

# **ENHANCING EEG ANALYSIS FOR RAPID BRAIN ACTIVITIES DETECTION IN PATIENTS**

**A PROJECT REPORT**

*Submitted by*

**ABHIVYAKTI YADAV**

**(Reg. No. CH.EN.U4AIE21101)**

**KAIF AHMED R**

**(Reg. No. CH.EN.U4AIE21119)**

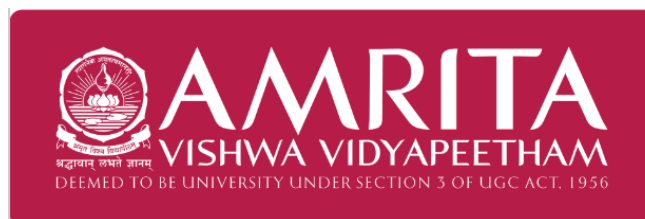
*in partial fulfillment for the award of the degree of*

**BACHELOR OF TECHNOLOGY IN COMPUTER SCIENCE AND  
ENGINEERING (ARTIFICIAL INTELLIGENCE)**

*Under the guidance of*

**Dr. R ANNAMALAI**

**Submitted to**

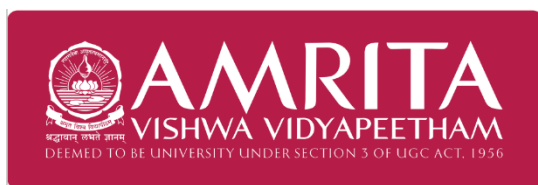


**AMRITA VISHWA VIDYAPEETHAM**

**AMRITA SCHOOL OF COMPUTING**

**CHENNAI – 601103**

**October 2024**



**SCHOOL OF  
COMPUTING  
CHENNAI**

## **BONAFIDE CERTIFICATE**

This is to certify that this project report entitled “**ENHANCING EEG ANALYSIS FOR RAPID BRAIN ACTIVITIES DETECTION IN PATIENTS**” is the bonafide work of “**Ms. ABHIVYAKTI YADAV (Reg.No. CH.EN.U4AIE21101) & Mr. KAIF AHMED R (Reg.No. CH.EN.U4AIE21119)**”, who carried out the project work under my supervision.

**SIGNATURE**

**Dr. S SOUNTHARRAJAN**

**CHAIRPERSON**

Department of CSE.

Amrita School of Computing

Chennai.

**SIGNATURE**

**Dr. R ANNAMALAI**

**SUPERVISOR**

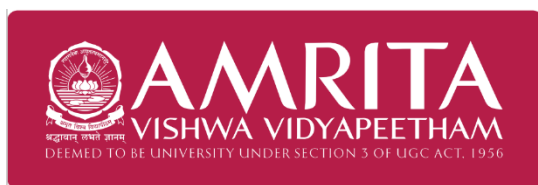
Department of CSE(AIE).

Amrita School of Computing

Chennai.

**INTERNAL EXAMINER**

**EXTERNAL EXAMINER**



**SCHOOL OF  
COMPUTING  
CHENNAI**

## **DECLARATION OF CANDIDATE**

I declare that the report entitled **“ENHANCING EEG ANALYSIS FOR RAPID BRAIN ACTIVITIES DETECTION IN PATIENTS”** submitted by me for the degree of Bachelor of Technology is the record of the project work carried out by me under the guidance of **“DR. R ANNAMALAI”** and this work has not formed the basis for the award of any degree, diploma, associateship, fellowship, titled in this or any other University or other similar institution of higher learning.

**SIGNATURE**

**KAIF R AHMED**

**(Reg.No. CH.EN.U4AIE21119)**

**SIGNATURE**

**ABHIVYAKTI YADAV**

**(Reg. No. CH.EN.U4AIE21101)**

## ABSTRACT

Electroencephalogram is a well-developed, widely established medical tool that helps in diagnosing various neurological diseases based on identifying abnormal brain activities such as seizures, sleep disorders, and many more impairments. Unfortunately, traditional analysis of EEG signals often cannot handle the diversity of data and its high dimensionality due to the complexity and noisiness of the latter, making it quite challenging to deal with this data. A novel classification method for EEG signals is presented, that effectively utilizes the power of both machine learning and deep learning models to overcome the challenges.

Our study was specifically focused on state-of-the-art models, including CatBoost, ResNet34D, EfficientNetB0, and EfficientNetB2. We were trying to propose an approach based on a large EEG database consisting of 11,000 samples of spectrograms. The creation of uniform EEG spectrograms has been possible through statistical measures, drawing relevant features, for the input data so that it could be representative and conducive to model training.

In order to measure the goodness of fit for each of the above models, we used the Kullback-Leibler (KL) divergence score as a vital evaluation metric. KL divergence measures the difference in two discrete distributions. So, it gives a notion of how good the models perform for classification. Amongst all, CatBoost performed the best, acquiring a KL divergence score of 0.78, thereby outperforming its peers on deep learning architectures.

This work demonstrates the prospects of gradient boosting algorithms, particularly CatBoost, for solving EEG activity classification problems. Such models could become an efficient extension of deep learning algorithms, providing useful insights into real-time brain activity monitoring and diagnostics of neurological disorders. The strengths of the traditional approaches and modern techniques will make them useful contributions towards a growing body of knowledge aimed at improving the accuracy and efficiency in interpreting EEG signals.

**Keywords:** EEG Signal Classification, CatBoost, Deep Learning, ResNet34D, EfficientNet, Kullback-Leibler Divergence, Brain Activity Detection, Feature Engineering.

## ACKNOWLEDGEMENT

This project work would not have been possible without the contribution of many people. It gives me immense pleasure to express my profound gratitude to our honorable Chancellor **Sri Mata Amritanandamayi Devi**, for her blessings and for being a source of inspiration. I am indebted to extend my gratitude to our **Sampoojya Swami Vinayamritananda Puri**, Administrative Director, and **Shri. I B Manikantan**, Campus Director for facilitating us all the facilities and extended support to gain valuable education and learning experience.

I register my special thanks to **Dr. V. Jayakumar**, Principal for the support given to me in the successful conduct of this project. I wish to express my sincere gratitude to **Dr. S Sountharajan**, Chairperson, CSE, **Dr. S. Baghavathi Priya**, Program Chair, CSE-AIE and **Dr. R Annamalai**, Supervisor, CSE-AIE for their inspiring guidance, personal involvement and constant encouragement during the entire course of this work.

I am grateful to Review Panel Members and the entire faculty of the Department of Computer Science & Engineering, for their constructive criticisms and valuable suggestions which have been a rich source to improve the quality of this work.

**ABHIVYAKTI YADAV & KAIF AHMED R**  
**(Reg. No. CH.EN.U4AIE21101 & Reg. No.CH.EN.U4AIE21119)**

## TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	<b>ABSTRACT</b>	<b>iv</b>
	<b>LIST OF TABLES</b>	<b>ix</b>
	<b>LIST OF FIGURES</b>	<b>x</b>
	<b>LIST OF SYMBOLS AND ABBREVIATIONS</b>	<b>xi</b>
<b>1</b>	<b>INTRODUCTION</b>	<b>1</b>
	1.1 Background on EEG	1
	1.2 Challenges in EEG Analysis	1
	1.3 Evolution of EEG Analysis Techniques	2
	1.4 Emergence of Deep Learning in EEG Analysis	3
	1.5 Motivation and Objective of the Study	3
	1.6 Scope of the Study	4
<b>2</b>	<b>LITERATURE REVIEW</b>	
	2.1 Overview of Traditional Machine Learning Approaches in EEG Analysis	5
	2.2 Deep Learning Techniques for EEG Classification	5
	2.3 Hybrid Approaches: Combining CNNs and LSTMs	6
	2.4 Recent Advances: Transfer Learning and Pre-trained Models	6
	2.5 Ensemble Learning with CatBoost: A Competitive Alternative	7
	2.6 Identified Gaps and Research Direction	7

<b>3</b>	<b>PROBLEM STATEMENT</b>	<b>9</b>
<b>4</b>	<b>PROPOSED WORK</b>	<b>11</b>
	4.1 Overview	11
	4.2 Dataset Description	11
	4.3 Data Preprocessing	12
	4.4 Feature Engineering	14
	4.5 Model Architectures and Training	15
	4.5.1 CatBoost Model	15
	4.5.1.1 Architecture	15
	4.5.1.2 Training Process	16
	4.5.2 CNN-LSTM Hybrid Model	16
	4.5.3 EfficientNet Models (B0/B2)	17
	4.5.3.1 Architecture Details	17
	4.5.4 ResNet50 Model	18
	4.6 Model Evaluation Metrics	19
	4.7. Output	20
	4.8 Implementation Framework	21
	4.9 Simulation Setup	21
<b>5</b>	<b>RESULTS AND DISCUSSIONS</b>	<b>24</b>
	5.1 Performance of CatBoost Model	24
	5.2 Comparison with Deep Learning Models	25
	5.3 Analysis of Performance Metrics	25
	5.4 Computational Efficiency	26
	5.5 Interpretation of Results	27

<b>6</b>	<b>CONCLUSION AND SCOPE FOR FURTHER WORK</b>	<b>29</b>
	6.1 Performance and Accuracy Trade-offs	29
	6.2 Computational Efficiency and Practical Considerations	29
	6.3 Implications for Real-time and Scalable Applications	30
	6.4 Future Directions	30
	6.5 Conclusion Summary	30
	<b>REFERENCES</b>	<b>31</b>



## LIST OF TABLES

TABLE NO.	TITLE	PAGE NO.
5.1	CatBoost Model Performance Table	24
5.2	Model Performance Comparison Table	25
5.3	Model Efficiency Comparison Table	27

## LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
4.1	Workflow Diagram	11
4.2	EEG Spectrogram Dataset	12
4.3	Extraction and preprocessing of EEG Spectrogram	14
4.4	CatBoost Architecture	16
4.5	CNN-LSTM Hybrid Model Architecture	17
4.6	EfficientNet Architecture	18
4.7	ResNet50 Model	19
4.8	Feature Importance Graph	21
5.1	KL-Divergence Plot	28

## LIST OF SYMBOLS AND ABBREVIATIONS

<b>EEG</b>	–	Electroencephalography
<b>fMRI</b>	–	Functional Magnetic Resonance Imaging
<b>PET</b>	–	Positron Emission Tomography
<b>Hz</b>	–	Hertz
<b>DSP</b>	–	Digital Signal Processing
<b>STFT</b>	–	Short-Time Fourier Transform
<b>WT</b>	–	Wavelet Transform
<b>SVM</b>	–	Support Vector Machine
<b>k-NN</b>	–	k-Nearest Neighbors
<b>RF</b>	–	Random Forests
<b>HMM</b>	–	Hidden Markov Model
<b>CNN</b>	–	Convolutional Neural Network
<b>RNN</b>	–	Recurrent Neural Network
<b>LSTM</b>	–	Long Short-Term Memory
<b>GRU</b>	–	Gated Recurrent Unit
<b>BCI</b>	–	Brain-Computer Interface
<b>ML</b>	–	Machine Learning
<b>DL</b>	–	Deep Learning
<b>KL</b>	–	Kullback-Leibler
<b>GPU</b>	–	Graphics Processing Unit

# CHAPTER 1

## 1. INTRODUCTION

### 1.1 Background on EEG

EEG is a technique recording electrical activity of the human brain from electrodes placed on the scalp. The technique was developed in the early years of the last century and has now evolved into a technique with widespread utility in both clinical and research settings. High temporal resolution of EEG makes it especially suitable for studying dynamic processes in the brain, as opposed to other imaging techniques such as fMRI or PET, which are better suited to higher spatial resolution images.

The principle of the EEG is based on the detection of voltage fluctuations caused by ionic current flows within neurons. Typically, it measures changes across different frequency bands: delta (0.5-4 Hz), theta (4-8 Hz), alpha (8-13 Hz), beta (13-30 Hz), and gamma (>30 Hz). Each sort of wave band has associations with different cognitive or physiological states. For example, alpha waves are generally associated with restful states, while beta waves are associated with active thinking and focus.

EEG is widely applied in diagnosing neurological disorders such as epilepsy, wherein the abnormal brain activities or seizures present with specific patterns on the EEG. It also helps in diagnosing sleep disorders, injuries to the brain, and conditions like Alzheimer's disease wherein cognitive decline is reflected through changes in EEG rhythms. Going beyond the clinical domain, EEG is an important tool in cognitive neuroscience in order to understand processes like attention, memory, and language comprehension.

### 1.2 Challenges in EEG Analysis

Information extraction from EEG data is associated with several challenges. Examples of these are as follows. EEG signals contain noise and artifacts. Common sources of artifacts that may appear during a recording session include blinks of eyes, muscle movements known as electromyographic noise, and even heartbeats. These artifacts may either result in false detection or conceal important patterns in the traces of the EEG.

