Inflammatory Bowel Disease Detection Using Machine Learning Techniques

Veerender Aerranagula
Department of Computer Science and Engineering
National Institute of Technology Warangal, India
va23csr1p04@student.nitw.ac.in

Raju Bhukya
Department of Computer Science and Engineering
National Institute of Technology Warangal, India
raju@nitw.ac.in

Abstract: *Crohn's Disease (CD) is an Inflammatory Bowel Disease (IBD) has seen a sharp rise around 50% worldwide. Therefore, researchers started looking for alternative ways and started applying computation-based deep learning algorithms. We proposed Machine Learning (ML) and Deep Learning (DL) techniques for identification of complex patterns present in the DNA sequences with a primary goal to improve the accuracy. The current study presents key findings of the performance of a variety of ML models decision tree, Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), Convolutional Neural Networks (CNN), CNN+LSTM, CNN+BiLSTM hybrids, resent, Multi-Layer Perceptron (MLP), Gated Recurrent Units (GRU), Transformer-based models, auto encoder with Feedforward Neural Network and other models for the classification of the disease. We analyzed a gene expression dataset obtained from the NCBI. Each model is evaluated based on accuracy, Area Under the Curve-Receiver Operating Characteristic (AUC-ROC), precision, and kappa. The experimental results are compared with state-of-the-art approaches from the existing literature, demonstrating the effectiveness of the proposed model. Among all evaluated methods, the proposed Sequence Read Archive (EMAT) model achieves the highest performance, attaining an accuracy of 88.12%. The comparative analysis confirms that EMAT stands the best-performing model setting a new benchmark.*

Keywords: *Bowel disease, crohn's disease, machine learning*.

Received May 1, 2025; accepted September 30, 2025
https://doi.org/10.34028/iajit/23/2/14

1. Introduction

Crohn's Disease (CD) has affected millions of people worldwide and the numbers are increasing in past decade. It is a chronic inflammatory condition affecting gastrointestinal tract and causing lesions anywhere from mouth to anus. It commonly presents itself in diverse forms of symptoms such as chronic diarrhea, nausea, transmural inflammation, vomiting, stomach pain, deep ulcerations, skin lesions, fatigue, weight loss, rectal bleeding and sometimes fevers. It causes intestinal blockage and in severe cases causes fistula. It was first observed in 1932 by Burrill Crohn and colleagues at Mount Sinai University and hence got its name from the investigator who separated this disease from intestinal tuberculosis. It was initially identified in Britain and northern Europe, later in other countries and geographical locations worldwide such as north America, where around 700,000 people suffer from CD. The disease is difficult to detect and treat despite research going for decades. Due to the prolonged and relapsing nature of the disease and its chronic symptoms, also the unprecedented surge in the number of cases worldwide, has forced researchers to think of methods to detect and diagnose faster.

CD is mostly diagnosed in people aged 18-35 years, and a smaller fraction of people aged 50-60 years. Smoking, taking antibiotics, nonsteroidal anti-inflammatory drugs have been found to increase the risk of developing CD [22, 23].

The actual cause of this disease is not understood but research is still ongoing on this topic, to understand the reason, but scientists have observed some trends in the patients. Factors like genes, lifestyle habits, immune system, ethnicity and environmental conditions play a key role. It was observed that many genes are related to CD. It's not exactly known about their role in the condition, but people having one or more of these genes are more prone to the disease. Also, certain bacteria in the gut microbiome are suspected to be associated with CD, but it is not certain if these bacteria cause CD.

Zhang *et al*. [23] CD for a long period of time was thought as incurable and the researchers believed it more to be a gut disorder due to lifestyle and treatment was focused on alleviating the symptoms, rather than eliminating it. Current methods employed to detect are laboratory examination of blood and stools, MRI scans, endoscopic methods and radiology imaging to evaluate the CD.

Tsai *et al*. [20] therapies such as corticosteroids, immunosuppressant, and biologic drugs that act on specific inflammatory pathways are employed. Surgical procedures such as bowel resection are utilized to treat this disease where the diseased portion of the bowel is excised and cut ends are anastomosed with healthy ends. Though this method provides relief by removing bowel obstruction, it is not the permanent cure as 20%

of the patients go for re-surgery every five years due to the relapsing nature of diseases Since manual methods used for detection like endoscopy and experimental analysis are time consuming, costly, vulnerable to human errors like mistaking for other gastrointestinal problems and also do not work for all patients. We proposed a set of Machine Learning (ML) algorithms that are more efficient and provide faster, accurate results that helps in optimizing the diagnosis.

