electronics interfaces, enabling control of smart devices and computers through mental commands. BCI could enhance gaming and virtual reality experiences, neurofeedback therapy and brain-computer music interfaces offer additional avenues for innovation. These advances could revolutionize how we interact with computers and boost personal well-being with thought-controlled interfaces and therapies.

2.2.2 Overall Description

2.2.2.1 Product Perspective

The current product is a python based , machine learning model designed to process pre-labeled, pre-processed EEG data. This data undergoes a 5-fold cross-validation process to ensure model robustness. Spatial graphs are constructed using methods like Residual Connections, Dilated Convolutions, and Multiresolution Pathways, while temporal graphs are developed through LSTM layers and attention mechanisms. These spatial and temporal graphs are fused into a single model, optimized to accurately predict the movement label based on the given EEG input.
Following successful validation and optimization, the product can be developed into a user-friendly website in the future, making it accessible and easier to use for a broader audience.

2.2.2.2 Product Features

1. **Multichannel EEG Data Handling**:

- Ability to process multichannel EEG data, including 22, 59, 118 channels.
- Pre-processing tools to remove noise, artifacts, and baseline drift, with segmentation into epochs corresponding to motor imagery tasks.

2. **Advanced Model Architecture**:

- Integration of stacked LSTM layers and graph convolutional networks (GCNs) to capture temporal dependencies and spatial relationships, respectively.
- Fusion of outputs from LSTM and GCN layers for a holistic representation of EEG data.

3. **User-Centric Design**:

- Focused on ease of use, the product minimizes the physical demands on the user, allowing those with severe disabilities to operate assistive devices with minimal effort.

4. **Interpretability with Grad-CAM**:

- Application of Grad-CAM to visualize important EEG channels and temporal segments contributing to model predictions, enhancing interpretability.

5. **Performance Evaluation**:

- Comprehensive evaluation using metrics such as accuracy, precision, recall, and F1 score.

2.2.3 External Interface Requirements

2.2.3.1 User Interfaces

The interface will include the following components:

- Dashboard: The main dashboard will provide an overview of the data processing pipeline, displaying the current status of data collection, preprocessing, and model training. Users will have access to visualizations that show the progress and performance metrics of the model in real-time.
- Data Input Forms: Users will be able to upload EEG data through a straightforward form that allows for easy selection and labeling of datasets. The form will support multiple data formats commonly used in EEG studies.
- Interactive Visualizations: Once the model has been trained, users can interact with visualizations of the model's performance. This includes graphical representations of the Grad-CAM outputs, highlighting the important EEG channels and temporal segments for each motor imagery task.
- Settings and Customization: The interface will provide options for users to customize the parameters of the neural networks, such as the number of LSTM layers, types of convolutional pathways, and attention mechanisms. These settings will be accessible through an intuitive menu.
- Accessibility: The interface will be designed to be accessible to users with varying levels of technical expertise, with clear instructions and tooltips provided throughout the platform.

2.2.3.2 Hardware Interfaces

This system is designed to interface with a variety of EEG hardware devices. The following hardware interfaces are essential:

- EEG Headsets: The system will support a wide range of EEG headsets used in motor imagery research. It will be compatible with devices that transmit data via USB, Bluetooth, or other wireless protocols.
- Connection Protocols: The system will establish connections with EEG devices using standard communication protocols such as USB for wired connections and Bluetooth for wireless connections. The setup process will include a step-by-step guide to ensure smooth integration with the hardware.
- Performance Requirements: To ensure accurate data collection and real-time processing, the system will require high data transfer rates, particularly for wireless connections. The interface will include diagnostic tools to assess and optimize the connection quality between the hardware and the software.
- Device Compatibility: The system will be tested for compatibility with various models of EEG devices, ensuring that it can handle differences in sampling rates, channel configurations, and other device-specific features.

2.2.3.3 Software Interfaces

The software interface is designed to be highly adaptable, allowing it to integrate seamlessly with various external software systems and tools used in EEG analysis. Key aspects include:

- Operating System Compatibility: The system will be compatible with major operating systems, including Windows, macOS, and Linux. This ensures that users can deploy the software on a variety of platforms according to their preferences and needs.
- APIs and SDKs: The system will provide APIs to allow for the integration of additional data processing or machine learning tools. These APIs will be well-documented, enabling users to extend the system's functionality or connect it with other research tools.
- Communication Protocols: The software will use standard communication protocols such as HTTP and TCP/IP for interactions with external databases or cloud services. This will allow for the secure transfer of data between the local system and remote servers, facilitating distributed processing if needed.
- Third-Party Software Integration: The system will be designed to integrate with third-party software commonly used in EEG research, such as MATLAB or Python-based data analysis tools. Users will be able to export data or results in formats that are compatible with these tools, ensuring flexibility in how they use and analyze the data.
- Error Handling: Robust error handling mechanisms will be implemented to manage any issues that arise during communication with external software or hardware. This will include logging errors, providing clear error messages to the user, and offering troubleshooting steps to resolve common issues.

2.2.4 Non-functional Requirements

2.2.4.1 Performance Requirements

- Real-time Processing: The system should be capable of processing EEG data in real-time, ensuring that motor imagery classification and response generation occur within a minimal delay (preferably under 100 milliseconds).
- Scalability: The framework should be scalable to handle increasing data volumes from multiple EEG channels and users without significant degradation in performance.
- Accuracy: The classification accuracy should consistently exceed existing benchmarks, with a target of achieving over 90% accuracy in motor imagery tasks across diverse datasets.
- Resource Efficiency: The system should optimize resource usage, minimizing CPU and memory consumption, especially when deployed on portable devices like smartphones or tablets.
- Throughput: The system must support high throughput, enabling the processing of large batches of EEG data for offline analysis without performance bottlenecks.

2.2.4.2 Safety Requirements

- User Data Protection: The system must ensure that the EEG data collected from users is handled in a way that prevents misuse or unauthorized access, protecting user privacy.
- System Reliability: The system should be reliable, with minimal downtime, especially when used in critical applications such as neurorehabilitation or assistive technology.
- Fault Tolerance: The framework should be designed to handle potential faults or errors gracefully, ensuring that any malfunction does not result in data loss or incorrect classifications.
- Compliance: The system should comply with relevant medical device regulations and standards, ensuring that it is safe for use in clinical settings and does not cause harm to users.

2.2.4.2 Security Requirements

- Data Encryption: All EEG data, both at rest and in transit, must be encrypted using industry-standard encryption protocols to prevent unauthorized access and ensure data integrity.
- User Authentication: The system should implement strong user authentication mechanisms, ensuring that only authorized individuals can access or modify sensitive data.
- Access Control: Fine-grained access control policies should be in place to ensure that different users (e.g., patients, clinicians, researchers) have access only to the data and system functionalities that are relevant to their roles.
- Regular Security Audits: The system should undergo regular security audits and vulnerability assessments to identify and address potential security threats.
- Incident Response: There should be a defined incident response plan in place to quickly address and mitigate any security breaches or data leaks.

2.3 Risk Analysis

Developing an EEG-based motor imagery classification system involves several risks:

Technical Risks

- EEG Signal Quality: Noise and artifacts may degrade signal quality, affecting model accuracy.
- Model Complexity: The use of GNN, BiLSTM, and DBAN increases the risk of overfitting, especially with limited data.
- Resource Limitations: Training complex models may strain computational resources, leading to delays.
- Hyperparameter Tuning: Finding optimal hyperparameters is challenging, possibly resulting in suboptimal performance.

Operational Risks

- Subject Variability: EEG signal differences across subjects may lead to inconsistent results.
- Misinterpretation: Users might misunderstand model outputs, leading to incorrect decisions.
- Generalization Issues: The model may not perform well on datasets outside the competition due to differing conditions.

Evaluation and Deployment Risks

- Evaluation Metrics: Inadequate metrics might misrepresent model performance.
- Deployment Challenges: Real-world deployment may face issues like latency, integration difficulties, or real-time processing demands.

System Security Risks

- Unauthorized Access: The system could be vulnerable to unauthorized access or tampering.
- Model Integrity: There is a risk of tampering with the model or its outputs, leading to incorrect classifications.