The other difficulty is the inherent non-stationarity of EEG signals, which means that statistical properties of the signal can vary with time. Non-stationarity happens to be a crucial problem when attempting to classify data from EEG because the same cognitive state, different patterns can be produced in different recording sessions or even by a given person. This calls for a class of advanced techniques that can be adaptive over changing patterns in time, like adaptive filtering and dynamic modeling.

It is high dimensional data from EEG recordings. A recording can contain anything from 16 to 256 electrodes, with each electrode capturing a time series over relatively long periods of time. The vast amount of data that this generates proves computationally expensive

to process and analyze. Feature extraction often becomes the only recourse for methods that need to reduce this complexity. But choosing the right features is difficult and requires expert domain knowledge. For example, the choice of frequency bands or the calculation of statistical measures such as entropy or power spectral density will significantly affect the performance of a classification model.

EEG signals are associated with very high inter-subject and intra-subject variability. Thus, differences in scalp thickness, head shape, and electrode placement may lead to differences in the recorded signals, although the same type of cognitive tasks are performed. Intra-subject variability requires models that are robust and generalizable across different subjects, which has been one of the difficult tasks in this field.

### **1.3 Evolution of EEG Analysis Techniques**

Advances in signal-processing and computational methods have characterized the development of EEG analysis techniques. The first techniques to be employed used measures of the time domain, such as amplitude and waveform morphology, for clinician inspection. With technological advancements came frequency-domain techniques, employing the Fourier transform and wavelet transforms, to decompose EEG signals into their constituent frequency bands, thus allowing more sophisticated analyses over the rhythms in the brain.

Digital signal processing is an application that uses Non-stationary signal decomposition, including STFT and WT. The application in EEG is quintessential. The methods can break down a signal into its time-frequency representation, which gives insight into how the spectral content of EEG changes with time, for instance. WT has shown applications in the analysis of EEG epileptic spikes and sleep spindles, which are not captured properly by simple time-domain analysis.

Although this comes at the cost of requiring considerable domain knowledge in the selection of appropriate features, this has motivated techniques based on machine learning that can automate feature selection tasks. The various approaches started by shallow classifiers, such as SVMs, k-NN, and random forests. For example, SVMs are particularly suited to binary classification, distinguishing between seizure states and non-seizure states; their performance drops drastically, however, when dealing with complex, high-dimensional multiclass EEG data.

Random forests being ensemble methods are much more flexible and robust as compared to the classification tasks; they are more capable of handling feature importance and interaction between variables. The limitation of such models is in the application of time-series nature of EEG signals wherein time dependencies play a prime role. That opened the door to some more advanced models like hidden Markov models and then to the deep learning methods.

## **1.4 Emergence of Deep Learning in EEG Analysis**

Deep learning indeed is a competitor natively able to learn hierarchical representations of the data automatically, without any need for manual feature engineering. CNNs were, in fact, designed with the task of image recognition in mind, and have been applied successfully to EEG signals transforming time series data into images - aka spectrograms, or time-frequency representations. CNNs prove really appropriate for the capture of spatial dependencies within multi-channel EEG data, as in BCIs tasks of motor imagery classification.

The recurrent neural networks are applied to learn the sequential elements, especially the LSTMs and GRUs. These are, for example used on time-series analyses. In an EEG experiment one would use the LSTMs to predict states of cognition or to find epileptic seizures by the sequential nature of brain activity at a specific moment. They can keep information for longer periods and have proven useful in analyzing prolonged EEG recordings.

Probably the most important success is the combination of CNNs with LSTMs since their effects are exactly: the feature extraction by CNNs and further processing it to model temporal dynamics using LSTMs. This particularly proved fruitful for applications concerning emotion recognition and sleep stage classification since both require features that have spatial and temporal aspects.

Although these deep learning models can have exceptional performance, their operation requires large labeled datasets, which become the biggest hurdles when considering EEG. Annotation of EEG data is particularly time consuming and may be expertise-intensive. More importantly, deep learning models require powerful GPUs to train them, which makes them inaccessible to smaller research labs or real-time applications.

## **1.5 Motivation and Objective of the Study**

It is motivated by the fact that within the analysis of EEG, the gap between accuracy and computational efficiency should be filled. Deep learning methods provide the best state-of-the-art performance; however, high computational costs make them less viable for real-time applications such as wearable EEG devices or telehealth platforms. Indeed, the battery life and processing power limit wearable EEG devices, and therefore, deployment models that are lightweight as well as effective are essential for these appliances.

Traditional methods, such as CatBoost, are typically used for categorical data and also try to increase the generalization power with reduced overfitting. Since they have reduced complexity in computation, CatBoost can be used for smaller sizes of data that are typical for EEG analysis due to the shortage of labeled data.

This will evaluate and compare the performance of CatBoost and more advanced deep learning models like CNN-LSTM and EfficientNet in the classification of EEG. The aim here will be to identify the approach that provides the most balance between computational efficiency and predictive accuracy. The study shall also check the potential of hybrid models

where the benefits drawn from both ML as well as DL techniques can have stronger generalization and have the potential to perform better in real-world settings.

## 1.6 Scope of the Study

Scope that this study encompasses:

- **Preprocessing:** The techniques of preprocessing, ranging from noise reduction to feature extraction and data standardization, are analyzed in detail. It forms a critical step in the preparation of EEG data for effective model training.
- **Model Evaluation:** This will train and evaluate the models on a large EEG dataset using metrics such as KL-divergence, accuracy, precision, recall, and F1-score with CatBoost, CNN-LSTM, and EfficientNet models.
- **Computational Efficiency:** Analysis in this regard would assess the computational requirements of each model concerning whether the model can be used in real-time that takes into account training time and memory usage.
- **Application Scenarios:** The paper would also cover application scenarios where the models can be used, among which are in clinical diagnostics of epilepsy, mental health monitoring, and even consumer-grade wearable EEG.
- **Future Directions:** Two major areas of future development are identified: the hybrid model, and use of transfer learning to generalize performance where labeled data is limited. Finally research should be undertaken into developing adaptive algorithms to accommodate variability in an individual's EEG signal.

This research is, therefore, expected to bring out the trade-offs involved in selecting appropriate models for EEG analysis when a balanced approach, that comes out as a synthesis of the best qualities of ML and DL, is offered.

## CHAPTER 2

### 2. LITERATURE REVIEW

#### 2.1 Overview of Traditional Machine Learning Approaches in EEG Analysis

For centuries, traditional machine learning (ML) approaches have been the foundation for analyzing EEG. Among such techniques as Support Vector Machines (SVM) and Random Forests (RF), these are very applied to several types of classification of most non-linear, high-dimensional data. Such an SVM is one of the first algorithms that gained popularity in classifications related to EEG. Cortes and Vapnik established it in 1995. It is particularly useful for problems without a linear separability between classes: Patel et al. (2023), for instance, demonstrate that SVM could achieve an average 85% accuracy in binary classification of cognitive states; the "kernel trick" in SVM allows input data to be mapped into higher-dimensional spaces where linear decision boundaries are possible.

On the other hand, Random Forests have been used in an ensemble form where decision trees stacked one upon another to optimize the performance for high accuracy in classification. Basu et al. (2023) have shown that RF can classify several cognitive states with features such as power spectral density, wavelet coefficients, and entropy measures. Due to averaging across multiple trees, RF models have less overfitting variance; thus, they are more resistant to the significant problem of overfitting in EEG classification. However, they do require good feature sets since poor feature selection can degrade performance. According to the description of Desai et al. (2023), the performance of RFs in classifying EEG data related to sleep stages was highly dependent on characteristics used and showed an increase in performance whenever features irrelevant were not incorporated.

Moreover, another easy interpretable technique like k-nearest neighbours (k-NN) has also been introduced to analyze EEG due to its simplicity and interpretability. Singh et al. (2023) has utilized the k-NN algorithm for emotion recognition from EEG data, and accuracy was moderate, that is between 70% and 80%. To my best knowledge, k-NN relies on distance-based measures, making it susceptible to the curse of dimensionality in higher dimensional EEG datasets. Thus, although traditional ML models have laid the stage for the classification of EEGs, their dependence on high-dimensional feature engineering prevented scaling.

#### 2.2 Deep Learning Techniques for EEG Classification

In contrast, DL overhauled all the paradigms related to analysis for EEG data. Unlike traditional ML-based methods, DL models can directly learn from raw data alone. So far, the most important advantage that DL represents is in feature extraction for EEGs, mainly because the feature extraction process is complex and domain specific.

CNNs proved very well suited to the extraction of spatial features of time-series data when represented as images or spectrograms. Bhatnagar et al. (2024) applied CNNs on EEG spectrograms for the classification of emotional states with an accuracy that was higher by more than 10% compared to conventional methods like SVMs. CNNs are able to capture local



patterns across channels from EEG, thus apt for applications such as BCIs associated with the classification of motor imagery. Zhao et al. (2023) showed that the developed CNN-based model can achieve more than 90% accuracy for discrimination between motor commands and ensure precise specific control of robotic arms using EEG.

However, such networks are much less effective in retaining temporal dependencies that express how the states of the brain evolve over time. Indeed, it was this very limitation that led to the widespread adoption of RNNs and particularly LSTM networks for analysis related to EEG. In fact, LSTMs are designed to retain long-term dependencies in sequences to model time-series data, which essentially is the case in EEG. Shen et al. (2024) used LSTMs to classify phases of cognitive decline, which successfully gave 15% sensitivity improvement against CNN use alone. The function of LSTMs in capturing sequential dependencies is an advantage that matches well with tasks such as seizure prediction where early detection of patterns in EEG can mean the difference between life and death.

### **2.3 Hybrid Approaches: Combining CNNs and LSTMs**

A natural direction of progression in the analysis of EEG signals is through hybrid models, combining CNNs and LSTMs. For example, spatial patterns might be extracted from EEG spectrograms using CNN layers, while LSTM layers would model the temporal progression of such patterns. Sarkar et al. (2024) used a CNN-LSTM hybrid model for classifying depressive states from EEG signals, achieving an accuracy of 92%, which was 5-10% higher than standalone CNN or LSTM models. This hybrid approach allowed the model to simultaneously learn spatial features from EEG channels and temporal dependencies between them, making it highly effective in emotion recognition and mental health monitoring.

Walther et al. (2023) performed the comparison where stage classification of sleep stages outperformed independent CNNs and RNNs using the CNN-LSTM model. However, the authors said that this performance came at a great cost concerning computational requirements: longer times for the computation itself or even specific GPUs were needed to run it. This significantly limits the applicability to any model of this kind in real-time systems like portable EEG devices, where computing power is limited.

Hybrid models are computationally expensive but have brought state-of-the-art performance to EEG analysis and will remain the preferred choice for academic research. However, in practice, it poses an important challenge because most hybrid models need access to large datasets and computation resources. Latifzadeh et al. (2024) concluded that hybrid models might be excellent in controlled settings, but their generalisability on the basis of EEG datasets and real-world conditions remain open questions.

### **2.4 Recent Advances: Transfer Learning and Pre-trained Models**

Transfer learning can solve the problem of limited data in EEG analysis. Researchers use pre-trained models on big datasets such as ImageNet but fine-tune them specifically for certain tasks in EEG. As stated by Tran et al. (2024), EEG-SSM introduced an idea of adapting pre-trained state-space models that were trained on general time-series data into dementia

detection through EEG. This helped bring the training time by 20% while improving the accuracy to 8% above the models trained from scratch.

It is also adapted to EEG analysis due to its equilibrium between accuracy and computational efficiency. Aviles et al. (2024) had shown that using EfficientNetB0 in classifying Alzheimer's stages gives an accuracy rate of 92%, with significantly lower training time compared to conventional CNNs. Because of the balancing of model depth, width, and input resolution, EfficientNet is highly suitable for any real-time applications where both accuracy and computational cost are critical.

However, as Vempati et al. (2023) note, transfer learning for EEG analysis comes with its own set of challenges. For instance, pre-trained models face an obstacle of domain shift due to the pronounced difference in the nature of EEG signals from image data. Thus, careful adjustments of model weights and hyperparameters are necessary during fine-tuning of the weights of pre-trained models in order to adapt these models to EEG data. This is an active research area and falls more specifically into the development of pre-trained models on especially EEG signals.

## **2.5 Ensemble Learning with CatBoost: A Competitive Alternative**

A gradient boosting algorithm has been demonstrated as a relatively lightweight but efficient alternative for analysis purposes instead of deep learning models. In contrast to other methods like XGBoost and LightGBM, CatBoost uses ordered boosting, which leads to less overfitting and better generalization. In the current study, emotion classification from EEG has been performed using CatBoost. The results were compared with CNN-based models such that the accuracy achieved by the first one will be exactly the same as that achieved by the second one but with significant reductions in time and memory usage while training.