Artificial Intelligence (AI) is a computer science discipline that has an interdisciplinary scope and is geared towards the formulation of algorithms, models, and systems that can imitate human-like cognitive activities such as learning, reasoning, problem-solving, and decision-making. AI covers many sub fields such as ML, natural language processing, computer vision, and robotics, and has applications in a wide range of fields including bioinformatics. ML which falls under the umbrella of artificial intelligence is becoming increasing prominent in disease prediction and classification in medical field. We explored various ML, Deep Learning (DL) and Natural Language Processing (NLP) models on human genome data to analyze and predict CD. The raw sequence dataset was obtained from NCBI database which were processed by Unal *et al*. [21]. Further sections the literature review of existing works, details about dataset used, data transformation techniques, ML models applied, training and testing results are discussed.

Twenty popular and well-known supervised ML models were reviewed and their performance was evaluated ResNet, decision tree, Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), Bidirectional LSTM (BiLSTM), Convolutional Neural Networks (CNN), CNN+LSTM, CNN+BiLSTM hybrids, Multilayer Perceptron (MLP), Gated Recurrent Units (GRU), transformer-based models, temporal CNN, feed forward neural networks, auto encoders, efficient net and their hybrids.

2. Literature Review

ML and DL are increasingly being used in IBD and related research.

Unal *et al*. [21] predicted CD using raw 16S rRNA sequence data from the human microbiome using k-mer based approach (with k=3, 4, 5) and converting raw sequence reads into De Bruijn graph representations and applied grid search. A total of seven ML models were evaluated with LightGBM (LGBM) achieving the highest accuracy across all k-mer lengths. This was 76.47% accuracy for k=5 and this accuracy was followed closely by Random Forest (RF) algorithm.

In a previous existing study Con *et al*. [4], have compared ML and DL models using their Area Under the Curve (AUC) scores. Studies say that DL models with repeated biomarkers learns complex patterns in a quick manner and more efficient in comparison to traditional ML models. Feed forward Artificial Neural Network (ANN) and RNN were used to predict the efficiency of anti-TNF therapy in treating CD and analyzing biomarker measurements. They achieved AUC scores of 0.710 and 0.754 respectively.

Pei *et al*. [14] identified genetic variants associated with CD and used AUC score as their main metric. It was found that preprocessing techniques like Quality Control (QC) and different amputation methods improved the score. Different ML models like linear penalized LR and non-linear models like Extreme Gradient Boosting (XGBoost), LightGBM, Cat Boost got similar AUC scores around 0.80. They used various biomarkers Peripheral Blood Routine Parameters (PBRPs) and built MLP-ANN model for differentiating Ulcerative Colitis (UC) from CD, (both of which are a form of IBD) as both share similar symptoms.

Romagnoni *et al*. [15] preprocessing techniques (QC, imputation, coding) significantly influence the outcome which results in genetic studies of CD. Feature importance analysis was conducted and it was found that stringent QC doesn't always produce optimal results. Maximum AUC achieved was around 0.80. Most SNPs with genomic nominal significance in previous Genome-Wide Association Study (GWAS) contributed to ML models. Gradient Boosting on decision Trees (GBT) better exploited biased information in missing values. Top performers in the "wisdom of the crowd" exercise used GBT methods, obtaining similar scores.

Olivera and Silverberg [13] in another process focuses on an important genetic, protein, and microbiological biomarkers for management of CD, such as pharmacogenomics markers like Thiopurine methyltransferase (TPMT) and Nudix Hydrolase 15 (NUDT15) for anticipating adverse events linked to thiopurines and Human Leukocyte Antigen (HLA) haplotypes for anti-TNF response. It highlighted promising techniques such as the analysis of gene signatures from Ribonucleic Acid (RNA) sequencing and spatial multi-omics.

Shu *et al*. [18] tested six methods which are Logistic Regression (LR), Decision Tree (DT), K-Nearest Neighbor (KNN), Multinomial Naive Bayes (MNB), MLP, and XGBoost for differentiation between CD and Intestinal Tuberculosis (ITB). Introduced SHAP and LIME interpretability methods. XGBoost was the model that showed the best performance achieving 86% accuracy and was tested in real clinical practice.

Veauthier and Hornecker [22] gave an overview of CD, covering genetic or environmental risk factors, clinical presentations and discussed about different diagnostic endoscopic procedures Ileocolonoscopy, capsule endoscopy, enteroscopy, and biologic treatments, imaging techniques, common symptoms etc.