3.1 Investigative Techniques

TABLE 4: Investigative Techniques

S.No	Investigative Techniques	Description	Investigative Projects Examples
1	Graph Neural Networks (GNNs)	GNNs are ideal for modelling EEG data, which can be naturally represented as graphs with electrodes as nodes and connections between them as edges. This spatial representation is crucial for understanding complex brain interactions during motor imagery tasks. Spatial Dependency Modelling: GNNs capture spatial dependencies between EEG channels, learning how different brain regions interact during motor imagery. Manifold Learning: GNNs reduce data complexity by mapping high-dimensional EEG data to lower-dimensional spaces while preserving intrinsic structures, making computation more efficient. Application: GNNs in this project model spatial relationships between EEG channels, enabling accurate classification of motor imagery tasks by identifying patterns across the brain's topology.	Developing models to classify motor imagery tasks by leveraging the spatial relationships between different EEG channels. - Research projects focusing on the application of graph-based manifold learning techniques in brain-computer interface (BCI) systems. - Systematic investigations into the spatial patterns of EEG signals during cognitive tasks using GNNs.
2	Bi-Directional Long Short-Term Memory (BI-LSTM)	EEG signals are time-series data with significant temporal dependencies, requiring models that can capture long-range patterns over time. BI-LSTMs process information in both forward and backward directions, ensuring full context understanding of EEG signals. Temporal Dynamics: BI-LSTMs excel in capturing the sequential nature of EEG data,	Designing temporal models that analyze the sequence of brain activities during motor imagery tasks. - Implementing BI-LSTM networks for real-time classification of EEG signals in BCI systems. - Comparative studies of LSTM and BI-LSTM networks for improving the temporal analysis of EEG data.

		where the order and timing of neural signals are essential for accurate motor imagery classification. Memory and Sequence Learning: BI-LSTMs' memory cells retain information across time steps, crucial for recognizing prolonged patterns in EEG signals, enhancing the model's ability to classify motor imagery tasks. Application: BI-LSTMs are used to process the temporal aspects of EEG signals, allowing the model to utilize both past and future data for better classification accuracy of motor imagery tasks.	
3	Deep Belief Attention Networks (DBANs)	EEG data often contains a mix of relevant and irrelevant information, necessitating models that can focus on the most critical features. DBANs integrate deep belief networks with attention mechanisms to prioritize important data features. Feature Importance: The attention mechanism in DBANs dynamically weighs the importance of different features in EEG data, ensuring the model focuses on signals most indicative of motor imagery tasks. Noise Reduction: By concentrating on relevant features, DBANs reduce the impact of noise, a common challenge in EEG analysis, leading to more accurate classification results. Application: DBANs are utilized in this project to enhance feature selection from EEG data, with the attention mechanism focusing on the most critical aspects, improving classification performance.	Development of DBAN-based models that prioritize important features in noisy EEG data. - Research on the effectiveness of attention mechanisms in improving the accuracy of motor imagery classification. - Projects investigating the role of deep belief networks in feature extraction for EEG data analysis.

4	Integration of Techniques	The complexity of EEG data requires a multi-faceted approach. By integrating GNNs, BI-LSTM, and DBANs, this project addresses different data aspects—spatial, temporal, and feature-specific—to improve classification accuracy and model efficiency. Spatial and Temporal Fusion: The combination of GNNs and BI-LSTM allows the model to learn both where and when neural activities occur, critical for understanding motor imagery tasks. Feature Selection and Dimensionality Reduction: DBANs ensure the model focuses on the most informative parts of EEG data, reducing noise and irrelevant features, and helping in dimensionality reduction without sacrificing accuracy. Application: The integrated approach enhances the model's ability to classify motor imagery tasks accurately by leveraging spatial patterns, temporal sequences, and critical feature importance, making it a powerful tool for BCI systems.	Comprehensive projects that integrate multiple neural network techniques for EEG signal classification. - Studies exploring the synergistic effects of combining spatial, temporal, and feature-specific techniques for better model performance in BCI applications. - Advanced BCI research projects that aim to improve motor imagery classification through the integration of GNNs, BI-LSTM, and DBANs.
5	Dimensionality Reduction Techniques (e.g., PCA, t-SNE)	EEG data is often high-dimensional, making it computationally intensive and prone to overfitting. Dimensionality reduction techniques like Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE) help simplify the data while retaining essential patterns. Feature Extraction: These techniques reduce the number of features in the data by identifying and retaining the most significant ones, making the data more manageable for machine learning models. Visualization: Dimensionality reduction also aids in visualizing the complex patterns in EEG data, providing insights into the data's structure	Application of PCA for reducing EEG data dimensionality before classification tasks. - Comparative studies of dimensionality reduction techniques in improving model efficiency for BCI systems. - Projects that leverage t-SNE for visualizing EEG data patterns and enhancing understanding of motor imagery tasks.

		and relationships between different motor imagery tasks. Application: PCA and t-SNE are used in this project to reduce the dimensionality of EEG data, improving the efficiency and interpretability of the model without losing critical information needed for classification.	
6	Noise Filtering and Data Preprocessing	EEG data is prone to noise from various sources, including muscle activity, eye movements, and external electromagnetic interference. Effective noise filtering and preprocessing are essential for enhancing signal quality before analysis. Noise Reduction Techniques: Techniques such as Savitzky-Golay filtering, baseline correction, and band-pass filtering are employed to remove unwanted noise and artifacts, improving the signal-to-noise ratio. Data Standardization and Normalization: Preprocessing steps also include standardizing and normalizing the data to ensure consistent and comparable results across different EEG recordings. Application: In this project, noise filtering and preprocessing techniques are applied to EEG data to ensure high-quality input for subsequent analysis and classification tasks.	Implementation of advanced noise reduction techniques in EEG signal preprocessing for BCI applications. - Research projects focused on the impact of different preprocessing methods on the accuracy of motor imagery classification. - Development of robust data preprocessing pipelines for handling noisy EEG data in real-time analysis.

3.2 Proposed Solution

Introduction

The proposed solution aims to enhance the classification of motor imagery tasks from EEG signals, a critical component in Brain-Computer Interface (BCI) systems. The solution leverages the power of Graph Neural Networks (GNNs), Bi-Directional Long Short-Term Memory (BI-LSTM), and Deep Belief Attention Networks (DBANs) to address the spatial, temporal, and feature-specific complexities inherent in EEG data. By integrating these advanced machine learning techniques, the solution seeks to improve the accuracy, robustness, and efficiency of motor imagery classification.

Overview of the Solution Architecture

The proposed solution consists of three primary components:

- **Graph-Based Manifold Learning Module (GNN):** This module is responsible for capturing the spatial relationships between EEG channels, enabling the model to learn the intricate topological patterns associated with motor imagery tasks.
- **Temporal Dynamics Module (BI-LSTM):** This module focuses on the temporal aspects of EEG signals, processing the sequential nature of brain activity to capture long-term dependencies that are crucial for accurate classification.
- **Feature Selection and Attention Module (DBAN):** This module enhances the model’s ability to focus on the most relevant features in the data, improving both the interpretability and performance of the classification task.
- The integration of these modules forms a comprehensive framework capable of handling the multi-dimensional nature of EEG data and extracting meaningful patterns for motor imagery classification.

Detailed Description of Each Component

1. Graph-Based Manifold Learning Module (GNN):

- **Functionality:** The GNN module constructs a graph representation of the EEG data, where each node represents an electrode, and the edges represent the connectivity between electrodes based on signal correlations. This graph structure allows the model to analyze the spatial dependencies between different regions of the brain. The GNN processes this graph using a series of convolutional layers that aggregate information from neighboring nodes, effectively capturing the spatial relationships that are critical for motor imagery classification.
- **Advantages:**
- **Spatial Awareness:** The GNN's ability to capture spatial dependencies enhances the model's understanding of how different brain regions interact during motor imagery tasks.
- **Dimensionality Reduction:** The GNN's manifold learning capabilities reduce the complexity of the EEG data, making the subsequent processing steps more efficient.

2. Temporal Dynamics Module (BI-LSTM):

- **Functionality:** The BI-LSTM module processes the sequential nature of EEG signals, capturing both past and future dependencies through its bi-directional architecture. This is crucial for understanding the temporal dynamics of motor imagery tasks, where the order and timing of neural signals play a significant role. The LSTM cells within the BI-LSTM network retain information over time, allowing the model to learn patterns that unfold across multiple time steps.
- **Advantages:**
- **Long-Term Dependencies:** The BI-LSTM's ability to capture long-term dependencies enhances the model's ability to recognize patterns that span over extended periods, which is essential for motor imagery classification.
- **Contextual Understanding:** By processing the EEG signals in both forward and backward directions, the BI-LSTM provides a more comprehensive understanding of the temporal context in which brain activities occur.

3. Feature Selection and Attention Module (DBAN):

- **Functionality:** The DBAN module enhances the model's ability to focus on the most informative features in the EEG data, dynamically adjusting the attention weights based on the importance of each feature. This selective focus helps the model to prioritize relevant information and ignore noise, leading to more accurate and interpretable results. The deep belief network component of DBAN is responsible for extracting high-level features from the EEG data, while the attention mechanism determines which of these features are most relevant for the classification task.
- **Advantages:**
- **Improved Accuracy:** By focusing on the most relevant features, the DBAN module reduces the impact of irrelevant information, leading to more accurate classifications.
- **Noise Reduction:** The attention mechanism helps to filter out noise in the EEG data, improving the robustness of the model's predictions.

4. Integration and Workflow

The integration of the GNN, BI-LSTM, and DBAN modules results in a coherent workflow that efficiently processes the EEG data and produces accurate motor imagery classifications. The workflow proceeds as follows:

- **Preprocessing:** The raw EEG data is first preprocessed to remove artifacts and normalize the signals. This step ensures that the data is in a suitable format for further processing.
- **Graph Construction:** The GNN module constructs a graph representation of the preprocessed EEG data, capturing the spatial relationships between electrodes.
- **Spatial Processing:** The GNN processes the graph through its convolutional layers, producing a lower-dimensional representation of the EEG signals that preserves spatial dependencies.
- **Temporal Processing:** The lower-dimensional EEG data is then passed to the BI-LSTM module, which captures the temporal dynamics of the signals through its bi-directional architecture.
- **Feature Selection and Attention:** The output of the BI-LSTM is fed into the DBAN module, where the most relevant features are selected and weighted according to their importance for the classification task.