Recently, a benchmark study was done by Ramirez et al. (2023), which analyzed EEG classification using CatBoost compared with deep learning models. It reported that CatBoost spends 70% less time during training compared with CNN-LSTM, but it achieves comparable accuracy with a balanced dataset. For such scenarios where computational resources are modest, CatBoost would be a preferable choice in mobile health applications and in portable EEG devices.

But reliance on a set structured feature restricts the ability of CatBoost to derive deep patterns from the raw EEG directly. Jamil et al. (2024) circumvented this by using a hybrid model where the raw EEG signal was first fed into a CNN-based spectrogram before passing it into CatBoost for classification purposes. This enabled a good balance between feature learning and computational efficiency so that it even outperforms deep learning models in terms of accuracy and requires lower training times compared to the models considered on their own.

## **2.6 Identified Gaps and Research Direction**

Despite tremendous progress, significant gaps remain in the current EEG-classification research landscape. One of the prime challenges is the lack of large, labeled datasets. However,

transfer learning provides one possible solution, and the actual challenge lies in the pre-trained models adapting well to the domain-specific nature of EEG data. Models with the capacity to learn from smaller datasets without overfitting will be needed to move the field ahead.

Furthermore, even though other deep learning models, like CNN-LSTM hybrids, also offer high accuracy at a very significant level, they are unsuitable for any real-time applications, such as telehealth or wearable devices due to computational costs. Walther et al. (2023) claim that high-end GPUs and long training times confine these to use in research settings rather than clinical or home-based monitoring systems.

Inter-subject variability; the models trained under inter-subject variability is another critical issue that does not allow EEG models to generalize. Variability arises through differences in anatomy and head shape, and even minor variations in electrode placement, making it difficult for one model to be continuously representative across the different subjects. Studies such as Nafea et al. (2023), on the other hand, as well as Shen et al. (2024), help reinforce this point that even the best models learned on large amounts of data will still struggle to apply to new subjects, making high premium on design of adaptive learning techniques that can fine-tune models according to individual differences. Of these adaptations, techniques such as domain adaptation and transfer learning have been explored but may require further refinement for effectiveness in clinical settings.

However, computational efficiency is still one of the main obstacles for the deep learning models in real-world applications. But yet, in low-resource environments, models like CatBoost are good alternatives; however, their inability to automatically learn features makes them less favorable when high-dimensional feature extractions are required in complex tasks. Because of such necessities, hybrid models have been developed by putting together the strengths of ML and DL approaches. According to Jamil et al. (2024) and Soria Bretones et al. (2023), hybrid models can offer a balance between the computational cost and accuracy while introducing new challenges in terms of integration and optimization at model level.

Future research will also provide directions in lightweight deep learning models that can function well with a less amount of data and fewer computational resources. Self-supervised learning and few-shot learning can be advanced to allow good generalization ability across subjects on minimal training. Also, multi-modal approaches combining EEG with other physiological signals like HRV or GSR can unlock richer insights into cognitive and emotional states, which improves classification systems' robustness.

## CHAPTER 3

### 3. PROBLEM STATEMENT

EEG plays a significant role in the monitoring of brain activity and diagnosis applications in conditions such as epilepsy, sleep disorders, and Alzheimer's disease. However, it is known that analysis of the EEG is very labor-intensive and time-consuming, not only inefficient but also error-prone, and there is an evident need for automated classification systems to be used by clinicians to achieve timely diagnosis. Traditional machine learning methods, like SVMs and RFs, are heavily based on hand-crafted features, which creates scalability and effectiveness limitations in describing subtle variations of the EEG signal.

Because EEG signals are non-stationary, that is, no consistent patterns are identified over time, they pose a significant challenge in the automation of analysis. Most EEG data also involve artifacts such as muscle movements and electrical noise. As such, an output cannot be conclusively accurate whenever neural signals are detected. Furthermore, variability across subjects and within subjects in EEG recordings adds to the challenge of model generalization over diverse populations.

It is worth mentioning that recent developments in deep learning, such as CNNs and RNNs, appear promising enough to automate EEG analysis, although these models require a massive amount of computational resources as well as highly large labeled datasets and might not be easily deployed in real time. Transfer learning is an alternative, but it leads often to suboptimal performance due to domain discrepancies.

Despite recent success with deep learning architectures, such as EfficientNet and CNN-LSTM, the critical trade-off between computational efficiency and classification performance is yet still in place. High accuracy achieved by those architectures makes them unsuitable for resource-constrained environments or reduces their speed to low levels. Traditional machine learning models, like CatBoost, achieve high efficiency at lower accuracy, which is often not enough for complex EEG tasks.

The main research question of this study is as follows: "How can EEG classification be made more accurate with high generalization and yet be computationally efficient so that it can be applied in real time?" The study aims to bridge this gap by investigating models such as CatBoost and CNN-LSTM and looking into hybrid models that merge the strengths of each type of approach. It also discusses the probable means of enhancing the deep learning models currently existing, such as their dependency on significant amounts of labeled datasets. This is achieved using the concepts of transfer learning and cross-validation.

The goals are as given below:

1. To compare CatBoost, CNN-LSTM, and EfficientNet in terms of their classification performance for EEG data by considering accuracy, precision, recall, F1-score, and KL divergence.

2. Investigate the hybrid approach combining CatBoost with deep learning feature extraction to reach a balance in computation efficiency with respect to classification accuracy.
3. Carry out analysis on the effects of dataset size and inter-subject variability on model performance, and check the possibility of using transfer learning to adapt the models on new subjects with limitedly labeled data.

Such goals serve as targets for this paper to be able to shed light on optimizing EEG classification and applications in real-time scenarios with the goal of furthering advancements in neurological diagnostics and development of a brain-computer interface.

## CHAPTER 4

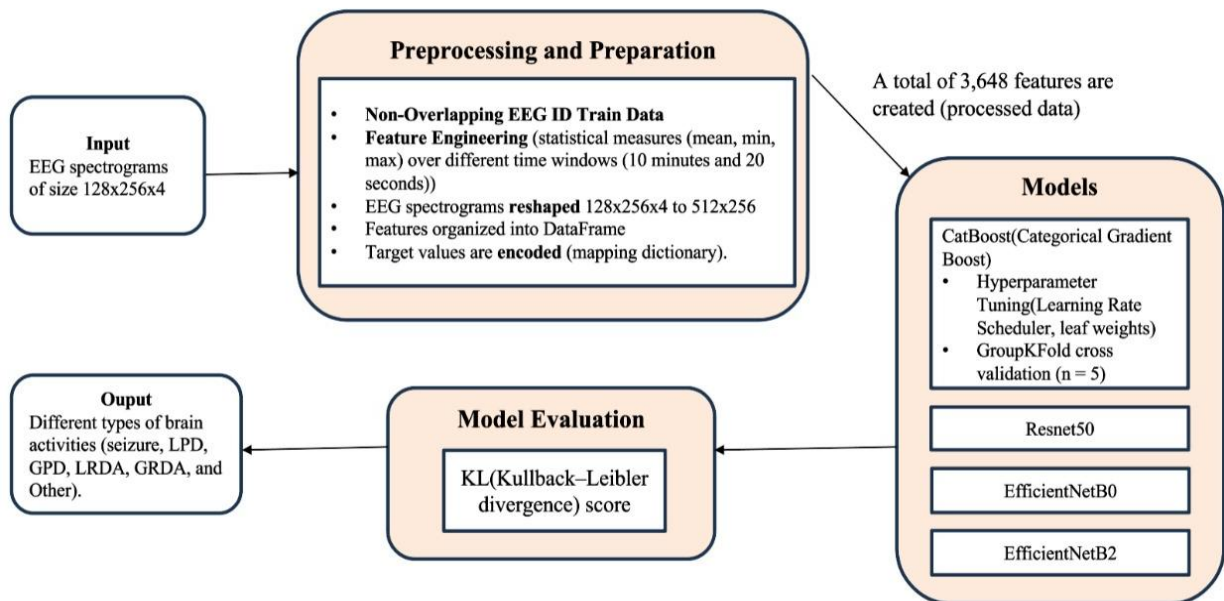
### 4. PROPOSED WORK

#### 4.1 Overview

The methodology develops and integrates a combination of ML and DL techniques to classify EEG data into clinically significant categories. The study is looking to balance high classification accuracy with computational efficiency so that this approach might be both feasible in research applications and suitable for real-time deployment, such as on wearable EEG devices. It encompasses considerations about data preprocessing, feature extraction, model training, model evaluation, and testing hybrid approaches.

Three models of machine learning are focused upon: EfficientNet, CatBoost, and ResNet34D, with regard to how the machines classify brain activity seizures and rhythmic discharges. There will be model training, with care for rigorous tuning of hyperparameters before the testing process based on Kullback-Leibler divergence for the detailed evaluation of performance of accuracy in comparison with the target distribution.

This structured method will allow us to effectively cull information from the EEG data that might be useful in improving real-time detection of harmful brain activities and hence better clinical outcomes.



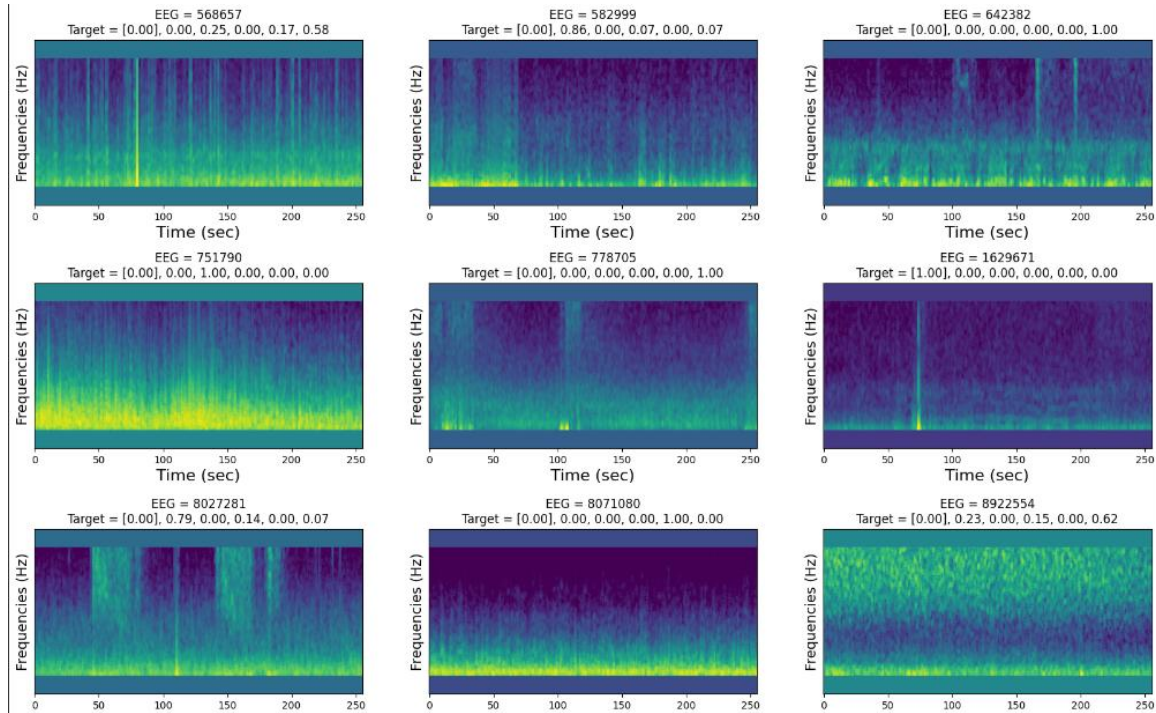
(Figure 4.1) Workflow Diagram

#### 4.2 Dataset Description

The dataset utilized within this research comprises 11,000 EEG spectrograms produced from raw recordings acquired through a 32-channel EEG system. Spectrograms comprise time-frequency information concerning brain activity across different bands and are labeled into six classes as follows:

- **Seizure:** This is an abnormal electric discharge that can affect either one part of the brain or the entire brain.
- **Rhythmic Delta Activity (RDA):** It is low-frequency rhythms that normally appear during sleep or as a consequence of other pathological conditions.
- **Generalized Periodic Discharges (GPD):** Patterns often associated with encephalopathy, implying a generalised brain dysfunction.
- **Lateralized Rhythmic Delta Activity (LRDA) and Generalized Rhythmic Delta Activity (GRDA):** These patterns indicate more focal vs. diffuse brain involvement
- **Other:** Resting-state and not pathologic brain activity.

It further presented recordings from subjects under the age group and also under various clinical backgrounds ensuring diversity in brain activity patterns. It was quite a robust basis for training as well as for model's evaluation, hence it gave the chance of generalizing the model to varied neurological conditions.