Ruan *et al*. [16] deep neural networks used for classifying endoscopic IBD images to classify healthy

and pathological, ulcerative colitis and CD, and healthy and. Ulcerative colitis using transfer learning on the used saliency and Guided Back Propagation (GuidedBackProp) algorithms for model interpretability dataset containing endoscopic images. Used saliency and guided back prop algorithms for visualizations of attribution maps.

Dasari and Bhukya [5] recent study used Deep Neural Networks (DNNs) a novel Enhanced Deep Viral Prediction Platform (EDeepVPP) and its hybrid model. Using 10-fold cross validation on human metagenomic datasets, the EdeepVPP- hybrid predictor achieved a high AUC-Receiver Operating Characteristic (ROC) (0.99) and AUC-PR (0.99), outperforming current state-of-the-art techniques functioning as an accurate recommendation system by properly predicting unknown viral sequences like COVID-19, Ebola, and Acquired Immunodeficiency Syndrome (AIDS).

Marlicz *et al*. [9] reviews the role of gut microbiota and various biomarkers (fecal, serological, metabolic) in CD. Decreased abundance of Faecalibacterium prausnitzii in mucosal biopsies is being investigated as a microbial biomarker associated with a higher risk of postoperative endoscopic recurrence. It has also the potential of fecal and serological markers, such as C-Reactive Protein (CRP) and Fecal Calprotectin (FCP), in examining activity of the disease.

Rymarczyk *et al*. [17] assessed histological disease activity in CD and UC by making use of digitized biopsy images. The models aimed to predict established histological scoring systems.

Nayak *et al*. [11] Global Histology Activity Score (GHAS) for CD and geboes score for UC. SA-AbMILP outperformed other models (RNN, Fisher Vector+Random Forest (FV+RF)) in accuracy and kappa values and achieved performance which was comparable to 5 independent pathologists.

Hendrycks and Kevin [7] classified CD using gene expression data obtained from Next-Generation Sequencing (NGS). 16 classifiers (linear and non-linear) were treatment-naive micro biome [12] new-onset CD (Gevers_CCFA_RISK) BioProject accession: PRJEB13679. University of California San Diego. Dimensionality reduction using Principal Component Analysis (PCA) improved the performance of linear models but observed that non-linear models outperformed all linear models.

Gevers *et al*. [6] kugathasan Extra Tree (ET) classifier was the one that achieved highest accuracy score of 0.7905, followed by XGBoost (0.7871) and AdaBoost (0.7743) models.

Sravanthi and Bhukya [19] propose a novel approach that integrates edge information, extracted from the input data, into the UNET architecture a well-established model for image segmentation. Our approach involves modifying the Attention Gate (AG) mechanism within the UNET to emphasize edge features during segmentation. This modification improves the precision of nucleus boundary delineation, particularly in cases with vague or overlapping boundaries, reducing segmentation errors and boosting overall accuracy.

Bhukya and Ashok [1] propose an alternative solution using a combination of deep autoencoder and MLP to overcome this bottleneck and improve the prediction performance. The microarray-based Gene Expression Omnibus (GEO) dataset was employed to train the neural networks. Experimental result shows that this new model, abbreviated as E-GEX, outperforms DL for Gene EXpression (D-GEX) by 16.64% in terms of overall prediction accuracy on GEO dataset. The models were further tested on an RNA-seq based 1000G dataset and E-GEX was found to be 49.23% more accurate than D-GEX

Bhukya [3] D-GEX project by University of California, Irvine approached the problem from a ML perspective, leading to the development of a multi-layer feed forward neural network to infer target gene expressions from clinically measured landmark expressions. Still, the huge number of genes to be inferred from a limited set of known expressions vexed the researchers. Ignoring possible correlation between target genes, they partitioned the target genes randomly and built separate networks to infer their expressions. This paper proposes that the dimensionality of the target set can be virtually reduced using deep auto encoders. Feed forward networks will be used to predict the coded representation of target expressions. In spite of the reconstruction error of the autoencoder, overall prediction error on the microarray-based GEO dataset was reduced by 6.6%, compared to D-GEX. An improvement of 16.64% was obtained on cross platform normalized data obtained by combining the GEO dataset and an RNA-seq based 1000G dataset.