- **Classification:** The final output from the DBAN module is passed to a classifier, which produces the motor imagery classification. The classifier could be a simple dense neural network or a more complex model, depending on the requirements.
- **Post-Processing and Output:** The classification results are post-processed to improve interpretability and then presented to the user or used for controlling a BCI device.

5. Expected Outcomes and Benefits

The proposed solution is expected to significantly improve the accuracy and robustness of motor imagery classification from EEG signals. By integrating GNNs, BI-LSTM, and DBANs, the solution addresses the key challenges of spatial, temporal, and feature-specific complexities in EEG data. The expected outcomes and benefits include:

- **Enhanced Classification Accuracy:** The integration of spatial and temporal processing with attention-based feature selection is expected to result in higher classification accuracy compared to traditional approaches.
- **Improved Generalization:** The model's ability to capture the full complexity of EEG data is expected to improve its generalization to new, unseen data, making it more reliable for real-world applications.
- **Reduced Computational Complexity:** The use of manifold learning and attention mechanisms is expected to reduce the computational complexity of the model, making it more efficient without sacrificing performance.
- **Advancement of BCI Technology:** The insights and methodologies developed in this project have the potential to advance the field of BCI technology, contributing to more effective and accessible BCI systems.
- In conclusion, the proposed solution offers a sophisticated and comprehensive approach to motor imagery classification from EEG signals, addressing the unique challenges of this task with advanced machine learning techniques. The integration of GNNs, BI-LSTM, and DBANs provides a powerful framework that enhances classification accuracy, efficiency, and robustness, contributing to the advancement of BCI technology.

3.3 Work Breakdown Structure (WBS)

.

1. Data Collection and Pre-processing

To acquire and prepare EEG data for analysis by removing noise and ensuring it is in a usable format for further processing.

1.1 EEG Data Acquisition

Collect EEG data from the BCI interface with 22, 59, and 118 channels for motor imagery tasks.

1.2 Data Pre-processing

- Baseline Removal: Eliminate baseline drift to enhance the signal quality.
- Filtration: Remove noise and artifacts to clean the EEG signals.
- Savitzky-Golay (Sgolay) Filtering: Smooth the EEG signal without distorting it, improving the quality of the data.
- Standardization: Normalize data to have zero mean and unit variance, making it comparable across subjects.
- Normalization: Scale data to a specific range, enhancing the effectiveness of the model's learning process.

2. Multiscale Feature Extraction

To extract meaningful features from EEG data across different scales, improving the model's ability to classify motor imagery tasks accurately.

2.1 Spatial Neural Network Design

Residual Connections: Facilitate gradient flow, preventing vanishing gradient issues and improving network performance.

Dilated Convolutions: Capture features at multiple spatial scales without increasing the computational load.

Multi-Resolution Pathways: Explicitly extract features at varying spatial resolutions, enhancing the model's spatial understanding.

2.2 Temporal Neural Network Design

- **LSTM with Skip Connections**: Capture long-term dependencies in temporal data while maintaining gradient flow over time.
- **Attention Mechanisms**: Focus the model's attention on the most informative temporal segments, improving classification accuracy.
- **Residual LSTM Blocks**: Improve learning of temporal features by easing gradient flow and stabilizing training.

3. Fusion Techniques

To integrate spatial and temporal features from different scales, enhancing the overall feature representation for better classification.

- **Feature Pyramid Fusion:** Combine features extracted at various spatial and temporal scales to form a comprehensive representation, capturing both fine and coarse details.
- **Attentional Fusion**: Dynamically emphasize task-relevant features while suppressing irrelevant information, increasing the discriminative power of the model.
- **Graph Fusion**: Integrate multi-channel EEG data using Graph Convolutional Networks (GCN) and Graph Attention Networks (GAT), considering the spatial relationships between EEG channels.

4. Regularization and Optimization

To ensure stable training, prevent overfitting, and improve the model's generalization performance.

- **Gradient Clipping**: Prevent exploding gradients during training, especially in deep architectures, ensuring stable learning.
- **Dropout and Batch Normalization**: Regularize the network and stabilize training, reducing overfitting and internal covariate shift.
- **Learning Rate Scheduling**: Adaptively adjust the learning rate during training to promote faster convergence and better generalization.

5. Model Training and Evaluation

To train the model and evaluate its performance using relevant metrics and techniques.

- **Model Training**: Train the model on pre-processed EEG data using optimization techniques like SGD or Adam.
- **Model Evaluation**: Evaluate the model's performance using metrics such as accuracy, precision, recall, and F1 score. Apply Grad-CAM for interpretability and visual insights.

6. Documentation and Reporting

To document the project process and results, and to prepare a final report summarizing the findings and conclusions.

- **Compile Documentation**: Keep detailed records of methods, experiments, and results throughout the project.
- **Final Report Preparation**: Prepare the final report detailing the entire project lifecycle, findings, and conclusions.

3.4 Tools and Technology

1. Data Collection and Preprocessing

1.1 Tools for Data Acquisition

- **EEG Data Sources:** Publicly available EEG datasets (e.g BCI Competition datasets).
- **Data Management Platforms:** Files are stored in a dedicated folder in Excel and CSV formats for data management.

1.2 Tools for Data Cleaning and Transformation

- **MATLAB**: Utilized for data cleaning and transformation with the following processes:
- **Savitzky-Golay Filter**: Applied for smoothing the EEG signals to reduce noise while preserving important signal features.
- **Baseline Correction**: Removed the baseline drift from the EEG data to ensure that the signals reflect true brain activity without baseline fluctuations.
- **Filtering**: Employed various filtering techniques to isolate relevant frequency bands and eliminate unwanted noise.
- **Normalization and Standardization**: Normalized and standardized the data to ensure consistency and comparability across different datasets and subjects.

2. Model Development

2.1 Graph-Based Manifold Learning Module (GNN)

Deep Learning Frameworks:

- TensorFlow and Keras for building and training Graph Neural Networks (GNNs).
- PyTorch for flexible and efficient GNN model development.

Graph Libraries:

- DGL (Deep Graph Library) for implementing and training graph-based neural networks.
- NetworkX for graph construction and manipulation.

2.2 Temporal Dynamics Module (BI-LSTM)

Deep Learning Frameworks:

- TensorFlow and Keras for designing and training bi-directional LSTM networks.
- PyTorch for custom LSTM implementations and experiments.

Sequence Processing Libraries:

- Numpy for handling sequence data.
- SciPy for additional statistical and signal processing functionalities.

2.3 Feature Selection and Attention Module (DBAN)

Deep Learning Frameworks:

- TensorFlow and Keras for constructing Deep Belief Networks (DBNs) and integrating attention mechanisms.
- PyTorch for implementing advanced attention mechanisms and DBNs.

Attention Libraries:

- Transformers Library by Hugging Face for implementing attention-based models if needed.
- TensorFlow Addons for additional attention mechanisms and features.

2.4 Version Control:

Git and GitHub for source code management and collaboration.

3. Testing and Validation

3.1 Testing Frameworks

Python Libraries:

- unittest or pytest for unit testing and integration testing of the codebase.
- TensorBoard for visualizing training and validation metrics.

Performance Testing Tools:

- Benchmarking Tools like perf or cProfile for performance profiling and optimization.

3.2 Validation Tools

Metrics Libraries:

- Scikit-learn for evaluating model performance using metrics such as accuracy, precision, recall, F1 score, and ROC-AUC.
- Confusion Matrix Visualization tools for analyzing model predictions.

5. Collaboration

5.1 Documentation Tools

- Google Docs for creating and maintaining project documentation.
- Markdown for writing and formatting technical documents and reports.