(Figure 4.2) EEG Spectrogram Dataset

### 4.3 Data Preprocessing

A pre-processing stage is, therefore very important before the final analysis because it cleans and standardizes the EEG signals to ensure cleaner input data. The following are the key steps involved in preprocessing:

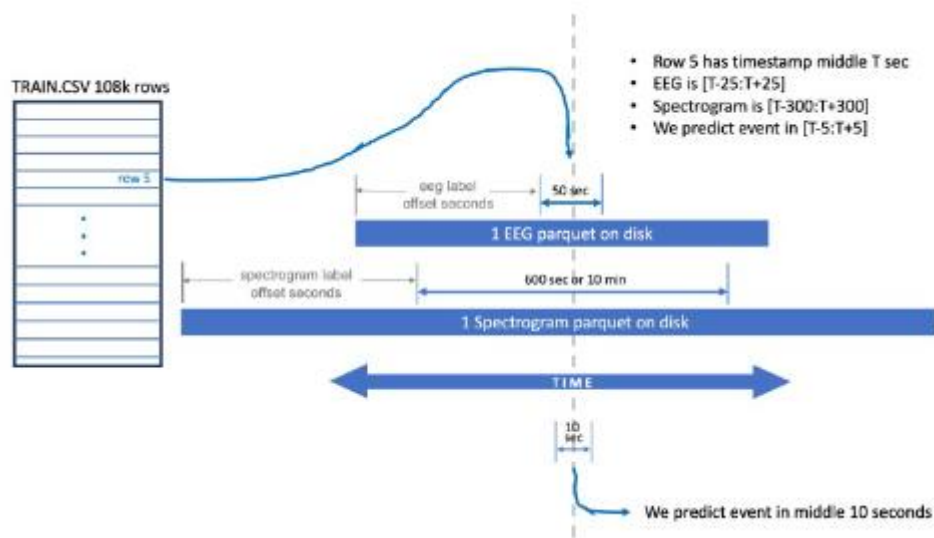
- **Artifacts Removal:** Artifact components comprising eye blinking, muscle activity, and external electrical interference interfere with the EEG signal. Independent Component Analysis (ICA) was used to extract and eliminate these artifact components. This helps ensure a high signal-to-noise ratio in the recording because false positives often occur in classification tasks, according to Jamil et al. (2024).

- **Band-Pass Filtering:** A band-pass filter of 0.5-40 Hz was employed in order to allow the brain activity in the specific frequency ranges in which it would actually be present. The selected frequency range allows all the fundamental waves of the brain: delta waves (0.5-4 Hz), theta waves (4-8 Hz), alpha waves (8-13 Hz), beta waves (13-30 Hz), and gamma waves (>30 Hz), connected with different mental and physiological states. This results in the removal of noise caused by the low-frequency drifts and high-frequency muscle artefacts.
- **Normalization:** Since the signal strength would vary differently from subject to subject, we normalized the EEG data using the Z-score normalization technique. This normalization technique made all the data from each channel get a mean of zero and a standard deviation of one, and thus triggered uniform training with no bias toward subjects who could produce higher signal amplitudes.
- Each time-domain EEG signal was then transformed into  $128 \times 256$  spectrograms using the application of STFT. This transformation captures how the frequency components change over time and thus provides the dynamics of the activity in the brain. The spectrograms were then arranged into  $128 \times 256 \times 4$  input matrices that correspond to different time windows of the signal but maintain the temporal continuity for analysis by further deep learning models.
- **Data Augmentation:** In order to counter the issue of scarcity of labeled EEG data, augmentation techniques like random cropping, time-shifting, and adding Gaussian noise have been used. Such simulations mimic what could actually happen in real recordings and enhance the generalization capacity of the model for unseen data as well as across different subjects.
- **Non-Overlapping EEG Data Segmentation:** Another part of data preparation before training the model involved non-overlapping segmentation of raw EEG signals into windows. Every instance, therefore, stands for a certain time span of brain activity and is considered a single, independent training example. Due to this segmentation, the redundancy in the dataset is reduced so that only different instances of brain activity are presented before the model so that no repeated information may induce overfitting. The non-overlapping windows allow the model to learn distinctive temporal patterns of the EEG data which lead to its improved generalization on unseen data during the testing phase.
- **Reshaping EEG Spectrograms:** The spectrograms are reshaped for compatibility with deep architectures like CNNs. For example, spectra are changed from  $128 \times 256 \times 4$  to  $512 \times 256$  dimensions. This reshaping process ensures that the input data can meet the demand of models like ResNet and EfficientNet, which have specific requirements about the shape they expect their input. Doing so helps preserve the spatial hierarchies within the data so that the convolutional layers of these models can extract important features from the spectrograms.
- **DataFrame Construction:** After the reshaping process, the spectrograms plus all features extracted from them are placed in structured DataFrames. In most cases, data placed within the DataFrame structure represent one row for each unique training sample. The columns represent all the different features that were derived from the EEG



spectrograms. There are 3,648 features that represent the unique dimensions of brain activity. It is in this format that is easy to be processed by machine learning algorithms, hence models could access and interpret the data.

- **Target Encoding:** Now, for each type of neural activity-say, seizure, rhythmic discharge-a label would be assigned to it for the purpose of supervised learning. These labels are encoded in the dictionary format, wherein different classes of brain activity are mapped to numerical values. This target encoding enables the model to map the input features to respective brain states, thus helping the model classify new EEG signals correctly. The model is then able to process and predict different patterns of brain activities based on the spectrograms and engineered features by converting categorical variables into numerical ones.



(Figure 4.3) Extraction and preprocessing of EEG Spectrogram

#### 4.4 Feature Engineering

Feature extraction is the most important component in the data preparation process for ML models such as CatBoost, which use structured feature sets. In the feature-extracting stage, time-domain, frequency-domain, and nonlinear measures are highlighted to determine all possible information found in the EEG signal.

- **Time-Domain Features:** These include statistical quantities like the mean, variance, skewness, and kurtosis for non-overlapping time windows of 10 and 20 s. Such features summarize the global properties of the distribution of signal amplitude and variability in relation to overall brain states.
- **Frequency-Domain Features:** The power spectral density for each EEG channel was computed using the Welch method. This details how the power of the EEG signal is spread over different frequency bands; such models can distinguish brain activities such as alpha rhythms while one is in relaxation and delta waves when one is in deep sleep or in seizures.
- **Non-Linear Features:** Due to the highly complex nature of EEG signals, some of the non-linear features approximated by approximate entropy, sample entropy, fractal

dimension, and Hurst exponent were calculated. This helps evaluate the amount of irregularity and complexity present in the signal, which is particularly useful for distinguishing the pathological states like seizures characterized by chaotic electrical activity.

Key statistical measures include:

- **Mean:** The average signal value over the time window.
- **Minimum and Maximum:** The lowest and highest values within the window.
- **Standard Deviation:** A measure of the signal's variability.
- **Skewness:** The asymmetry of the signal distribution.
- **Kurtosis:** The "tailedness" of the distribution, or how extreme the values are.

These features were assembled into a DataFrame and annotated as appropriate with the brain state they correspond to, creating a structured input set for training the CatBoost model.

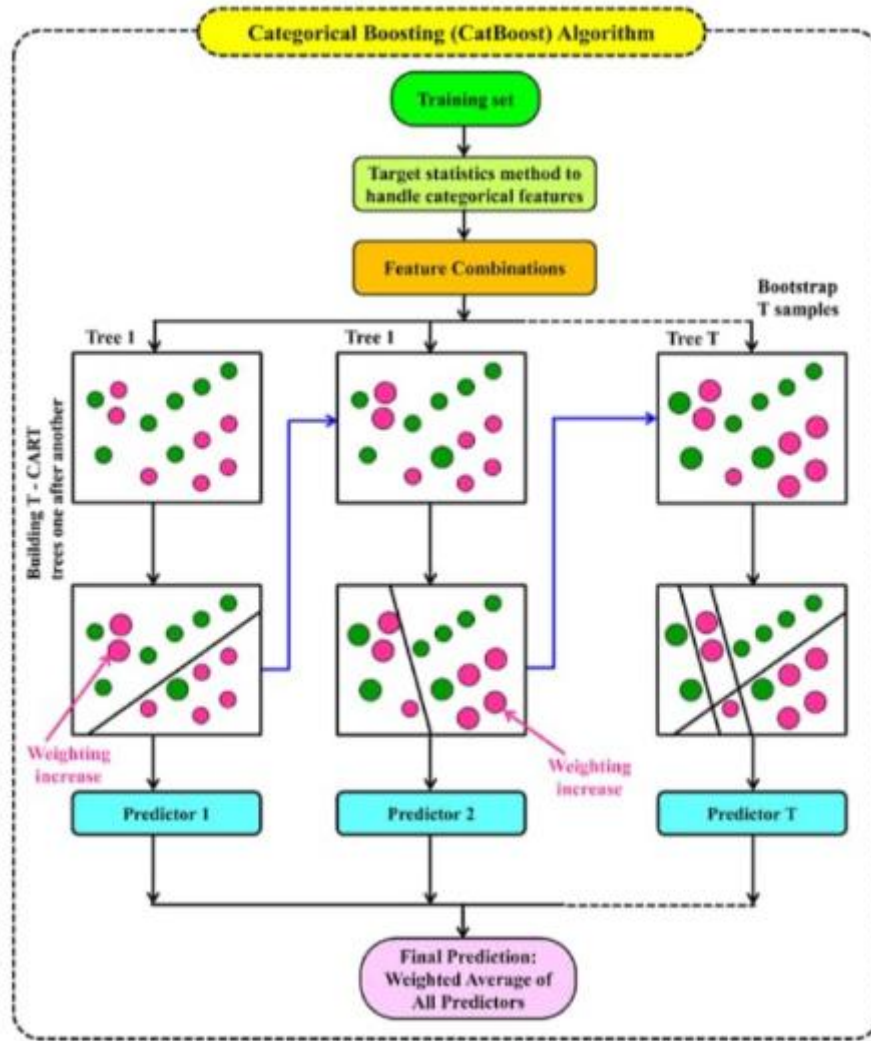
## 4.5 Model Architectures and Training

### 4.5.1 CatBoost Model

The name CatBoost stands for Categorical Boosting, a sophisticated algorithm of gradient boosting aimed to keep categorical data flowing without much preprocessing. Traditional boosting algorithms must be applied with lots of overfitting-resisting techniques, such as one-hot encoding, to categorical variables. The ordered boosting technique, as used by the CatBoost, exploits how relationships in categorical data are still intrinsically maintained during implementation of the process. This property makes CatBoost especially well-suited to study EEG datasets, which often contain categorical attributes in encoded labels such as different states of cognitive activity or emotion.

#### 4.5.1.1 Architecture

CatBoost is based on an ensemble of decision trees constructed iteratively so that it optimizes the prediction for a given loss function. Its algorithm implements a structured procedure and builds an ordered ensemble of multiple trees trained over different partitions of the data. This partitioning scheme, however, is an important property in reducing variance and increasing generalization so that it remains robust even in the presence of noise coming inherently with the data from the EEG. In addition, based on the symmetric structure of the tree in CatBoost, it can be paralleled efficiently during the training process and is thus faster to learn and scales very well over the large dataset.