Bhukya and Dasari [2] Accurate splice signal prediction is a cornerstone of gene regulation, biomedical research, and drug discovery. Milletari *et al*. [10] effective splice boundaries detection requires knowledge of the relationship, dependencies, and characteristics of nucleotides in the surrounding region of splice sites.

Bhukya [3] although most of the existing computational techniques classify true and false sites, the classification performance purely depends on the extracted structure-based features.

Con *et al*. [4] the state-of-the-art CNN models achieved excellent performance through automated feature extraction for splice sites, but the degree of model interpretability is relatively weak. To address these challenges, we propose an interpretable CNN framework called InterSSPP for accurate splice site identification.

The literature survey presents about the both ML and DL methods which significantly enhance CD and IBD research across genetic biomarker and histopathological data. Tree-based ML models (e.g., XGBoost,

LightGBM, Gradient Boosting) consistently used for the strong and stable performance, while DL models excel at capturing complex, high-dimensional patterns, particularly in sequencing data. Pre-processing steps such as quality control, imputation, and feature engineering play a critical role in model performance. Collectively, these findings suggest that hybrid, interpretable, and multi-omics-driven ML/DL frameworks in IBD research.

3. Methodology and Implementation

3.1. Overview

In the proposed work a systematic approach consisting of multiple stages. The Figure 1 shows the high-level view of the approach followed. It consists of stages for acquiring the dataset, further processing to make handling easier such as converting from Sequence Read Archive (SRA) to FASTQ, then combining the data into one file per each class.

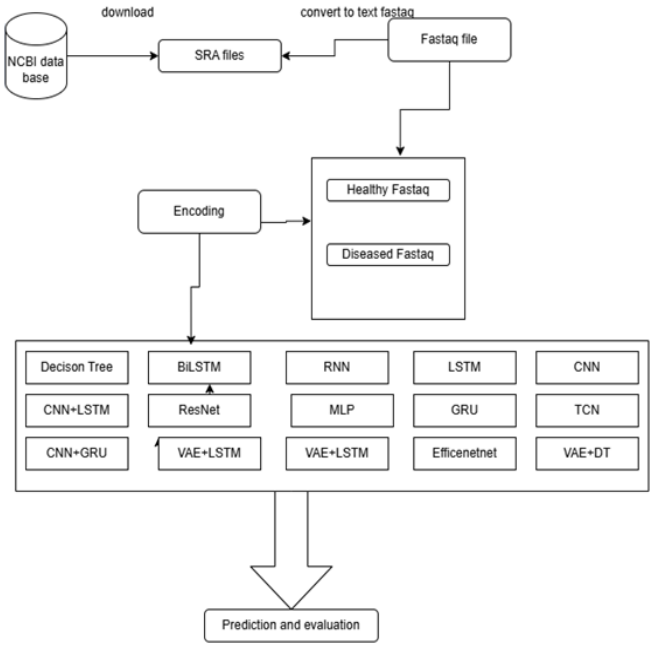


Figure 1. Work flow of the process followed in this study.

Then comes the encoding stage where we have opted for encoding method to convert the string representation of the sequences into numerical representation so that we can feed it to our ML models as input. The encoded data is given to the models while setting aside data for testing appropriately. We split the entire encoded data into training and testing sets and the model is trained only on training data after training, the model is saved and the stage of prediction and evaluation. For prediction, we give the models the testing data that we previously kept aside as input and make it predict the appropriate class for each sequence present in that data. We take the probability values and then process them and use them to evaluate our model. For model evaluation the metrics we have used are accuracy, precision, kappa and AUC. We have evaluated all the models and compared them with existing models. The model that gives best performance from all of our models is then compared with the best performing existing model. Our best performing model is then considered as the proposed model of this study.

3.2. Data Preprocessing

The data is present in FASTQ file format as it contains information of the sequences and also the quality scores. When we get the dataset, it's in SRA format. To make it readable we first convert it into FASTQ format using a simple shell script. It is to note that to download the dataset we have used SRA Toolkit. This SRA Toolkit is provided by NCBI [3]. We made use of prefetch command of the toolkit to download the dataset in SRA format. As stated earlier these files are then converted to FASTQ using a simple shell script. It is to noted that we did not include the samples with diagnosis status as 'control' because they are very minute in number.