4. Design Specifications

4.1 System Architecture Diagram

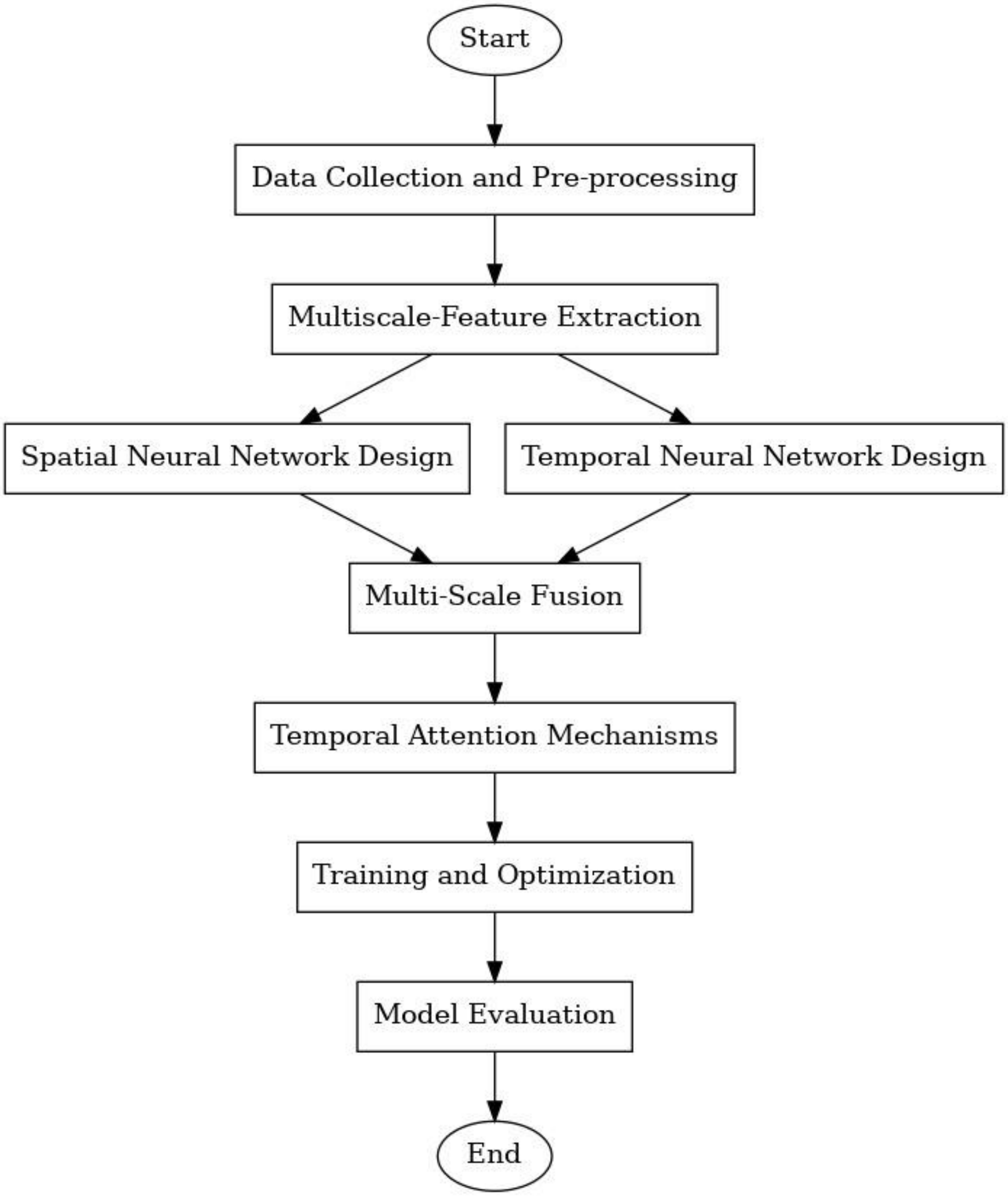
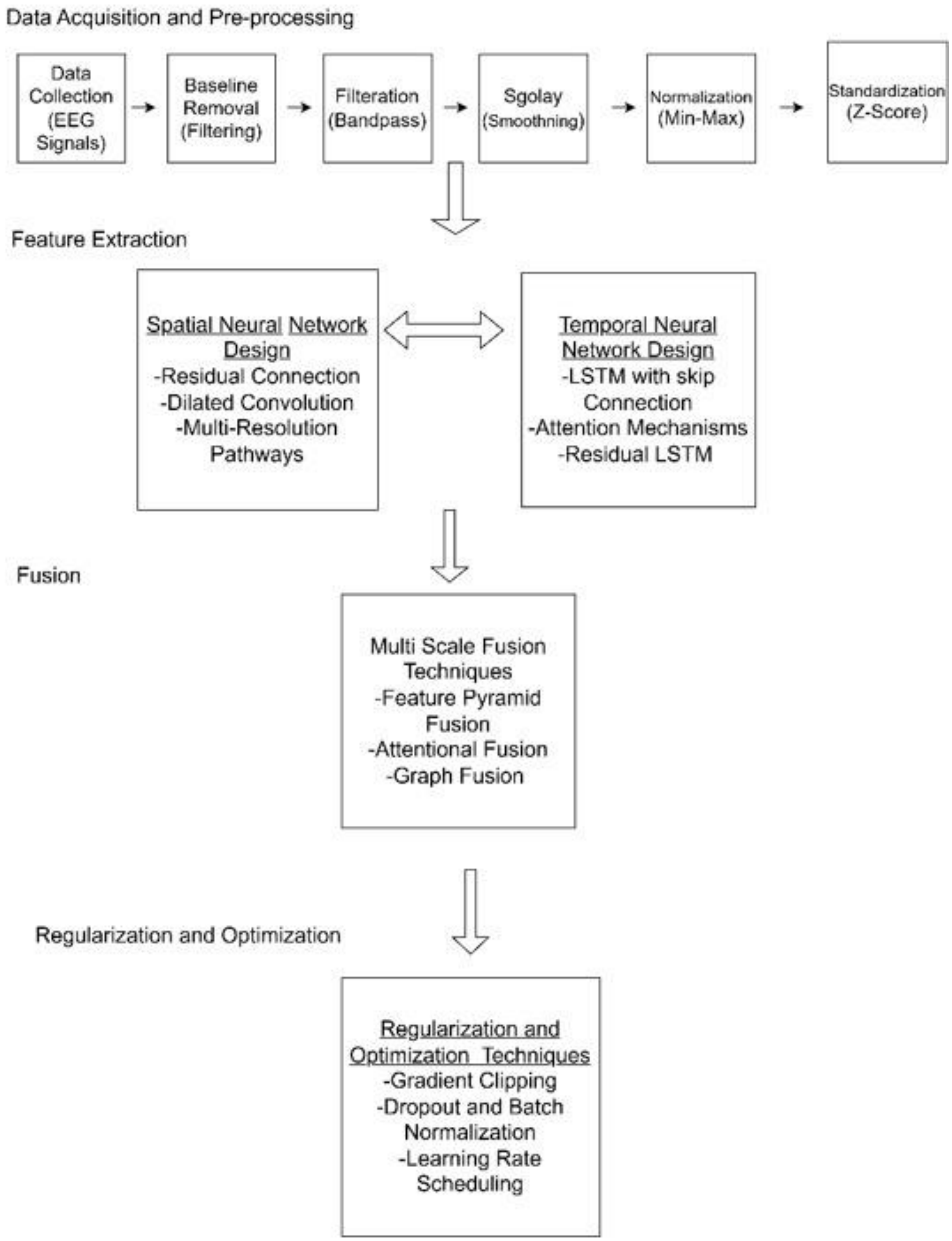