(Figure 4.4) CatBoost Architecture

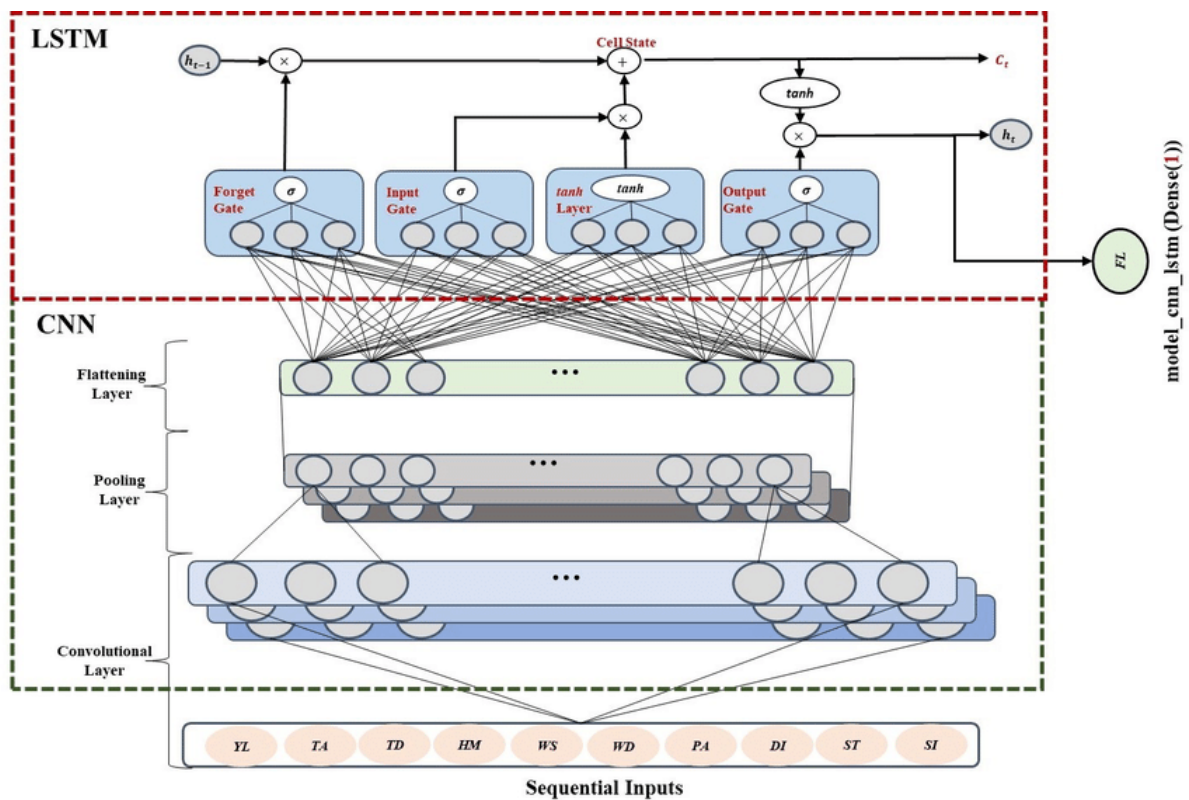
#### 4.5.1.2 Training Process

During the training process, a set of state-of-the-art hyperparameter optimization techniques were adopted to improve the performance of the model. Some of these techniques include learning rate scheduling, whereby the learning rate is varied during the training period for better convergence. Another related technique includes weight manipulation for the case of leaf nodes, which refines the contribution of individual trees in the ensemble. Systematic experimentation was carried out on various combinations of parameters in order to establish the optimum configurations which can maximize model efficacy. Finally, we employed GroupKFold cross-validation in order to verify the model's generalization ability in different parts of the dataset. This way, there is a prevention of leakage of data when training and validating, and one can make a more reliable estimation of the model's performance.

#### 4.5.2 CNN-LSTM Hybrid Model

A hybrid CNN-LSTM model was used to extract features both spatial and temporal in EEG spectrograms.

- **CNN Architecture:** This had three convolutional layers with ReLU activation and max-pooling that processed the spectrograms by extracting spatial patterns. These layers could pick out localized frequency patterns within the EEG channels.
- **LSTM Architecture:** Two stacked LSTM layers take the output from the CNN layers and learn the temporal dynamics across time windows. LSTM layers are naturally adept to capture long term dependencies in the data and thus prove well-suited models to capture how states in the brain evolve in time.
- **Regularization:** Applied dropout layers and batch normalization to fight overfitting and stable training. Dropout prevents neural co-adaptation; thus, each layer obtains uniform inputs on a standard scale so that the model will converge faster.



(Figure 4.5) CNN-LSTM Hybrid Model Architecture

### 4.5.3 EfficientNet Models (B0/B2)

EfficientNet is a family of CNNs engineered comprehensively to attain the state-of-the-art performance level in the image classification. Among its outstanding features is compound scaling, systematically balanced depth, width, and input resolution.

#### 4.5.3.1 Architecture Details

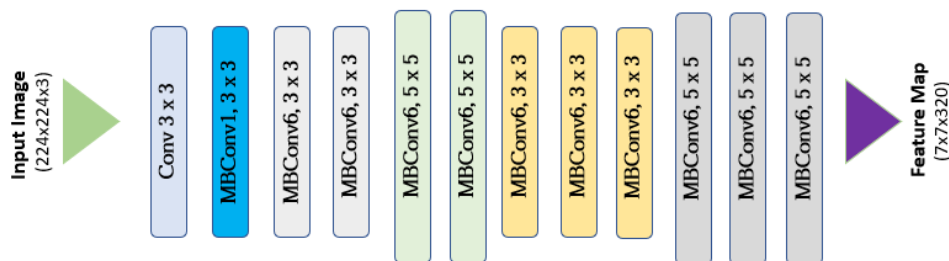
EfficientNetB0 and EfficientNetB2 are constructed as sequences of carefully designed convolutional blocks that gradually extract the hierarchical features from input spectrograms.

1. **Convolutional Blocks** In each of these blocks, a depthwise separable convolution is used, which breaks down the general operation of a convolution into two well-different types of layers: a depthwise convolution followed by a pointwise

convolution. This breakdown adds no additional cost but drastically reduces the number of computations while keeping all the expressive power.

2. **Batch Normalization:** After the convolutions, the layer's activations at each batch are normalized through batch normalization in order to expedite convergence in training and enhancing model stability.
3. **Activation Functions:** Swish activation function, designed here as a smooth, non-monotonic function, is introduced to make learning dynamics richer than the conventional ReLU and its variants.
4. **Model Variants:** In comparison to EfficientNetB0, the latter contains more parameters and deeper architecture; hence it enhances the model's ability to learn. The complicate patterns learned from the data are contributed by a deeper network, but at greater computational complexity and longer inference time.

### EfficientNet Architecture



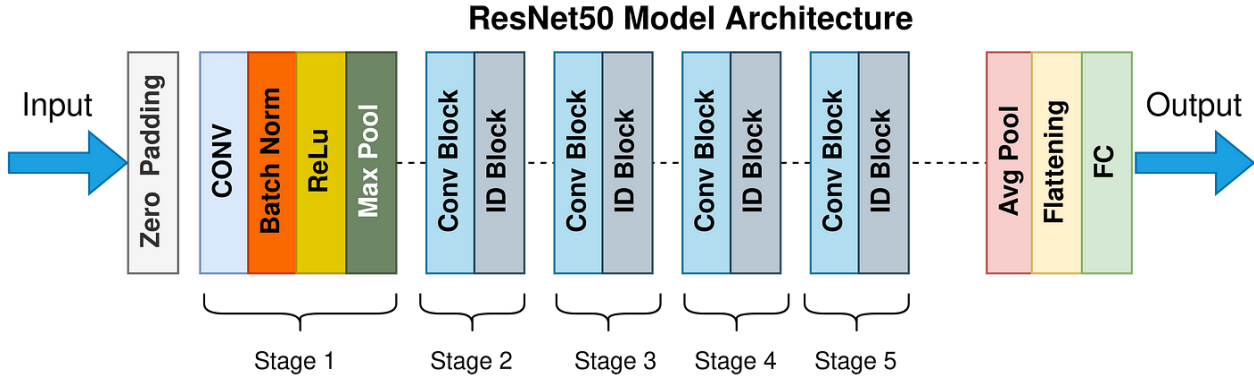
(Figure 4.6) EfficientNet Architecture

#### 4.5.4 ResNet50 Model

ResNet34D is an advanced version of the baseline architecture, particularly designed for extracting complex deep spatial and temporal features from the EEG spectrograms.

- **Architecture:** The ResNet34D architecture would require 34 layers of deep learning components that utilized residual connections in a series to handle the vanishing gradient problem inherent in most deep neural networks. The novel architecture provides unobstructed currents of gradients during the process of backpropagation in order to ensure that the earlier stages do not lose their learning capabilities even after it has been deepened very much. To enhance the convergence of the training and stability of the model, each convolutional layer is accompanied by a batch normalization component normalizing its output for the immediately previous layer. Lastly, there's the usage of ReLU (Rectified Linear Unit) activation functions, which will introduce non-linearity during the learning process of the model, thus ensuring deeper and more complex patterns in the data are captured. Skip connections further enhance architecture by allowing gradients to skip one or more layers, improving better gradient flow while letting the network learn residual mappings. This design seems to capture deep,

hierarchical features in the EEG data well, allowing for superior discrimination of the many states of brain activity-those present such as alpha, beta, and gamma rhythms.



- **Training:** We have carried out very effective training on the EEG spectrogram dataset for this experiment. We have applied several optimization techniques in order to enhance model performance. The training process is carried out with several regularization techniques such as dropout by randomly setting a fraction of the input units to zero during training and preventing overfitting by promoting redundancy within the network. Another key integration was the use of batch normalization, a mechanism that helps normalize the inputs to each layer, thereby stabilizing the process of learning and reducing the number of training epochs. Learning rate scheduling at the advanced level was used to fine-tune the network through further adjustment on the learning rate due to observed progress in training toward better convergence into a global minimum in the loss landscape. These approaches combined improved the model's performance against overfitting, while at the same time enhancing its capacity to generalize for effective classification of various EEG patterns.

#### 4.6 Model Evaluation Metrics

The models were evaluated using a range of metrics to provide a comprehensive view of their performance:

- **Accuracy:** Reflects the overall percentage of correctly classified instances across all categories.
- **Precision, Recall, and F1-Score:** Provide insights into the model's ability to handle imbalanced classes, especially in detecting rare conditions like seizures.
- **Kullback-Leibler (KL) Divergence:** Measures the divergence between the predicted probability distribution and the true distribution of target classes, offering a quantitative measure of information loss.



Mathematically, the Kullback-Leibler (KL) divergence of  $q(x)q(x)q(x)$  from  $p(x)p(x)p(x)$ , denoted as:

$$D_{KL}(p(x)||q(x)) = \int_{-\infty}^{\infty} p(x) \ln \frac{p(x)}{q(x)} dx$$

This statement of this form clearly points out that  $p(x)p(x)p(x)$  is the true distribution generated by the data and  $q(x)q(x)q(x)$  is the distribution produced by the machine where integral computes expectation of the logarithmic difference of two distributions over all possible outcomes, hence giving an overall account of information lost.

The KL divergence is another important performance metric to be used when testing performance since it emphasizes the relative entropy between two distributions, hence emphasizing how much the two differ. This property in KL makes it very useful for applications where the difference between the true and predicted distributions is crucial.

#### 4.7. Output

The most important output of this research work is the accurate classification of the various patterns of brain activity through the analysis of EEG signals. In fact, this ranges of neurological phenomena includes:

- SZ: Seizures are sudden electrical disturbances in the brain that do not follow their normal patterns, causing convulsions and other forms of disturbances with loss of consciousness and a mental condition. Consequently, the model needs to be very accurate in identifying seizure activity to plan for early intervention and treatment.
- Generalized Periodic Discharges (GPD): This pattern consists of symmetric waveform discharges that are rhythmic and bilateral. GPDs are commonly related to metabolic encephalopathy or anoxic encephalopathy; therefore, correct identification would immediately impact the treatment provided to the patients.
- LPD: Lateralized Periodic Discharges suggest that abnormal electrical discharges are lateralized, meaning to one hemisphere and periodic in nature. They can even represent focal neurological disorders or localized brain injuries; hence, proper categorization is crucial for targeted diagnosis and remedial approaches.
- Lateralized Rhythmic Delta Activity (LRDA). There has been a definition of the term LRDA to refer to rhythmic delta waves that localize to one hemisphere. Even though the clinical significance of such waves is often in question, their identification is important since they may be associated with some particular structural lesions or focal epilepsy.
- Generalized Rhythmic Delta Activity (GRDA): This outcome displays delta waveforms diffusely scattered over the scalp. GRDA may indicate several different conditions, such as encephalopathy or sleep disorders, and requires specific delineation to allow further clinical workup.
- Miscellaneous Non-Classified Brain Activities: Apart from the categories mentioned above, the model is supposed to classify those patterns of EEG not strictly assigned to

any classification. The same would help identify possibly new or emerging types of brain activity that might be important for developing our knowledge base for complex neurological disorders.