The contents of the FASTQ file look like in Figures 1 and 2. This is sanger FASTQ definition of the files. It has four kinds of lines. The first line that contains the symbol '@' contains the ID. Next beside the ID there can be comments. Beside that we see that there is length information of the sequence. In the next line the actual sequence is present. The sequence information is present in FASTA format. The letters present should only be the IUPAC for DNA/RNA and no whitespace is allowed. We can observe that in the third line there is a '+' character. This says that the sequence is finished. Then we can see the quality line or lines depending on the length of the sequence. It contains the quality scores. After parsing this by parsers, many of them give output as 4 lines that can perhaps be long based on the sequence lengths.

```
@ERR1368880.51 1939.100003_50 length=175
TACGGAGGATCCGAGCGTTATCCGGATTTATTGGGTTTAAAGGG
AGCGTAGATGGATGTTTAAGTCAGTTGTGAAAGTTTGCGGCTCA
ACCGTAAAATTGCAGTTGATACTGGATATCTTGAGTGCAGTTGA
GGCAGGCGGAATTCGTGGTGTAGCGGTGAAATGCTTAGATATT
+ERR1368880.51 1939.100003_50 length=175

CCCCCBCCBFFFGGGGGGGGGGHGGGGGHGHHHHHGGGHHFHH
GHGGGGGGGHHHHHHHHHEGHHHHHHHHHHHGGHHHHHHHGGGG
GGHHHHGGHGHHHHHHHHHHHHHHHHHHHHHHHHHHHHHGH
HHHHHHHGGGGGGG-<FFDFCFE<.CFGHGG? E. GFBC00;0<0:::0
@ERR1368880.52 1939.100003_51 length=142
TACGTAGGGAGCAAGCGTTGTCCGGAATTACTGGGTGTAAAGGG
AGCGTAGGCGGGTGCTCAAGTTGAATGTGAAATGCAGAGGCTCA
ACCTCTGAATTGCGTTCAAAAATGAGTATCTTGAGTGAAGTAGA
GGCAGGCGGG
+ERR1368880.52 1939.100003_51 length=142
```

Figure 2. FASTQ file sample after conversion from SRA format.

Figure 3 represents an example after we have added the data into one file. It is to note that this is also in FASTQ format. This format will be easy to parse but not as fast as a plain text file that only contains the sequences. But still in our testing this was not much of a hindrance.

Each sample has a different folder after downloading the dataset. So, we make use of a simple python script to combine the different FASTQ files into one two files. One file for IBD samples and other for non-IBD samples. We know which samples are of IBD and which are not by a csv file containing the metadata. That file contains the diagnosis information along with much more information. This file is downloaded from NCBI. Separating both classes have sampled nearly 33,000 sequences from each class as input to our models meaning a total of 66,000 sequences were used.

```
@ERR1368879.1464 1939.10001_1463 length=132
TACGTAGGAGGATCCGAGCGTTATCCGGATTTATTGGGTTAAAG
GGAGCGTAGGAT
+
BBBCBCCFFCFFGGGGGGGGGGGGGGGGGGGHHHHHHHHHHHH
HHHGGGGHHHHH
@ERR1368879.2085 1939.10001_2084 length=132
TACGTAGGAGGAGCGAGCGTTATCCGGATTTACTGGGTGTAAAG
GGAGCGTAGGCC
+
AAAA>CC>A)>2AAAEGEEGGEEE?AFBIFAS32324BDDONG?1EEEG
H33E8)//7644BFF38
@ERR1368879.2862 1939.10001_2861 length=132
TACGTAGGAGGAGCGAGCGTTGTCCGGAATTTACTGGGTGTAAA
GGGAGCGTAGGC
+
CCCBBFFCFFBBGGGGGGGGGGGGGGGGGGEGAFFHHHHHHHHG
HHHHHHHHHGG
@ERR1368879.2955 1939.10001_2954 length=132
TACGTAGGAGGAGCGAGCGTTGTCCGGAATTTACTGGGTGTAAA
GGGAGCGTAGGC
```

Figure 3. File sample after preprocessing.

Now as the ML models need numbers as input, we further preprocess the data. From both the FASTQ files have to extract the sequences and then encode them into one-hot encoding format. This encoding gives each feature a vector representation. The vectors are binary vectors as in they only contain zeros and ones. The encoding scheme is as follows Figure 4.

```
A -> [1, 0, 0, 0]
C -> [0, 1, 0, 0]
G -> [0, 0, 1, 0]
T -> [0, 0, 0, 1]
```

Figure 4. Encoding scheme followed in this study.