FIGURE 1: Methodology Block Diagram

4.2 Design Level Diagram



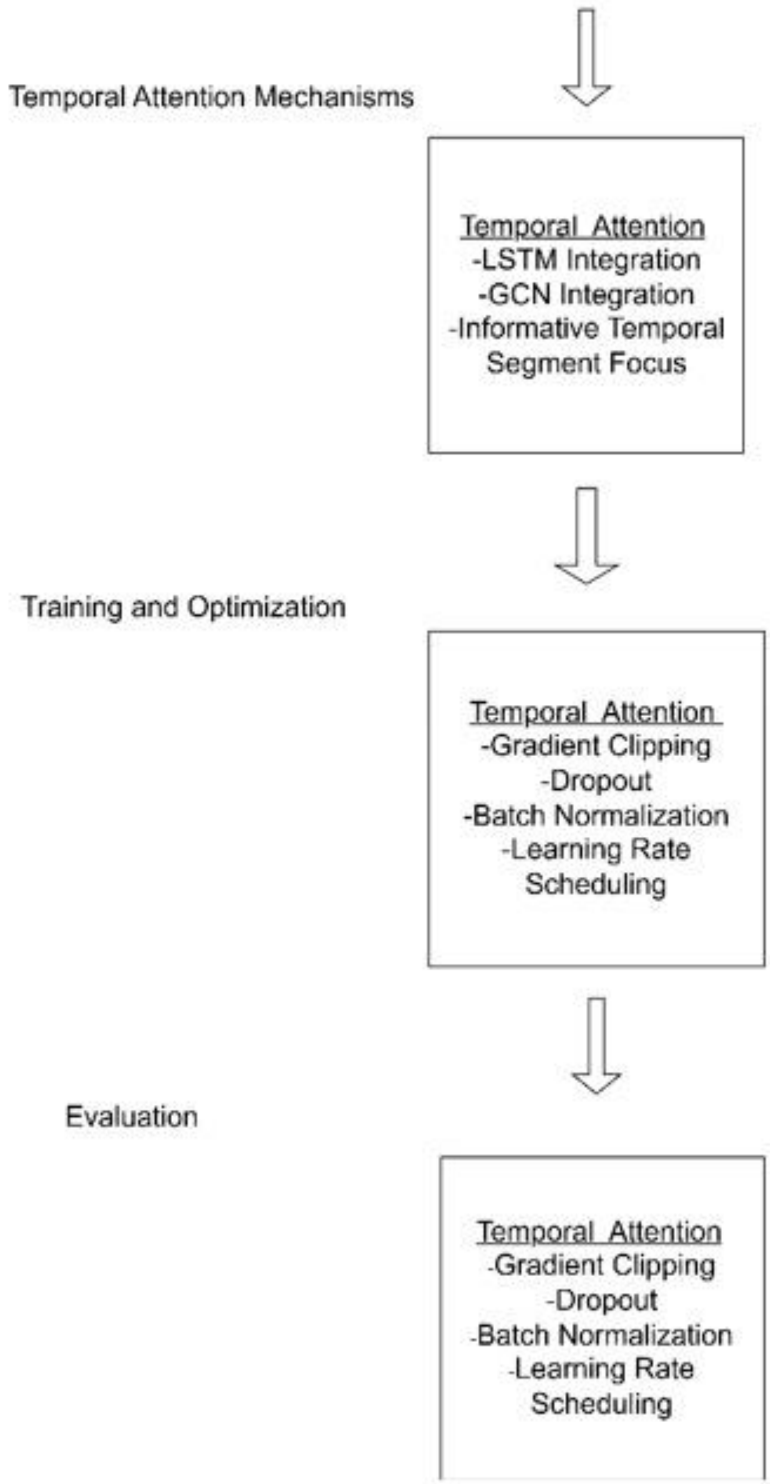
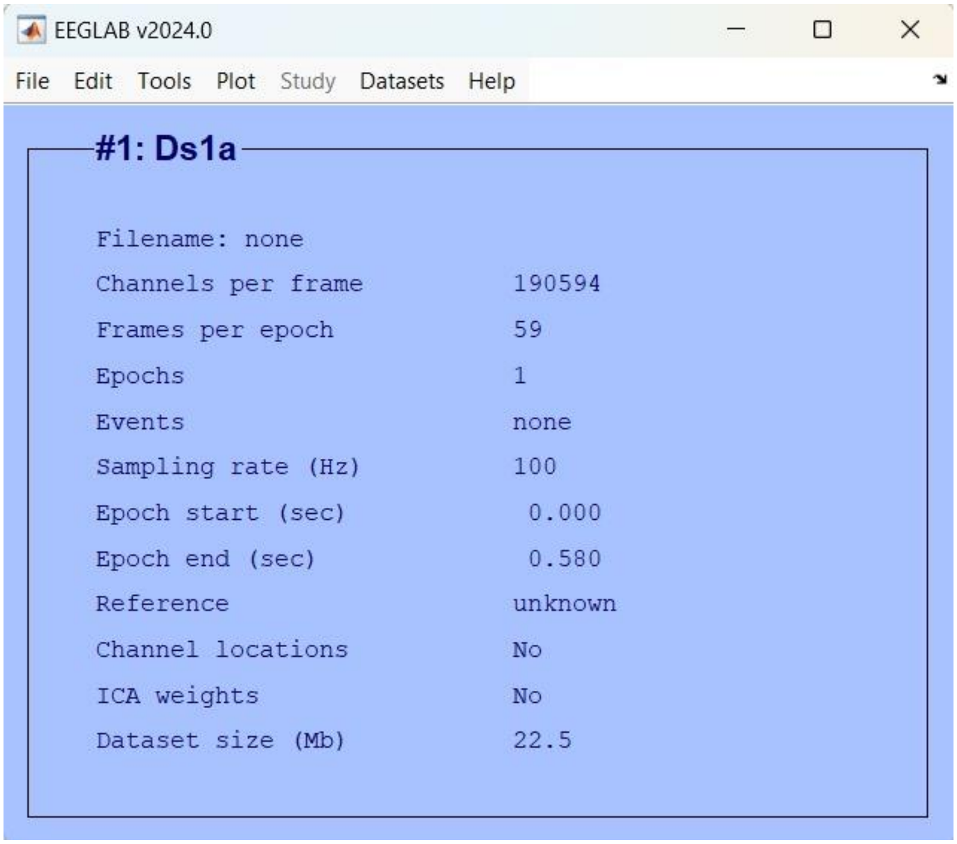


FIGURE 2: Design Level Diagram

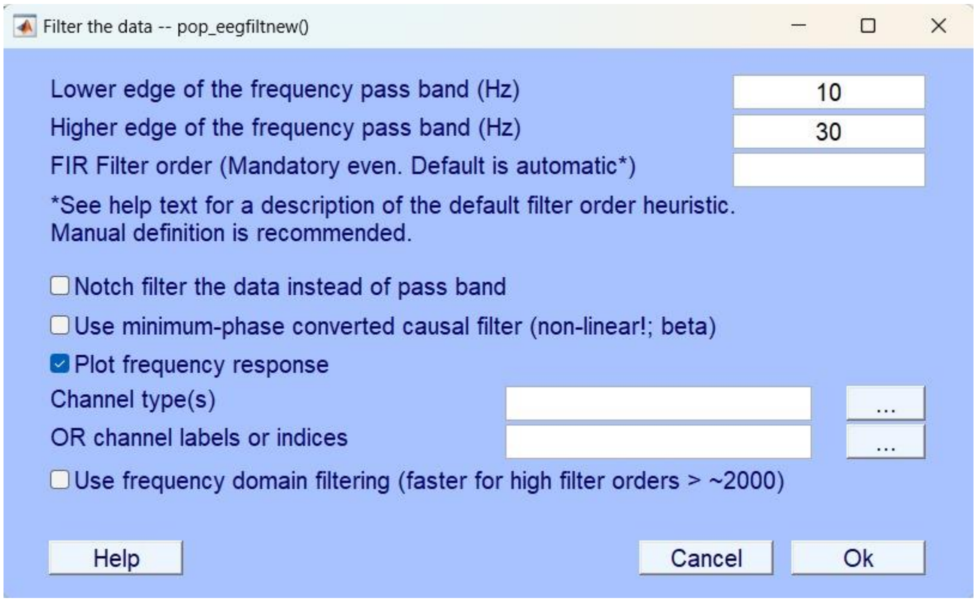
4.3 Working Snapshots

Data Preprocessing –

1. Baseline Removal



2. Band Pass Filtration



3. Sgolay Smoothening

Editor - C:\Users\vansh\AppData\Local\Microsoft\Windows\INetCache\IE\BNQDT89U\sgole1[1].m [Read Only]

sgole1[1].m | sgolay3[1].m | sgolay2[1].m

```matlab
% Load your data and select one channel
data = eeg_data; % Replace eeg_data with your actual data
channel = 29; % Select the channel to apply the filter

% Define the range of window sizes (odd sizes from 101 to 999)
window_sizes = 101:2:999;
num_sizes = length(window_sizes);

% Initialize arrays to store log dispersion values
log_dispersions = zeros(num_sizes, 1);

% Apply Savitzky-Golay filter for each window size and calculate log dispersion
for i = 1:num_sizes
    window_size = window_sizes(i);
    smoothed_data = sgolayfilt(data(:, channel), 3, window_size);
    log_dispersions(i) = std(log(abs(data(:, channel) - smoothed_data)));
end

% Find the optimal window size
[~, optimal_index] = min(log_dispersions);
optimal_window_size = window_sizes(optimal_index);


% Apply Savitzky-Golay filter with the optimal window size
optimal_smoothed_data = sgolayfilt(data(:, channel), 3, optimal_window_size);
optimal_window_size
```

Editor - C:\Users\vansh\AppData\Local\Microsoft\Windows\INetCache\IE\KY5G0S5M\sgolay2[1].m * [Read Only]

sgole1[1].m | sgolay3[1].m | sgolay2[1].m *

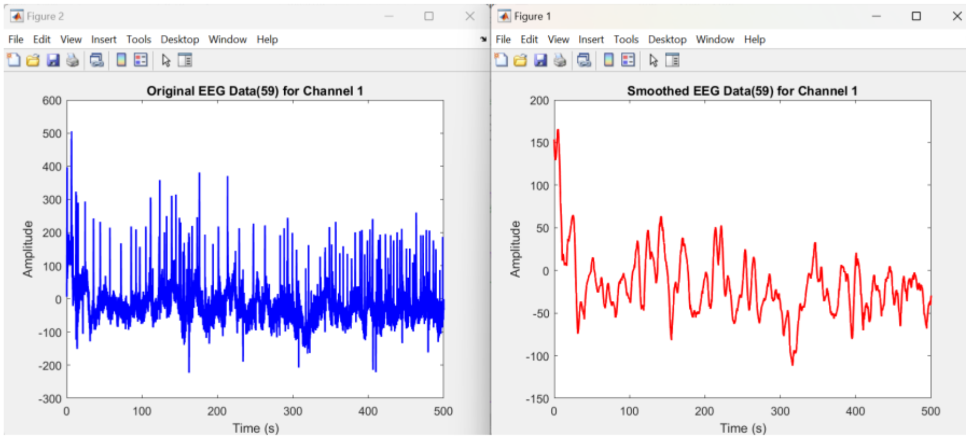
```matlab
% Load your data
data = eeg_data; % Replace your_data with your actual data
% Define the optimal window size
optimal_window_size = 971; % Assuming 701 is the optimal window size (change this to the actual optimal size)
% Initialize array to store mean log dispersion for each channel
mean_log_dispersions = zeros(size(data, 2), 1);

% Calculate mean log dispersion for each channel
for channel = 1:size(data, 2)
    % Apply Savitzky-Golay filter with the optimal window size
    smoothed_data = sgolayfilt(data(:, channel), 3, optimal_window_size);

    % Calculate the logarithm of the absolute differences
    log_abs_diff = log(abs(data(:, channel) - smoothed_data));

    % Calculate the mean log dispersion for the channel
    mean_log_dispersion = mean(log_abs_diff);

    % Store the mean log dispersion for the channel
    mean_log_dispersions(channel) = mean_log_dispersion;
end
% Display mean log dispersion for each channel
disp('Mean Log Dispersion for Each Channel:');
disp(mean_log_dispersions);

% Calculate and display overall mean log dispersion
overall_mean_log_dispersion = mean(mean_log_dispersions);
disp(['Overall Mean Log Dispersion: ' num2str(overall_mean_log_dispersion)]);
```