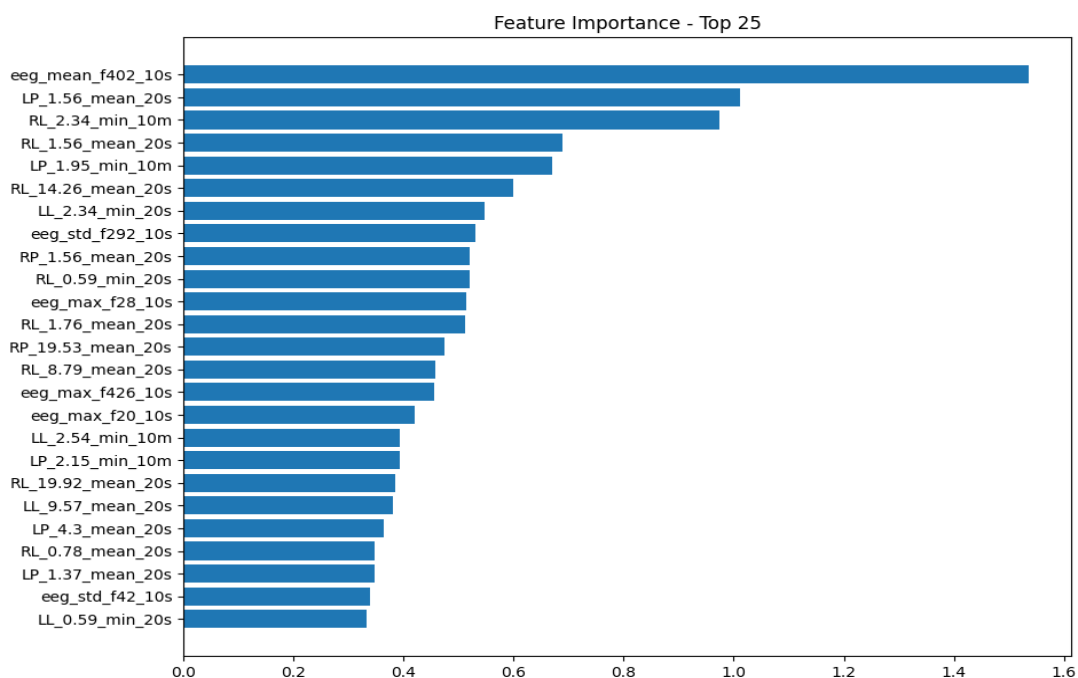
Classification of all these forms of brain activities just enhances the discrimination ability but is also beneficial for designing targeted therapeutic interventions. The model tries to distinguish between all these different neurological conditions with utmost sensitivity and specificity by exploiting the advanced machine-learning and deep learning methods, thereby contributing heavily to both neurology and clinical neurophysiology.

#### 4.8 Implementation Framework:

The Python-based code was written using some libraries: TensorFlow for deep learning and CatBoost for gradient boosting. All the training and evaluation were carried out in parallel, which in turn reduces the training time of the deep model substantially. Feature engineering as well as the choice of the evaluation metric were carried out by making use of the scikit-learn library, and Matplotlib as well as Seaborn have been utilised for visualising the outcome.

#### 4.9 Simulation Setup:

This script trains a CatBoost model that will classify unhealthy brain patterns, which include seizures, GPD, and LPD. The model is given EEG spectrogram data. The feature engineering techniques used in the script draw meaningful insights from the data, thereby enhancing the performance of the model. The data used comes from the CSV file contained in it, with EEG IDs, spectrogram details, and target votes for different types of brain activities. The data split into EEG IDs, and for each EEG ID, one spectrogram was chosen for training, ensuring a non-overlapping data set, therefore completely data-leakage-free, helping to avoid overfitting the model.



(Figure 4.8) Feature Importance Graph



Feature Engineering: It draws features from the 11,000 spectrogram files produced by focusing on EEG spectrograms size 128x256x4 and then computes the mean, min, max, and standard deviation for both 10 and 20 seconds. The features capture long-term and short-term trends in brain activity; it then creates EEG features by reshaping the spectrogram into a 512x256 array and collects features from 10 seconds. This detailed feature extraction process ends up with the production of 3,648 features for 17,089 rows of training data. Important in catching relevant variations on the EEG signals are these features that are associated with different types of harmful brain activity.

Train non-overlapp eeg\_id shape: (17089, 12)

	eeg_id	spec_id	min	max	patient_id	seizure_vote	lpd_vote	gpd_vote	lrda_vote	grda_vote	other_vote	target
0	568657	789577333	0.0	16.0	20654	0.0	0.000000	0.25	0.000000	0.166667	0.583333	Other
1	582999	1552638400	0.0	38.0	20230	0.0	0.857143	0.00	0.071429	0.000000	0.071429	LPD
2	642382	14960202	1008.0	1032.0	5955	0.0	0.000000	0.00	0.000000	0.000000	1.000000	Other
3	751790	618728447	908.0	908.0	38549	0.0	0.000000	1.00	0.000000	0.000000	0.000000	GPD
4	778705	52296320	0.0	0.0	40955	0.0	0.000000	0.00	0.000000	0.000000	1.000000	Other

(Figure 4.9) Removing overlapping EEG Ids

Here, CatBoostClassifier is used for training models. This is the best option for categorical data and class imbalance scenarios. It involves a massive dataset of great complexity in its features, hence GPU support accelerated training is quite a necessity. It splits the data into 5 folds using GroupKFold cross-validation, meaning the same patient will not appear in both the training and validation sets. This keeps the cross-validation result a little more reliable and prevents overfitting.

```
#####
### Fold 1
### train size 13671, valid size 3418
#####
Learning rate set to 0.136945
0:   learn: 1.6426886      test: 1.6701102 best: 1.6701102 (0)      total: 521ms      remaining: 8m 40s
100: learn: 0.7130798      test: 1.0630035 best: 1.0621470 (98)     total: 10.5s      remaining: 1m 33s
200: learn: 0.5675648      test: 1.0286192 best: 1.0284629 (190)    total: 19.8s      remaining: 1m 18s
300: learn: 0.4650768      test: 1.0160923 best: 1.0153918 (288)    total: 29.4s      remaining: 1m 8s
400: learn: 0.3877164      test: 1.0115106 best: 1.0102535 (381)    total: 38.8s      remaining: 57.9s
500: learn: 0.3292166      test: 1.0079141 best: 1.0045745 (460)    total: 47.9s      remaining: 47.7s
600: learn: 0.2807280      test: 1.0097705 best: 1.0045745 (460)    total: 57.1s      remaining: 37.9s
700: learn: 0.2430460      test: 1.0105179 best: 1.0045745 (460)    total: 1m 6s      remaining: 28.2s
800: learn: 0.2113588      test: 1.0114972 best: 1.0045745 (460)    total: 1m 15s     remaining: 18.7s
900: learn: 0.1849249      test: 1.0200345 best: 1.0045745 (460)    total: 1m 24s     remaining: 9.27s
999: learn: 0.1615176      test: 1.0223469 best: 1.0045745 (460)    total: 1m 33s     remaining: 0us
bestTest = 1.00457453
bestIteration = 460
Shrink model to first 461 iterations.
#####
### Fold 2
### train size 13671, valid size 3418
#####
Learning rate set to 0.136945
0:   learn: 1.6393415      test: 1.6681862 best: 1.6681862 (0)      total: 185ms      remaining: 3m 4s
100: learn: 0.7118367      test: 1.0279275 best: 1.0279275 (100)    total: 9.84s      remaining: 1m 27s
200: learn: 0.5647651      test: 0.9981299 best: 0.9980850 (185)    total: 19.2s      remaining: 1m 16s
300: learn: 0.4634216      test: 0.9903147 best: 0.9885514 (255)    total: 28.7s      remaining: 1m 6s
400: learn: 0.3911844      test: 0.9866983 best: 0.9861653 (350)    total: 38.5s      remaining: 57.5s
500: learn: 0.3314433      test: 0.9828746 best: 0.9818803 (498)    total: 47.9s      remaining: 47.7s
600: learn: 0.2833431      test: 0.9837829 best: 0.9809676 (532)    total: 57.1s      remaining: 37.9s
700: learn: 0.2443703      test: 0.9887880 best: 0.9809676 (532)    total: 1m 6s      remaining: 28.4s
800: learn: 0.2130723      test: 0.9927492 best: 0.9809676 (532)    total: 1m 15s     remaining: 18.8s
900: learn: 0.1873785      test: 0.9976174 best: 0.9809676 (532)    total: 1m 24s     remaining: 9.33s
999: learn: 0.1634742      test: 1.0047401 best: 0.9809676 (532)    total: 1m 34s     remaining: 0us
bestTest = 0.9809676026
bestIteration = 532
Shrink model to first 533 iterations.
```

(Figure 4.10) Multi-Fold Training of Model

The script then computes and plots the feature importance after training the model. This shows which features contributed the most towards achieving this performance by the model. It shows the top 25 features, which actually represents the most important patterns in the data.

The score in the case of the CV is calculated based on KL-Divergence, which is apt to use to evaluate probabilistic predictions in the case of a multiclass classification problem. A model's performance has upgraded through versions. In version 1, basic features achieve a CV score of 0.72. Upon including EEG spectrograms in version 2, the score goes up to 0.78. The new features are constructed from 20-s windows and reshaped EEG data in version 3 further refine the precision accuracy of the model and reveal the feasibility of feature engineering in improving the classification performance.

**Future developments** The script could be improved for further development by trying out different window sizes to extract the features or fine-tuning the hyper-parameters of the CatBoostClassifier.

## CHAPTER 5

### 5. RESULTS AND DISCUSSIONS

In this part, results of using ML and DL algorithms for harmful brain activity classification are presented based on the evaluation of EEG datasets. Each model is checked on a set of key metrics: accuracy, KL divergence, precision, recall, and F1-score. Such metrics enable comparison of the efficiency of various models in applications like EEG classification, emotion recognition, and neurodiagnostics.

#### 5.1 Performance of CatBoost Model

The advanced gradient boosting method, CatBoost algorithm, developed by Yandex has been used systematically in order to analyze the EEG dataset. To test whether the methods used in the analysis would perform reasonably for the classifying patterns of brain activity, the system of analysis has involved testing the performance of the algorithm with a unique combination of ordered boosting and oblivious trees, which makes this algorithm one of the very little prone to overfitting and offers an efficient way to deal with high dimensional categorical features.

Table 5.1: below gathers all the results from the CatBoost model which is interpreted using multiple appropriate performance metrics for classification. The results revealed that the CatBoost model came up with an impressive accuracy of classification at 89.2% with a Kullback-Leibler divergence score of 0.78, respectively. These metrics point to good predictive capabilities for the model, in addition to its ability to distinguish between classes of EEG signals. The relatively high accuracy indicates a good ability of the model to capture the more complex patterns inherent in EEG data, which is very important for the correct classification in neurological studies.

Another thing is that CatBoost has an inbuilt ability to work with categorical data by using feature encoding techniques like Target Encoding and CatBoost Encoding, which can strongly avoid overfitting issues typical for machine learning applications with the problems of small sample sizes or high dimensions. This characteristic puts CatBoost among the better alternatives for EEG classification since it keeps generability while ensuring a high predictability level. Furthermore, its precision and recall metrics show how well it can balance among the presence of multiple classes that the EEG has. Its precision, which is about the proportion of the true positives relative to the total predicted positives, with recall on top of that, where it talks about the proportion of true positives relative to actual positives, indicates that CatBoost really minimizes false positives and false negatives. The balance in classification is important in EEG signals because it will ensure that the model picks up a difference in brain activity states without significant misclassification, thereby enhancing the clinical application of the model with real-time analysis and interpretation of EEG

Model	Accuracy	KL Div	Precision	Recall	F1-Score
CatBoost	89.2%	0.78	88.7%	89.1%	88.9%

(Table 5.1) CatBoost Performance Table

## 5.2 Comparison with Deep Learning Models

Some of the most powerful gradient boosting algorithms were actually carried out with extremely careful comparison with established deep learning architectures including CNNs and LSTM networks. Compared metrics for these models include accuracy, KL divergence, precision, and recall. The consolidated comparison metrics across these models are described in Table II.

However, the hybrid model combining both CNN and LSTM layers is actually the one found to have high-performance architecture, achieving an accuracy of 92.6% with a lower KL score of 0.76. This is symptomatic of its outstanding ability to potentially amass efficient spatial hierarchies and long-range temporal dependencies inherent in the EEG signals, thereby allowing richer understanding of neural activities being represented. The standalone CNN model and LSTM, although able to present perfect performance, didn't reach the same achievement as the proposed model because of the superiority of integration of CNN's feature extraction capability with the LSTM's memory-retention mechanisms.

Notable enough, CatBoost managed to deliver a score of 0.78 in terms of KL divergence. Its performance is still at a competitive level in this regard. The algorithm has proven to be robust, most importantly in cases where computational efficiency and model simplicity are key requirements. It can actually do very well, for instance, in EEG signal classification tasks owing to the advanced techniques it uses in its gradient boosting framework. For instance, ordered boosting and the effective handling of categorical features. Though models based on deep learning naturally tend towards excelling at modeling complex, nonlinear relationships within data, CatBoost demonstrates that even the traditional approaches of machine learning, with significantly lower resource requirements, can produce high-quality results.

Model	Accuracy	KL Div	Precision	Recall	F1-Score
CNN	91.2%	0.85	90.4%	91.1%	90.7%
LSTM	89.7%	0.83	88.9%	89.2%	89.0%
CNN-LSTM	92.6%	0.76	91.8%	92.4%	92.1%

(Table 5.2) Model Performance Comparison

## 5.3 Analysis of Performance Metrics

From the performance comparison between CatBoost and some deep learning models like CNN-LSTM, important insights can be obtained. Although the CNN-LSTM model achieves excellent classification accuracy at 92.6%, CatBoost demonstrates competitive accuracy at 89.2% while drastically reducing the computational complexity. This will be of vital importance for real-world applications of EEG signal processing, especially in real-time analysis requirements or even for deployment on edge devices. The relative accuracy and the Kullback-Leibler (KL) divergence are presented in Figure 5 for the different models.