This produce is a unique vector for each of the features. This unique representation allows the model to easily differentiate between each of the nucleotides A, C, T and G in Figure 4. For example, let's take the DNA sequence 'AGCGATCT' in Figure 5. The corresponding representation of the sequence is shown in Figure 6. The similar procedure is followed for every sequence that we have in our dataset. This means that the resulting shape will be number of sequences * 4.

$$
\begin{aligned}
\mathrm{A} &\longrightarrow [1,0,0,0]\\
\mathrm{G} &\longrightarrow [0,0,1,0]\\
\mathrm{C} &\longrightarrow [0,1,0,0]\\
\mathrm{G} &\longrightarrow [0,0,1,0]\\
\mathrm{A} &\longrightarrow [1,0,0,0]\\
\mathrm{T} &\longrightarrow [0,0,0,1]\\
\mathrm{C} &\longrightarrow [0,1,0,0]\\
\mathrm{T} &\longrightarrow [0,0,0,1]
\end{aligned}
$$

Figure 5. Mapping for the example sequence.

$$
\begin{matrix}
1 & 0 & 0 & 0\\
0 & 0 & 1 & 0\\
0 & 1 & 0 & 0\\
0 & 0 & 1 & 0\\
1 & 0 & 0 & 0\\
0 & 0 & 0 & 1\\
0 & 1 & 0 & 0\\
0 & 0 & 0 & 1
\end{matrix}
$$

Figure 6. Encoded sequence.

3.3. Proposed Model

The dataset used in this study was obtained from publicly available biological sequence repositories. The collected data consists of labeled sequence samples required for supervised learning. Care was taken to ensure data consistency and completeness before further processing. Data preprocessing was performed to remove noise and inconsistencies from the raw sequences. This included cleaning invalid or duplicate entries and standardizing sequence formats.

The preprocessed data was then prepared for feature extraction and model input. Biological sequences were transformed into numerical representations suitable for ML models. Sequence encoding techniques were applied to preserve the sequential and structural characteristics of the data, enabling effective learning by DL architectures.

With all biological sequences now converted into numerical representations, the data is prepared for ML analysis. This transformation allows the sequences to be processed by various ML models. The DNA sequences are given to the ML models, and we get the prediction of Inflammatory Bowel Disease (IBD) using the sequences. Given the input sequences are encoded by us using One-Hot encoding, gets prediction through ML models if it is corresponding to the IBD or not. We have considered about twenty different ML models for the part of proposed study.

Each of the twenty models are trained and then tested after the training is done. For this purpose, we have split the dataset into training and testing sets. It is important to make sure that testing of the model is proper by making sure that the testing set and training set will not overlap. To do this we opted to split the dataset into non overlapping sets where eighty percent of the data will be reserved for the training of the models as training set.

In this study we proposed a transformer-based model Enhanced Multi headed Attention Transformer (EMAT), which performed better in comparison to all existing models. Figure 7 shows the architecture view of the transformer model EMAT. As sequences can be considered as text, we have opted to use models that are used in natural language processing. Our transformer-based model leverages multi-head attention mechanisms to effectively capture complex patterns and dependencies within biological sequence data, leading to state-of-the-art performance.

In this study, we implement a transformer-based model named EMAT. The architectural design of

EMAT is illustrated in Figure 7. Since biological sequences can be represented as ordered symbol sequences similar to text, models commonly used in NLP are well suited for this task.

Figure 7 presents visual representation of the architecture of the proposed model. It contains an input layer that is shaped appropriately to facilitate proper taking of input. Then it goes to dense layer which is for projecting the input to higher dimensional space which is required for transformers then followed by Transformer blocks, there are three transformer blocks in the proposed model. Each of the blocks makes use of the concept of multiheaded attention to learn the patterns in the given input. The outer segments of each of the blocks contain activation function as Gaussian Error Linear Unit (GELU) which outperformed other functions such as Exponential Linear Unit (ELU) and ReLU consistently in different practical tasks. After the transformer blocks, there is a Flatten layer that converts its input to a flat vector. This flat vector is then fed to dense layer which also makes use of GELU activation function. Then comes the dropout which randomly drops the output of neurons for the purpose of mitigating overfitting. Finally, it's given to the output layer which gives out the final output.