Editor - C:\Users\vansh\AppData\Local\Microsoft\Windows\INetCache\IE\LMQNAWTI\sgolay3[1].m [Read Only]

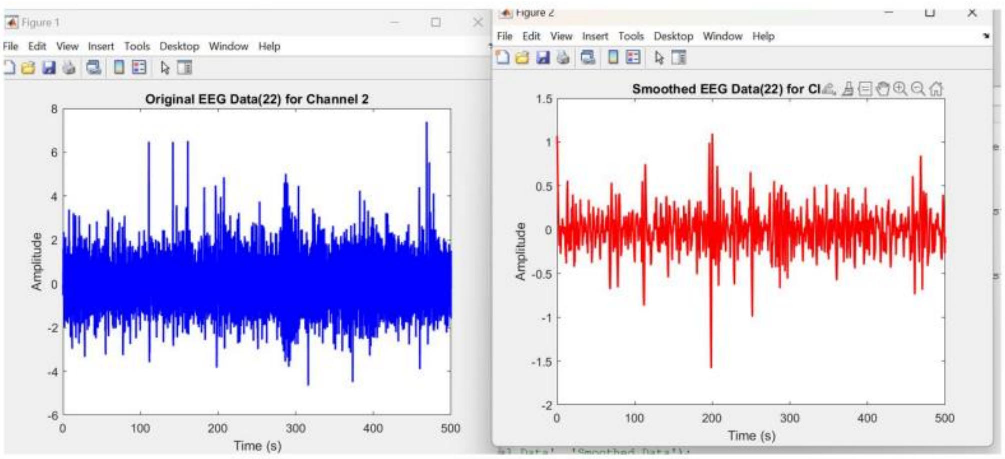
sgole1[1].m | sgolay3[1].m | sgolay2[1].m

```matlab
% Assuming eeg_data is a matrix of size [num_samples, 59] where each column represents a channel
[num_samples, num_channels] = size(eeg_data);
sampling_rate = 100;  % Assuming a sampling rate of 1000 samples per second
time = (0:num_samples-1) / sampling_rate;  % Time vector in seconds

% Applying Savitzky-Golay filtering to each channel
window_size = 103;
polynomial_order = 3;
smoothed_data = sgolayfilt(eeg_data, polynomial_order, window_size);

% Plotting the original and smoothed data for one channel (e.g., channel 1)
channel_to_plot = 1;
figure;
%plot(time, eeg_data(:, channel_to_plot), 'b', 'LineWidth', 1.5);
%hold on;
plot(time, smoothed_data(:, channel_to_plot), 'r', 'LineWidth', 1.5);
xlabel('Time (s)');
ylabel('Amplitude');
title('Original and Smoothed EEG Data for Channel 1');
legend('Original', 'Smoothed');
xlim([0, 500]);
```

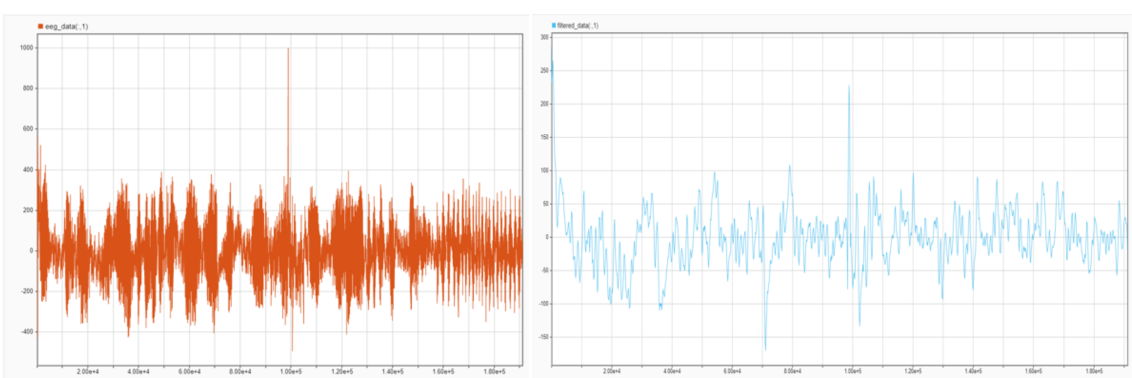
Log Dispersion




```
Sum of Log Dispersion for Original Data (Channel 1): 783326.9291
Sum of Log Dispersion for Smoothed Data (Channel 1): 740280.8008
```

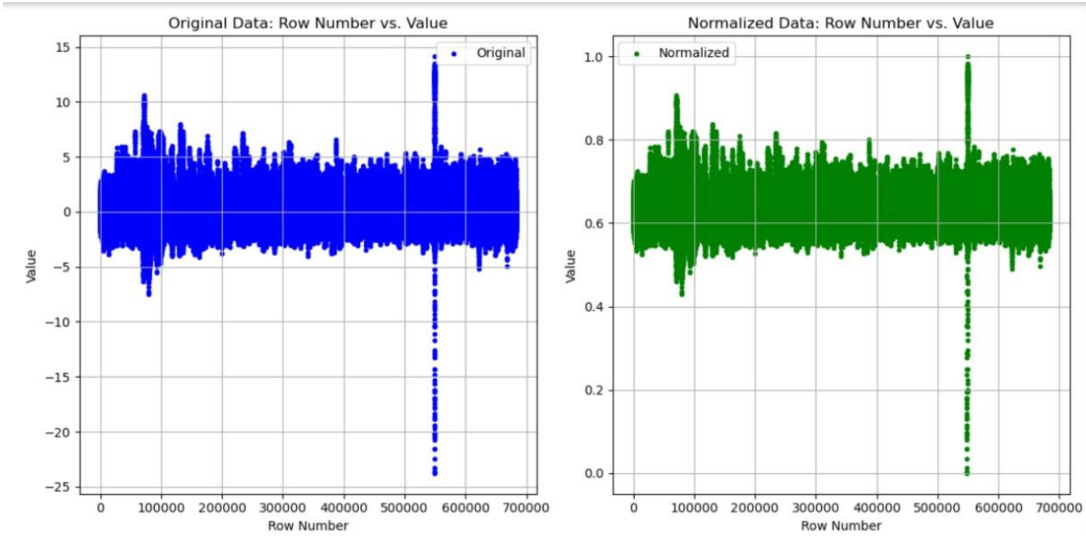



```
% xlim([0, 50000]);
Sum of Log Dispersion for Original Data (Channel 2): -709279.6918
Sum of Log Dispersion for Smoothed Data (Channel 2): -1811099.9793
```



```
Sum of Log Dispersion for Original Data (Channel 1): 817914.3119
Sum of Log Dispersion for Smoothed Data (Channel 1): 565228.029
```