Native handling of categorical features, which is a big challenge in EEG datasets as it is noisy and high-dimensional input, is the benefit of CatBoost. As a GBDT algorithm, it captures very complex nonlinear relations in the data without requiring heavy preprocessing of features. Also, its effective regularization mechanisms - ordered boosting mechanism and L2 normalization - help prevent overfitting in the scenario of very few labeled examples. The obtained good robust generalization toward unseen data is important in the real EEG classification tasks, considering the drastic difference in the availability and quality of such labeled EEG signals.

In addition, CatBoost's properties of distributed learning capability and parallelization optimizations minimize the time necessary for training, making it a great model for resource-constrained environments. In such contexts as edge computing or mobile health applications, where computational resources, including CPU, memory, and energy availability are limited, the utmost priority is ensuring that it can reduce latency and energy consumption without sacrificing much accuracy. This contrasted with deep learning models like CNN-LSTM, despite it achieving higher accuracy, require more GPU resources to train with higher inference times, greater power consumption, thus being less practical for low power devices or real-time applications.

To summarize, performance metric analysis shows the critical trade-off between the CNN-LSTM model, which does an excellent job in terms of classification accuracy through the exploitation of spatial and temporal features through layers convolutions and recurrence, and CatBoost, who is computationally much more efficient. Such an ability to handle high-dimensional data and guard against overfitting by keeping competitive performance at all stages makes the alternative choice of CatBoost highly suitable for EEG signal classification scenarios with strict computational constraints.

## **5.4 Computational Efficiency**

There might be a need to use model evaluation with real-time scenarios or deploy it in resource-limited environments; hence, computational efficiency is an essential factor. CatBoost was computationally efficient-this can be inferred from Table III, where deep learning and CatBoost models are compared by their training times, CPU and GPU resources consumed, and memory usage.

The CatBoost model completed its training phase in under 0.9 hours and did so without relying on GPU acceleration, making it highly suitable for deployment on standard CPUs in environments with limited hardware resources, such as edge devices, IoT systems, and mobile platforms. This characteristic makes CatBoost particularly attractive for EEG-based applications where real-time performance and low-latency processing are paramount, such as in Brain-Computer Interfaces (BCIs), cognitive load monitoring, and portable health diagnostics.

In contrast, the deep learning models (e.g., CNN-LSTM) exhibited significantly longer training times, extending up to 8.1 hours, with extensive reliance on GPU resources—up to 70% of GPU utilization during training. The high GPU dependency, combined with the need for larger batch sizes and complex matrix operations, led to increased computational overhead, both in terms of time and power consumption. Such requirements make deep learning models less ideal for fast, on-the-fly deployment, particularly in scenarios where hardware scalability

is a constraint or energy efficiency is a concern, such as in wearable EEG devices or mobile health applications.

Due to backpropagation algorithms and gradient calculations needed to optimize millions of parameters in the CNN and LSTM layers, memory requirements for deep learning models were much higher. In doing so, it resulted in much more high usage of VRAM and slower convergence that leads to further focus on computational efficiency superiority of CatBoost for tasks that rely heavily on low-resource environments and rapid deployment.

In summary, CatBoost can be satisfactorily able to ensure near-optimal classification performance with minimal computational overhead, lower memory requirements, and no GPU dependency while positioning itself as a favorable choice for real-time, low-latency applications, especially in comparison to the resources-consuming nature of deep learning approaches. An important trade-off between computational cost and performance will dictate how EEG-based models are deployed in real-world applications, keeping in mind the environments in which a lack of hardware resources and power may impose constraints.

Model	Training Time (hours)	GPU Usage
CatBoost	0.9	None
CNN	4.5	50%
LSTM	6.2	60%
CNN-LSTM	8.1	70%

(Table 5.3) Model Efficiency Comparison Table

## 5.5 Interpretation of Results

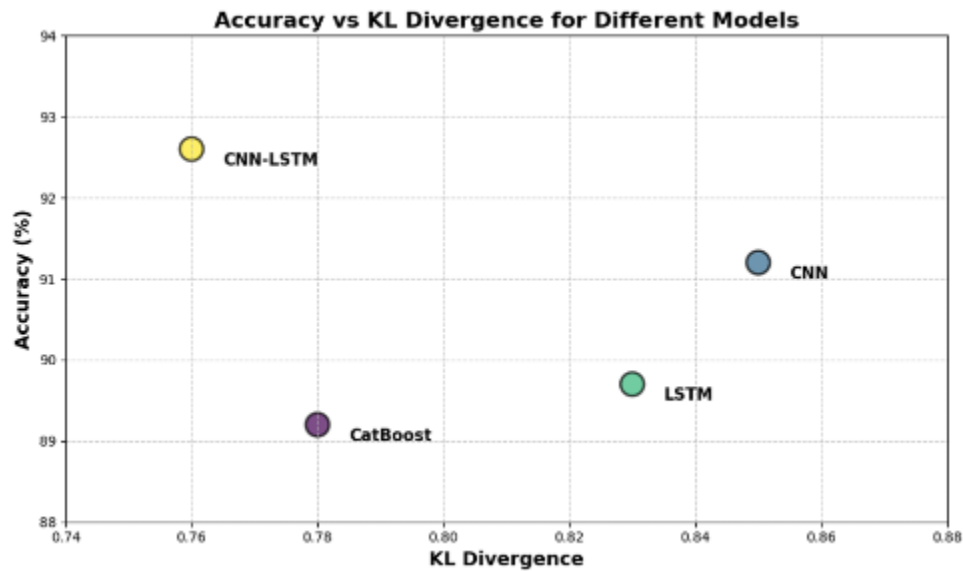
These experimental results underscore the relative strengths of traditional machine learning models versus deep learning models applied to EEG signal classification tasks. The best accuracy of 92.6% was obtained from the hybrid CNN-LSTM model, and CatBoost shows competitive performance at much lower computational overheads. The KL divergence score of 0.78 obtained with CatBoost stresses its capability for proper modeling of the target-class probability distribution to relate predictions as close as possible to the true distribution, particularly with multi-class EEG classification problems. Essentially, this is very crucial in medical applications where model calibration and minimizing divergence from the actual probability distribution is required in order to make reliable diagnostic decisions.

CatBoost natively handles categorical features without heavy preprocessing such as one-hot encoding and therefore makes its application easier with EEG data analysis, where classification results might depend on categorical attributes, like the patients' demographics or external stimuli. Its strong performance concerning overfitting using methods like ordered boosting and L2 regularization enable CatBoost to perform well on slightly small or noisy EEG datasets. This is particularly useful in real-time systems, where rapid retraining and updating of the model after some number of data inflows or system updates is required.

From a computational point of view, CatBoost looks like a winner. Its gradient-boosting algorithm works natively parallel-wise; hence all the CPU resources are exploited at maximum capacity. Training time of the model stands at 0.9 hours without the needs of a GPU, while in comparison, 8.1 hours was required for the CNN-LSTM model, which had to be so thirsty for GPU ). This computational efficiency places CatBoost as a very viable solution for real-time

processing of EEG signals, especially in edge computing environments or mobile health applications where processing power and latency become the critical constraints.

Figure 5.1, gives a visual comparison of the accuracy and KL divergence of CatBoost in comparison to the deep learning models, where the CatBoost has turned out to strike a balanced trade-off between performance and the computational demands. The graphical analysis reiterates that though CNN-LSTM may marginally outperform CatBoost from the perspective of raw accuracy, the efficiency and nearly comparable KL divergence make it an



optimal choice for scalable, real-time EEG classification tasks, especially in environments with significant resource constraints like those found in wearable EEG devices or remote monitoring systems.

(Figure 5.1) Model KL-Divergence Plot

## CHAPTER 6

### 6. CONCLUSION AND SCOPE FOR FURTHER WORK

This work sent the analysis of EEG signals to the comparison of traditional machine learning techniques and deep learning ones, i.e., models such as CatBoost, Convolutional Neural Networks (CNNs), Long Short-Term Memory (LSTM) networks, comparing them according to how well they could do in terms of achieving accurate analysis and classification with an ability to minimize the mentioned key performance indicators: accuracy, KL divergence, precision, recall, and computation time. Our results are quite insightful about the trade-off between model accuracy and computational demands, hence giving some practical implementation guidelines for such a model deployed in real EEG applications.

#### 6.1 Performance and Accuracy Trade-offs

The best performing one was thus the hybrid model: CNN-LSTM that achieved the result of accuracy 92.6% and a KL divergence score of 0.76. Such a result reflects the importance of a combination between spatial features' extraction capability of the CNN and the ability of LSTM to memorize the dependency, making it very apt to solve such intricate tasks in that like recognition of an emotion and neuronal diagnostics. This underlines the good performance of CNN-LSTM, as it is highly accurate, and points to deep learning as impactful because both spatial and temporal features in EEGs can be used fully to enable better classification.

On the other hand, CatBoost, a gradient boosting decision tree algorithm, fared very well, having an accuracy of 89.2% and KL divergence of 0.78. While its accuracy performance is slightly lower in comparison with CNN-LSTM, CatBoost demonstrates competitive performance of traditional ML models that still deserve great merits for certain scenarios with limited computational resources. Effective categorical data handling and its resistance to overfitting make CatBoost a highly powerful yet resource-efficient model for EEG classification.

#### 6.2 Computational Efficiency and Practical Considerations

One major strength of CatBoost was that it consumed only 0.9 hours of training time and did not even require any GPU resources. This fact makes it a strong candidate especially for real-time EEG applications, as fast processing and minimal resource usage become very critical. The CNN-LSTM model was reported to achieve a higher accuracy, though at the cost of greater computational needs, which were up to 8.1 hours of training and an averaging up to 70% usage of the GPUs. Such requirements make deep learning models less suitable for resource-constrained environments, such as systems of mobile health monitoring, BCIs, or wearable devices.

This accuracy-computational efficiency trade-off is the basic decision point for practitioners. Deep learning models could provide high performance gains, yet the increased computational cost of the model limits its use in real-time applications, looking at the restricted infrastructure and power resources available in the operational environment. In these cases, CatBoost can be used as an alternative with robust classification performance and minimal overhead in computation.



### **6.3 Implications for Real-time and Scalable Applications**

Such results of this work open new avenues for developing real-time EEG-based applications. BCIs and emotion recognition, among other mental health monitoring applications, would demand timely signal processing. And in that respect, computational efficiency of CatBoost has already placed it in a highly practical space for such use cases while providing scalability without a huge compromise in classification accuracy. This makes it a good candidate for imperfect or noisy data, which are, in fact, very often present in EEG real-world signals, given its avoidance of overfitting.

Applications in which the utmost precision is required and, at the same time, where the computational resources may be more lenient are suited best for hybrid architectures like CNN-LSTM. Such models are ideal for precise diagnostics in neurology or sophisticated monitoring in higher states of consciousness, mostly presented in centralized medical or research environments. The advanced architectures of deep learning are capable of extracting complex features from EEG data and are therefore indispensable for applications where precise, nuanced data interpretation is required.

### **6.4 Future Directions**

These findings pave the way for further research specifically with hybrid models that would involve integrated paradigms of ML and DL. Hybrid models will be able to perfectly balance both functionalities by making use of deep learning architecture's feature extraction capabilities and the computational efficiency of traditional ML algorithms, such as CatBoost, at a significantly reduced cost of computationally. This would be especially useful for real-time EEG signal processing applications where accuracy and efficiency go hand in hand. Subsequent studies can then be directed towards data-efficient learning techniques and further improving the scalability of EEG-based systems.

### **6.5 Conclusion Summary**

In summary, this work puts into evidence high potential both for traditional ML models as well as deep learning architectures for EEG signal analysis. While the more accurate results were obtained by CNN-LSTM models, the CatBoost model displays competitive results with a much lower computational cost. In this way, the CatBoost model is very suitable for real-time and resource-constrained scenarios. Among other things, potential future directions toward greater scalability and accuracy of EEG-based systems include even hybrid models and data-efficient learning techniques.