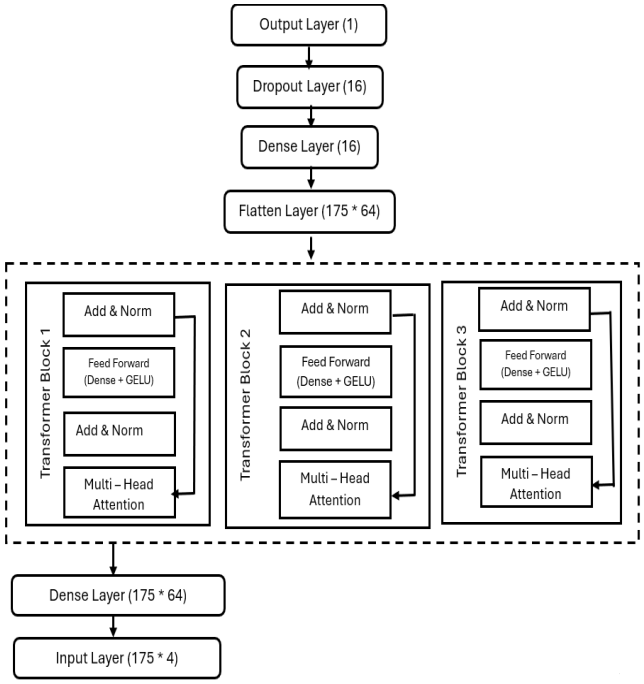


Figure 7. The architecture of transformer model-EMAT.

EMAT leverages advanced transformer-based architectures, particularly multi-head attention mechanisms which addresses key challenges in IBD detection**,** such as complex feature extraction, heterogeneous data integration, and subtle lesion localization. Here are some of the IBD challenges like:

- IBD lesions e.g., ulcers, edema, erosions exhibit diverse morphological patterns in endoscopic images.
- IBD diagnosis relies on multi-modal data (endoscopy, histopathology, lab results), which are inherently heterogeneous.
- Early-stage IBD lesions e.g., aphthous ulcers are small and easily missed.
- Clinicians may require explainable AI for IBD diagnosis.
- Extracting multi-scale lesion features through parallel attention heads.
- Integrating multi-modal data for holistic diagnosis.
- Localizing subtle anomalies via dynamic spatial weighting.

3.4. Training and Testing

As mentioned, we created the training and testing sets. For the training of the models, we make use of only the training set. After the training of each model is done the model is saved and provided it with the testing set of the data. As we have used a total of 66,000 sequences, the size of training set is 53,000 sequences and 13,000 sequences are for testing. Then testing set of the model will make predictions for each of the record. Model will try to classify that particular record into one of the two classes. We keep track of the predictions made by the model as it will be important for the evaluation of the performance of each of the models. Using the predictions made by the model and comparing them with the actual class that is supposed to be for a sequence, we can see performance for each model. For assessing the performance of various models, we have made use of the parameters like accuracy, kappa, precision and ROC AUC score.

If the dataset has an unequal distribution of the two classes (e.g., far more samples from one class than the other), the model may become biased toward the majority class. For instance, if one class (e.g., "disease" samples) significantly outcomes the other ("control" samples), the model may achieve high accuracy by simply predicting the majority class, while performing poorly on the minority class. Metrics like precision and AUC scores can detect such bias. Accuracy can be misleading if classes are imbalanced. Precision focuses on false positives but may not capture false negatives well. AUC is useful but may hide poor performance in the minority class if the majority class dominates. Kappa measures agreement beyond chance, but its interpretation depends on class balance.

The 80/20 train-test split is a well-established standard in ML and DL, particularly suitable for large-sized biological datasets of 63,000 sequences.

3.4.1. Reasons for 80% Training

- Model training needs data: DL models like GRU and RNN require a significant number of examples to learn meaningful patterns in high-dimensional input (e.g., 53,000 sequences).

- Better feature learning: with more training data, the model generalizes better and avoids over fitting, especially important in genomics where subtle patterns define class differences.

3.4.2. Reasons for 20% Testing

- Sufficient size for evaluation: a 20% test set of 13,000 sequences is large enough to Provide statistically meaningful metrics (accuracy, ROC AUC, etc.). Reflect real-world generalization behavior.
- Unbiased performance estimate: since the test set remains untouched during training and Synthetic Minority Over-sampling Technique (SMOTE) balancing, it offers a realistic view of how the model would perform on new, unseen samples.
- Cross-validation for robustness: while the 80/20 split is effective, it evaluates model performance on a single partition of the data, which may be sensitive to how the data was randomly split. To improve robustness and generalization, k-fold cross-validation is recommended. Stratified k-fold cross-validation ensures that each fold maintains the original class distribution (IBD/control), making results more reliable. It reduces the variance of performance estimates by averaging results across multiple splits. Cross-validation is especially useful during hyper parameter tuning and model selection, ensuring the chosen model is not over fitted to a particular data split.