4. Normalization & Standardization



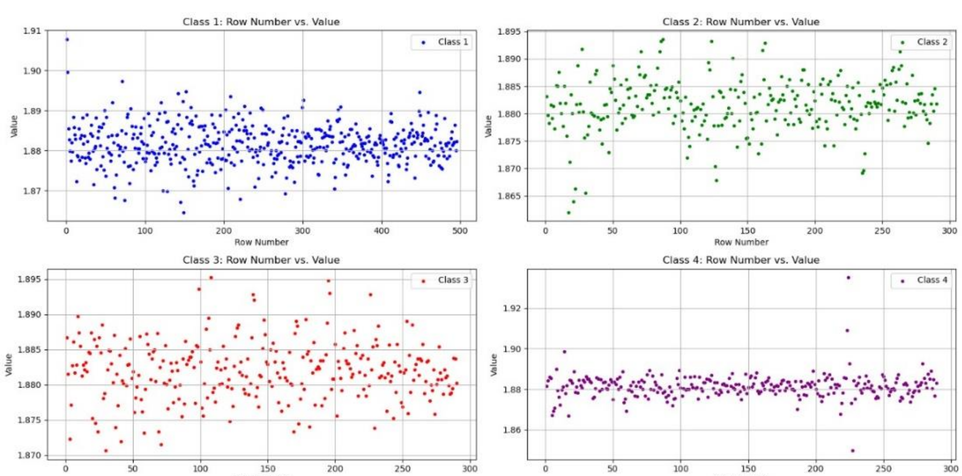
Feature Extraction-

Temporal

```
[2]: import pandas as pd
import numpy as np
import tensorflow as tf
from tensorflow.keras.layers import LSTM, Input, Dense, Add
from tensorflow.keras.models import Model

# Load the data
file_path = r"D:\Tools\BCICIV_1_mat\label_1_data.csv"
data = pd.read_csv(file_path)

# Separate the EEG data and ground truth
eeg_data = data.iloc[:, :-1].values  # first 59 columns
ground_truth = data.iloc[:, -1].values  # last column

# Function to create sliding windows
def create_sliding_windows(data, window_size=500, stride=500):
    windows = []
    for start in range(0, len(data) - window_size + 1, stride):
        window = data[start:start + window_size]
        windows.append(window)
    return np.array(windows)

# Create sliding windows
sliding_windows = create_sliding_windows(eeg_data, window_size=500, stride=500)
# Reshape for LSTM input
sliding_windows = sliding_windows.reshape(sliding_windows.shape[0], sliding_windows.shape[1], sliding_windows.shape[2])

# Build LSTM with skip connections for feature extraction
input_layer = Input(shape=(500, 59))
lstm_out = LSTM(64, return_sequences=False)(input_layer)
dense_out = Dense(64)(lstm_out)  # Match the dimension of lstm_out

# Skip connection
skip_connection = Add()([lstm_out, dense_out])

output_layer = Dense(32)(skip_connection)  # Final output layer for dimensionality reduction

model = Model(inputs=input_layer, outputs=output_layer)
model.compile(optimizer='adam', loss='mse')

# Extract features
extracted_features = model.predict(sliding_windows)

# Convert features into a DataFrame
features_df = pd.DataFrame(extracted_features)

# Save the extracted features to a new CSV file
features_df.to_csv("D:\Tools\BCICIV_1_mat\label1data.csv", index=False)

print("Feature extraction completed. Features saved to 'extracted_features.csv'.")

6/6 ━━━━━━━━━━━━━━━━━━━━ 1s 96ms/step
Feature extraction completed. Features saved to 'extracted features.csv'.
```

Spatial

```python
import pandas as pd
import numpy as np

# Load the CSV file
file_path = r"D:\desktop\Capstone-data\csv data\training_5_ground_truth_1.csv"
data = pd.read_csv(file_path)

# Separate features and ground truth
features = data.iloc[:, :-1].values  # First 22 columns as a numpy array
ground_truth = data.iloc[:, -1]  # Last column

# Define the window size
window_size = 500

# Function to extract features with a sliding window of 500 rows
def sliding_window_feature_extraction(data, window_size=500):
    num_rows = data.shape[0]
    num_features = num_rows // window_size
    extracted_features = np.zeros((num_features, data.shape[1]))

    for i in range(num_features):
        start_idx = i * window_size
        end_idx = start_idx + window_size
        window = data[start_idx:end_idx]
        extracted_features[i] = window.mean(axis=0)  # Example: mean feature

    return extracted_features

# Apply the sliding window
sliding_window_features = sliding_window_feature_extraction(features, window_size=window_size)

# Function to apply residual connection
def apply_residual_connection(features, residual_weight=0.5):
    num_features = features.shape[0]
    residual_features = np.zeros_like(features)
    residual_features[0] = features[0]  # No residual for the first feature

    for i in range(1, num_features):
        residual_features[i] = features[i] + residual_weight * features[i - 1]

    return residual_features

# Apply residual connection to features
features_with_residuals = apply_residual_connection(sliding_window_features)

# Truncate ground truth to match the number of extracted features
ground_truth_truncated = ground_truth[:(features_with_residuals.shape[0] * window_size):window_size].reset_index(drop=True)

# Combine features and ground truth into a single DataFrame
final_data = pd.DataFrame(features_with_residuals)
final_data['Ground_Truth'] = ground_truth_truncated

# Save the result to a new CSV file
final_data.to_csv('extracted_features_with_residuals_gt1.csv', index=False)
```



5. Conclusions and Future Scope

5.1 Work Accomplished

1. Data Acquisition

- EEG signals were acquired from three distinct datasets obtained through BCI (Brain-Computer Interface) competitions, each utilizing different configurations of EEG electrodes:
  Dataset 1: Recorded using a skull cap equipped with 22 electrodes.
  Dataset 2: Recorded using a skull cap equipped with 59 electrodes.
  Dataset 3: Recorded using a skull cap equipped with over 118 electrodes.
- These datasets were originally stored in GDF (General Data Format) files, which are standard in the field of EEG research. To facilitate further analysis and processing, these files were converted into CSV (Comma-Separated Values) format using MATLAB, ensuring compatibility with various data processing and machine learning tools.

2. Data Preprocessing:

- Following channel selection, the EEG data underwent a comprehensive preprocessing pipeline designed to enhance signal quality and prepare the data for subsequent neural network analysis:
- Handling Missing Values: The data was first checked for any missing values, specifically NaNs (Not a Number) and zeros, which can introduce noise and bias into the analysis. These values were systematically removed or imputed to ensure the integrity of the dataset.
- Baseline Correction: To account for any drift or baseline shifts in the EEG signals, baseline correction was applied. This process involved subtracting the mean value of a pre-stimulus period from the entire signal, thereby normalizing the data and removing any extraneous low-frequency noise.
- Filtering: The EEG signals were then subjected to bandpass filtering(10-30hz) to isolate the frequency bands of interest. This step is crucial in EEG analysis, as it helps to remove high-frequency noise (e.g., muscle artifacts) and low-frequency drift, while retaining the relevant signal components associated with motor imagery tasks.
- Smoothing (Sgolay Filtering): After filtering, the data was smoothened using Savitzky-Golay (Sgolay) filtration for order of 3. This technique fits successive polynomials to the data in a

moving window, effectively reducing noise while preserving the shape and features of the EEG signal. The result is a cleaner signal that is better suited for subsequent analysis.

- Normalization: To ensure that the data from different channels and subjects was on a comparable scale, normalization was applied. This step involved scaling the data to a common range, typically between 0 and 1, which is essential for minimizing the impact of outliers and ensuring that the neural network can learn effectively from the data.
- Standardization: Finally, standardization was performed to further refine the data. This process involved transforming the data to have a mean of zero and a standard deviation of one, ensuring that each feature contributed equally to the analysis. Standardization is particularly important for neural network models, as it helps in speeding up convergence during training and improving overall model performance.

3. Spatial-Temporal Feature Extraction:

In the process of analyzing EEG signals for motor imagery classification, it is crucial to extract both spatial and temporal features to capture the intricate dependencies across EEG channels and the dynamic nature of brain activity over time. The following methods were implemented to achieve this:

- Graph Neural Networks (GNNs): To effectively capture the spatial dependencies between EEG channels, Graph Neural Networks were employed. GNNs treat the EEG channels as nodes in a graph, where the edges represent the relationships or correlations between these channels. This approach allows for the extraction of spatial features that encapsulate the topographical structure of the brain's electrical activity, providing a richer representation of the spatial patterns present in the EEG data.
- Bi-Directional Long Short-Term Memory (Bi-LSTM) Networks: For modeling the temporal dynamics of the EEG signals, Bi-Directional LSTM networks were utilized. LSTMs are well-suited for handling sequential data due to their ability to learn long-term dependencies. The bi-directional nature of these LSTM networks ensures that both past and future temporal contexts are considered, enhancing the extraction of temporal features that reflect the evolution of brain activity over time.

4. Multiscale Feature Extraction

To further refine the spatial and temporal features extracted from the EEG data, a multiscale feature extraction approach was adopted. This involves the design and implementation of specialized neural network architectures to capture features at different scales.

1. Spatial Neural Network Design

- Residual Connections: Within the spatial neural network architecture, residual connections, also known as skip connections, were implemented. These connections allow gradients to flow more easily through the network during training, effectively mitigating the vanishing gradient problem. This architectural choice enhances the model's ability to learn deep spatial representations by preserving the flow of information across layers.
- Dilated Convolutions: Dilated convolutions were utilized to increase the receptive field of the convolutional layers without increasing the number of parameters. This technique enables the network to capture spatial features at multiple scales efficiently, allowing it to recognize patterns that span different spatial resolutions within the EEG data.
- Multi-Resolution Pathways: To explicitly capture features at varying spatial scales, multi-resolution pathways were designed within the network. These pathways incorporate parallel convolutional layers with different kernel sizes, enabling the extraction of spatial features at multiple resolutions. This approach ensures that the network can simultaneously focus on both fine-grained and coarse-grained spatial patterns, enhancing the overall robustness of the spatial feature extraction process.

2. Temporal Neural Network Design

- LSTM with Skip Connections: To capture temporal dependencies within the EEG signals, LSTM layers with skip connections were employed. These skip connections facilitate the flow of gradients across long temporal sequences, ensuring that the network can effectively learn from temporal patterns that span multiple time steps. This approach is particularly beneficial for modeling complex temporal dynamics in EEG data.
- Attention Mechanisms: Attention mechanisms were integrated within the temporal neural network to dynamically focus on the most informative segments of the EEG data. By adaptively attending to relevant temporal scales during feature extraction, the network can prioritize the temporal patterns that are most critical for accurate motor imagery classification, thereby improving overall model performance.
- Residual LSTM Blocks: To further enhance the learning of robust temporal representations, residual LSTM blocks were designed and implemented. These blocks, which incorporate residual connections, ease the flow of gradients through the network, mitigating the vanishing gradient problem. This architectural design encourages the network to learn more stable and reliable temporal features, contributing to better generalization and performance.