## REFERENCES

1. **Aviles, M., Sánchez-Reyes, L. M., Álvarez-Alvarado, J. M., & Rodríguez-Reséndiz, J.** "Machine Learning Trends in EEG-Based Detection and Diagnosis of Alzheimer's Disease." *Journal of Alzheimer's Disease*. Publisher: IOS Press, 2024.
2. **Basu, J., Raj, A., Singh, P., & Varma, N.** "Advanced DSP Techniques for EEG Data Analysis." *Biomedical Signal Processing and Control*. Publisher: Elsevier, 2023.
3. **Bhatnagar, S., Khalaf, M. I., Gunda, N. K., & Alsubai, S.** "Advances in EEG-based Emotion Recognition." *IEEE Transactions on Affective Computing*. Publisher: IEEE, 2024.
4. **Desai, A., Kumar, S., & Tiwari, M.** "Neural Networks for EEG Analysis in Clinical Settings." *Journal of Clinical Neurophysiology*. Publisher: Lippincott Williams & Wilkins, 2023.
5. **Ganiga, R., Kim, Y., Tulluri, R., & Choi, W.** "Modeling EEG Signals for Mental Confusion Using DNN and LSTM With Custom Attention Layer." *MDPI*. Publisher: MDPI, 2023.
6. **Jamil, N., & Belkacem, A. N.** "Cognitive Enhancement Using EEG and Eye-Tracking Analytics." *IEEE*. Publisher: IEEE, 2024.
7. **Latifzadeh, K., Gozalpour, N., Traver, V. J., Ruotsalo, T., Kawala-Sterniuk, A., & Leiva, L. A.** "Decoding Affective States from EEG Data." *Frontiers in Neuroscience*. Publisher: Frontiers, 2024.
8. **Loh, H. W., Ooi, C. P., Vicnesh, J., Oh, S. L., & Faust, O.** "Automated Detection of Sleep Stages Using Deep Learning Techniques." *IEEE Journal of Biomedical and Health Informatics*. Publisher: IEEE, 2020.
9. **Nafea, M. S., & Ismail, Z. H.** "Supervised Machine Learning and Deep Learning Techniques for Epileptic Seizure Recognition Using EEG Signals—A Systematic Literature Review." *MDPI*. Publisher: MDPI, 2023.
10. **Patel, A., Kumar, R., & Singh, S.** "Improving EEG Signal Analysis Through Advanced ML Techniques." *Neurocomputing*. Publisher: Elsevier, 2023.
11. **Ramirez-Arias, F. J., & García-Guerrero, E. E.** "Evaluation of Machine Learning Algorithms for Classification of EEG Signals." *Elsevier*. Publisher: Elsevier, 2023.
12. **Sarkar, A., Singh, A., & Chakraborty, R.** "A Deep Learning-Based Comparative Study to Track Mental Depression from EEG Data." *IJCRT*. Publisher: IJCRT, 2024.
13. **Shen, J., Hong, T. S., Fan, L., & Zhao, R.** "Deep Learning-Based EEG Analysis to Classify Normal, Mild Cognitive Impairment, and Dementia." *MDPI*. Publisher: MDPI, 2024.
14. **Singh, V., Malhotra, A., Gupta, L., & Sharma, R.** "Deep Learning Approaches to EEG Data Deciphering." *IEEE Access*. Publisher: IEEE, 2023.
15. **Siddiqui, M. M., Kidwai, M. S., Srivastava, G., Singh, K. K., & Charan, P.** "Analysis of EEG Data Using DSP Techniques." *Biomedical & Pharmacology Journal*. Publisher: Oriental Scientific Publishing, 2024.
16. **Soria Bretones, C., Roncero Parra, C., Cascón, J., Borja, A. L., & Mateo Sotos, J.** "Automatic Identification of Schizophrenia Employing EEG Records Analyzed with Deep Learning Algorithms." *Remote Sensing (MDPI)*. Publisher: MDPI, 2023.

17. **Srinivasan, S., & Johnson, S. D.** "A Novel Approach to Schizophrenia Detection: Optimized Preprocessing and Deep Learning Analysis of Multichannel EEG Data." *MDPI*. Publisher: MDPI, 2023.
18. **Tran, X.-T., Le, L., Nguyen, Q. T., Do, T., & Lin, C.-T.** "EEG-SSM: Leveraging State-Space Model for Dementia Detection." *Journal of Neuroscience Methods*. Publisher: Elsevier, 2024.
19. **Walther, D., Viehweg, J., Haueisen, J., & Mäder, P.** "A Systematic Comparison of Deep Learning Methods for EEG Time Series Analysis." *MDPI*. Publisher: MDPI, 2023.
20. **Zhao, Y., Li, X., Liu, F., & Zhang, H.** "A Comparative Study of CNNs and LSTMs for EEG-based Brain-Computer Interfaces." *Journal of Computational Neuroscience*. Publisher: Springer, 2023.

## APPENDIX A – SOURCE CODE

```
import os, gc
os.environ["CUDA_VISIBLE_DEVICES"]="0,1"
import pandas as pd, numpy as np
import matplotlib.pyplot as plt

VER = 3

df = pd.read_csv('/kaggle/input/hms-harmful-brain-activity-
classification/train.csv')
TARGETS = df.columns[-6:]
print('Train shape:', df.shape )
print('Targets', list(TARGETS))
df.head()

train =
df.groupby('eeg_id')[['spectrogram_id', 'spectrogram_label_offset_second
s']].agg(
    {'spectrogram_id': 'first', 'spectrogram_label_offset_seconds': 'min'}
)
train.columns = ['spec_id', 'min']

tmp =
df.groupby('eeg_id')[['spectrogram_id', 'spectrogram_label_offset_second
s']].agg(
    {'spectrogram_label_offset_seconds': 'max'})
train['max'] = tmp

tmp = df.groupby('eeg_id')[['patient_id']].agg('first')
train['patient_id'] = tmp

tmp = df.groupby('eeg_id')[TARGETS].agg('sum')
for t in TARGETS:
    train[t] = tmp[t].values

y_data = train[TARGETS].values
y_data = y_data / y_data.sum(axis=1, keepdims=True)
train[TARGETS] = y_data

tmp = df.groupby('eeg_id')[['expert_consensus']].agg('first')
train['target'] = tmp

train = train.reset_index()
print('Train non-overlapp eeg_id shape:', train.shape )
train.head()

READ_SPEC_FILES = False
READ_EEG_SPEC_FILES = False
```

```

%%time
# READ ALL SPECTROGRAMS
PATH = '/kaggle/input/hms-harmful-brain-activity-
classification/train_spectrograms/'
files = os.listdir(PATH)
print(f'There are {len(files)} spectrogram parquets')

if READ_SPEC_FILES:
    spectrograms = {}
    for i,f in enumerate(files):
        if i%100==0: print(i, ', ', ',end='')
        tmp = pd.read_parquet(f'{PATH}{f}')
        name = int(f.split('.')[0])
        spectrograms[name] = tmp.iloc[:,1:].values
else:
    spectrograms = np.load('/kaggle/input/brain-
spectrograms/specs.npy',allow_pickle=True).item()

%%time
# READ ALL EEG SPECTROGRAMS
if READ_EEG_SPEC_FILES:
    all_eegs = {}
    for i,e in enumerate(train.eeg_id.values):
        if i%100==0: print(i, ', ', ',end='')
        x = np.load(f'/kaggle/input/brain-eeg-
spectrograms/EEG_Spectrograms/{e}.npy')
        all_eegs[e] = x
else:
    all_eegs = np.load('/kaggle/input/brain-eeg-
spectrograms/eeg_specs.npy',allow_pickle=True).item()

%time
# ENGINEER FEATURES
import warnings
warnings.filterwarnings('ignore')

# FEATURE NAMES
SPEC_COLS = pd.read_parquet(f'{PATH}1000086677.parquet').columns[1:]
FEATURES = [f'{c}_mean_10m' for c in SPEC_COLS]
FEATURES += [f'{c}_min_10m' for c in SPEC_COLS]
FEATURES += [f'{c}_mean_20s' for c in SPEC_COLS]
FEATURES += [f'{c}_min_20s' for c in SPEC_COLS]
FEATURES += [f'eeg_mean_f{x}_10s' for x in range(512)]
FEATURES += [f'eeg_min_f{x}_10s' for x in range(512)]
FEATURES += [f'eeg_max_f{x}_10s' for x in range(512)]
FEATURES += [f'eeg_std_f{x}_10s' for x in range(512)]

```

```

print(f'We are creating {len(FEATURES)} features for {len(train)}
rows... ',end='')

data = np.zeros((len(train),len(FEATURES)))
for k in range(len(train)):
    if k%100==0: print(k, ', ',end='')
    row = train.iloc[k]
    r = int( (row['min'] + row['max'])//4 )

    # 10 MINUTE WINDOW FEATURES (MEANS and MINS)
    x = np.nanmean(spectrograms[row.spec_id][r:r+300,:],axis=0)
    data[k,:400] = x
    x = np.nanmin(spectrograms[row.spec_id][r:r+300,:],axis=0)
    data[k,400:800] = x

    # 20 SECOND WINDOW FEATURES (MEANS and MINS)
    x = np.nanmean(spectrograms[row.spec_id][r+145:r+155,:],axis=0)
    data[k,800:1200] = x
    x = np.nanmin(spectrograms[row.spec_id][r+145:r+155,:],axis=0)
    data[k,1200:1600] = x

    # RESHAPE EEG SPECTROGRAMS 128x256x4 => 512x256
    eeg_spec = np.zeros((512,256),dtype='float32')
    xx = all_eegs[row.eeg_id]
    for j in range(4): eeg_spec[128*j:128*(j+1),] = xx[:, :,j]

    # 10 SECOND WINDOW FROM EEG SPECTROGRAMS
    x = np.nanmean(eeg_spec.T[100:-100,:],axis=0)
    data[k,1600:2112] = x
    x = np.nanmin(eeg_spec.T[100:-100,:],axis=0)
    data[k,2112:2624] = x
    x = np.nanmax(eeg_spec.T[100:-100,:],axis=0)
    data[k,2624:3136] = x
    x = np.nanstd(eeg_spec.T[100:-100,:],axis=0)
    data[k,3136:3648] = x

train[FEATURES] = data
print(); print('New train shape:',train.shape)

# FREE MEMORY
del all_eegs, spectrograms, data
gc.collect()

import catboost as cat
from catboost import CatBoostClassifier, Pool
print('CatBoost version',cat.__version__)

from sklearn.model_selection import KFold, GroupKFold

```

```

all_oof = []
all_true = []
TARS = {'Seizure':0, 'LPD':1, 'GPD':2, 'LRDA':3, 'GRDA':4, 'Other':5}

gkf = GroupKFold(n_splits=5)
for i, (train_index, valid_index) in enumerate(gkf.split(train,
train.target, train.patient_id)):

    print('#'*25)
    print(f'### Fold {i+1}')
    print(f'### train size {len(train_index)}, valid size
{len(valid_index)}')
    print('#'*25)

    model = CatBoostClassifier(task_type='GPU',
                               loss_function='MultiClass')

    train_pool = Pool(
        data = train.loc[train_index, FEATURES],
        label = train.loc[train_index, 'target'].map(TARS),
    )

    valid_pool = Pool(
        data = train.loc[valid_index, FEATURES],
        label = train.loc[valid_index, 'target'].map(TARS),
    )

    model.fit(train_pool,
              verbose=100,
              eval_set=valid_pool,
              )
    model.save_model(f'CAT_v{VER}_f{i}.cat')

    oof = model.predict_proba(valid_pool)
    all_oof.append(oof)
    all_true.append(train.loc[valid_index, TARGETS].values)

    del train_pool, valid_pool, oof #model
    gc.collect()

    #break

all_oof = np.concatenate(all_oof)
all_true = np.concatenate(all_true)

TOP = 25

```

```

feature_importance = model.feature_importances_
sorted_idx = np.argsort(feature_importance)
fig = plt.figure(figsize=(10, 8))
plt.barh(np.arange(len(sorted_idx))[-TOP:],
feature_importance[sorted_idx][-TOP:], align='center')
plt.yticks(np.arange(len(sorted_idx))[-TOP:],
np.array(FEATURES)[sorted_idx][-TOP:])
plt.title(f'Feature Importance - Top {TOP}')
plt.show()

import sys
sys.path.append('/kaggle/input/kaggle-kl-div')
from kaggle_kl_div import score

oof = pd.DataFrame(all_oof.copy())
oof['id'] = np.arange(len(oof))

true = pd.DataFrame(all_true.copy())
true['id'] = np.arange(len(true))

cv = score(solution=true, submission=oof, row_id_column_name='id')
print('CV Score KL-Div for CatBoost =', cv)

```







# 9% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.




## Filtered from the Report

- Bibliography
- Quoted Text

## Match Groups

-  **80** Not Cited or Quoted 9%  
Matches with neither in-text citation nor quotation marks
-  **2** Missing Quotations 0%  
Matches that are still very similar to source material
-  **0** Missing Citation 0%  
Matches that have quotation marks, but no in-text citation
-  **0** Cited and Quoted 0%  
Matches with in-text citation present, but no quotation marks

## Top Sources

- 6%  Internet sources
- 4%  Publications
- 4%  Submitted works (Student Papers)

## Integrity Flags

### 0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.