The above methods were applied to the EEG data after separating the files according to the ground truth labels. By aligning the data preprocessing and feature extraction processes with the specific motor imagery tasks, the system was able to extract meaningful spatial and temporal features that are critical for accurate classification. This systematic approach ensures that the model captures the most relevant patterns within the EEG data, leading to improved performance in motor imagery classification tasks.

5.2 Conclusions

The project has successfully laid a robust foundation for EEG-based motor imagery classification by meticulously preprocessing the data and implementing advanced spatial and temporal feature extraction techniques. Using EEG data from 22, 59, and 118 channels, we have completed baseline correction, bandpass filtration, Savitzky-Golay smoothing, log dispersion calculation, and the subsequent standardization and normalization of the data. Building on this, we designed and implemented sophisticated neural network architectures with residual connections, dilated convolutions, and multi-resolution pathways to capture spatial features, while also leveraging LSTM layers with skip connections, attention mechanisms, and residual LSTM blocks to extract and enhance temporal features.

Moving forward, the focus will shift towards integrating Graph Neural Networks (GNNs), BiLSTMs, and Deep Belief Attention Networks (DBANs) to further optimize the system. The next steps will involve refining channel selection, reducing dimensionality through manifold learning, and training the model on multiple datasets to enhance generalization. We will also explore interpretability with Grad-CAM and benchmark the performance against state-of-the-art methods to ensure the development of a highly accurate, reliable, and efficient EEG-based motor imagery classification system. This progression will play a significant role in advancing neurorehabilitation, assistive technology, and brain-computer interfaces.

5.3 Environment, Economic, and Social Benefits

The development of advanced neural networks for motor imagery classification using EEG data offers notable environmental, economic, and social benefits, making a positive impact on society and the environment.

1. Environmental Benefits

This project contributes to environmental sustainability by:

- Reducing Material Waste: By enhancing accuracy in motor imagery classification, the project minimizes the need for excessive materials, such as printed data and redundant hardware components, thereby reducing waste.

- Energy Efficiency: The use of efficient neural network architectures lowers computational demands, resulting in reduced energy consumption and a smaller carbon footprint, particularly in real-time EEG data processing.
- Promotion of Remote Healthcare: Improved BCIs support remote healthcare, reducing travel and the associated greenhouse gas emissions, while also lessening the environmental impact of traditional healthcare practices.

2. Economic Benefits

The project's economic impact includes:

- Cost-Effective Healthcare Solutions: Enhanced accuracy in motor imagery classification leads to more effective and tailored neurorehabilitation, reducing treatment costs and easing the financial burden on healthcare systems and patients.
- Increased Productivity in Research: Streamlined research processes and faster data analysis save time and resources, accelerating technological advancements and economic growth in the neurotechnology sector.
- Market Expansion Opportunities: Successful outcomes may drive market expansion in BCIs and assistive devices, fostering job creation, economic growth, and the development of new products and services.

3. Social Benefits

The social impact is significant, particularly in:

- Empowerment of Individuals with Disabilities: By improving BCI effectiveness, the project enhances independence and quality of life for individuals with motor impairments, promoting greater social and economic participation.
- Support for Neurorehabilitation: The project advances neurorehabilitation, leading to better patient outcomes, faster recovery, and reduced burden on caregivers and healthcare providers.
- Promotion of Inclusive Innovation: By focusing on accessibility and usability, the project ensures that advanced neurotechnology benefits a diverse range of individuals, promoting social inclusion and bridging the gap between technological advancements and societal needs.
- Educational Impact: The project contributes to education and training in neurotechnology, fostering the development of future experts and ensuring the sustainability of the field.

5.4 Future Work Plan

1.Multi-Scale Fusion:

- Feature Pyramid Fusion: Merge features extracted at different spatial and temporal scales using feature pyramid fusion techniques. This allows for the creation of a comprehensive feature representation that captures information across multiple scales.
- Attentional Fusion: Employ attention mechanisms to dynamically combine multiscale features based on their relevance to the task at hand. This enhances the discriminative power of the feature representation while reducing the impact of irrelevant information.
- Graph Fusion: If dealing with multichannel data, leverage graph-based fusion techniques to integrate features across different channels effectively. This can be achieved using graph convolutional networks or graph attention networks tailored to the spatial relationships between EEG channels.

2. Regularization and Optimization:

- Gradient Clipping: Apply gradient clipping to prevent exploding gradients during training, especially in deep architectures with residual connections.
- Dropout and Batch Normalization: Incorporate dropout and batch normalization layers to regularize the network and stabilize training by reducing overfitting and internal covariate shift.
- Learning Rate Scheduling: Use learning rate scheduling techniques to adaptively adjust the learning rate during training, promoting faster convergence and better generalization.

3. Temporal Attention Mechanisms:

- Incorporate temporal attention mechanisms to focus on informative segments of the EEG data.
- Use mechanisms such as attentional pooling or attentional gating to dynamically weigh the importance of different temporal segments.
- Enhance the discriminative power of the model by emphasizing relevant temporal dynamics

4. Model Architecture:

- Design a model architecture that integrates stacked LSTM layers with graph convolutional neural networks (GCN).
- Use stacked LSTM layers to capture temporal dependencies within the EEG data.
- Employ GCN to capture spatial dependencies between EEG channels, treating the multichannel EEG data as a graph structure.
- Fuse the output representations from the LSTM and GCN layers to combine temporal and spatial information.

5. Training and Optimization:

- Split the pre-processed data into training, validation, and testing sets.
- Train the model using the training set with appropriate optimization techniques, such as stochastic gradient descent (SGD) or Adam optimization.
- Tune hyperparameters through cross-validation on the validation set to optimize model performance.
- Regularize the model to prevent overfitting, using techniques like dropout or L2 regularization.
- Interpretability with Grad-CAM: Apply Grad-CAM to visualize the regions of interest in the EEG data that contribute most to the model's predictions.
- Generate class activation maps highlighting the important EEG channels and temporal segments for each motor imagery task.
- Interpret the Grad-CAM visualizations to gain insights into the neural correlates of motor imagery tasks captured by the model.

6. Evaluation:

- Evaluate the trained model's performance on the held-out testing set using appropriate metrics such as accuracy, precision, recall, and F1 score.
- Assess the model's generalization capability across different subjects or experimental conditions.
- Compare the model's performance with baseline methods or state-of-the-art approaches in motor imagery classification.

REFERENCES

[1] S. Biao, Z. Liu, Z. Wu, C. Mu, T. Li. "Graph convolution neural network based end-to-end channel selection and classification for motor imagery brain–computer interfaces." IEEE transactions on industrial informatics vol. 19(9), 2022, pp.9314-9324.

[2] M. Arvaneh, C. Guan, K. K. Ang and C. Quek. "Optimizing the Channel Selection and Classification Accuracy in EEG-Based BCI," in IEEE Transactions on Biomedical Engineering, vol. 58, no. 6, June 2011, pp.1865-1873.

[3] M. Almufareh, S. Kausar, M. Humayun, S. Tehsin. “Leveraging Motor Imagery Rehabilation for Individuals with Disabilities” Internet: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10572951/ , Sep. 29, 2023 [August. 22, 2024]

[4] L. Haifeng, Z. You, Y. Guo, X. Hu. "MAST-GCN: Multi-Scale Adaptive Spatial-Temporal Graph Convolutional Network for EEG-Based Depression Recognition.", in IEEE Transactions on Affective Computing, 2024.

[5] Y. Wang, W. Cui, T. Yu, X. Li, X. Liao “Dynamic multi-graph convolution based channel-weighted transformer feature fusion network for epileptic seizure prediction”, in IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2023.

[6] C. Yi, Y. Qiu, W. Chen, C. Chen, Y. Wang, P. Li, P. Xu, X. Zhang, L. Jiang, D. Yao, F. Li, L. Yang. "Constructing Time-Varying Directed EEG Network by Multivariate Nonparametric Dynamical Granger Causality." IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol. 30, 2022.

[7] S. Biao, H. Zhang, Z. Wu, Y. Zhang, T. Li. "Adaptive spatiotemporal graph convolutional networks for motor imagery classification." in IEEE Signal Processing Letters, 2021, pp.219-223

[8] X. Li, B. Qian, J. Wei, A. Li, X. Liu, Q. Zheng. “Classify EEG and reveal latent graph structure with spatio-temporal graph convolutional neural network”, in IEEE International Conference on Data Mining (ICDM), November 2019, pp. 389-398.

[9] Izenman, A. J. “Introduction to manifold learning. Wiley Interdisciplinary Reviews: Computational Statistics”, 2020, 439-446.

[10] Cheng, L., Li, D., Yu, G., Zhang, Z., Li, X., & Yu, S. “A motor imagery EEG feature extraction method based on energy principal component analysis and deep belief networks”, in IEEE Access, 2020, pp.21453-21472.

[11] https://www.bbci.de/competition/

[12] https://openbci.com/community/publicly-available-eeg-datasets/