Empirical testing or manual trial-and-error for hyper parameter tuning often lacks reproducibility, scalability, and efficiency. To overcome this, we implemented Bayesian optimization, a probabilistic model-based approach to efficiently explore the hyper parameter space. In our case, Bayesian optimization was applied to tune hyper parameters of the Logistic Regression (LR model). Instead of exhaustively searching all combinations like in grid search or sampling randomly as in random search, Bayesian optimization selects the next set of hyper parameters to evaluate based on the performance of previous trials, using a surrogate function.

We have optimized using the following parameters: C (inverse of regularization strength), max_iter (maximum number of iterations for solver convergence). By doing so, we were able to systematically improve the model's performance, leading to a better-balanced. Confusion matrix and improved metrics such as accuracy and F1-score. This automated approach ensures reproducibility**,** efficiency, and better exploration of the hyper parameter space**,** especially when compared.

3.5. Handling Imbalance with SMOTE

To address this issue, we applied SMOTE only on the training set after splitting the data. SMOTE generates synthetic samples for the minority class (controls) by interpolating between existing control samples in feature space. This approach balances the training data without duplicating data or leaking synthetic information into the test set. The test set was kept in its original imbalanced form to reflect real-world class distributions and ensure unbiased evaluation.

4. Importance of Balanced Data

- Prevents the model from being biased toward the majority class.
- Ensures better detection of minority class (e.g., healthy controls).
- Improves key metrics like recall, F1-score, and ROC-AUC.
- Avoids misleading accuracy scores in imbalanced scenarios.
- Enhances the model's generalization and fairness crucial in medical diagnosis.

Limitations and Potential Biases

Despite the strength of the PRJEB13679 dataset in providing high-quality 16S rRNA sequencing data for IBD classification, certain limitations and potential sources of bias exist.

1) Class Imbalance

The dataset exhibits a notable imbalance, with significantly more IBD samples than healthy controls samples. This can bias the learning algorithm toward the disease class and affect the model's generalizability. While SMOTE was used to balance the training set, synthetic oversampling may not fully capture the complexity of real control samples.

2) Institutional Sampling Bias

All samples were collected from a single source (Massachusetts General Hospital), which may limit population diversity in terms of genetics, lifestyle, environment, and geography. As a result, the model may perform sub optimally on samples from other regions or demographics.

3) Cross-Sectional Data Only

The dataset provides only one sample per individual without any temporal tracking. This limits the model's ability to learn from disease progression, treatment response, or relapse patterns.

4) Region-Specific Sequencing Resolution

The study is based solely on the V4 region of the 16S rRNA gene. While widely used, this region may lack the resolution to distinguish certain bacterial taxa accurately, potentially affecting the quality of the k-mer features used for graph representation. These limitations do not invalidate the results but provide important

context. Acknowledging them highlights the need for future work with more balanced, diverse, and longitudinal datasets to enhance model robustness and general applicability.

Model Evaluation

Evaluation of models is necessary to understand their performance. For this we have selected metrics like accuracy, AUC, precision, and kappa. For each of the models, we have got the values of these parameters. Using those values can understand the performance of each of the models used in our study. Inclusion of confusion matrix as one more way of testing helps us better understands the classification. This is due to the confusion matrix showing us the counts of the predictions into classes. Information about the calculation of the metrics we used in this study.

1) Accuracy

Accuracy measures the number of instances that are correctly classified from the total instances that are to be classified. Following the same, the formula for accuracy can be used as follows:

Accuracy=X+Y/(X+Y+Z+W).

2) Precision

Precision evaluates how many true positive predictions are present in all the positive predictions; hence it measures how many instances that are said to be positive are actually positive. It helps understand what the rate of the positive predictions being correct.

Precision (P)=X/(X+Z)

Where

X: count of true positives.
Y: count of true negatives.
Z: count of false positives.
W: count of false negatives.

3) ROC-AUC Score

It measures how well the model can distinguish between classes present. It is the area under the curve that is made by plotting true positive rate one one axis and false positive rate on the other. This curve is called as ROC curve.

4) Cohen's Kappa

This is a metric for analyzing the agreement between prediction and the truth. A value of 1 shows the agreement is perfect while if its 0 means its random chance. Negative values show disagreement.

5. Experimental Results and Discussion

The sequence dataset was taken from NCBI website (Bioproject PRJEB13679) [12], which was also used in the study of Gevers *et al*. [6] and Unal *et al*. [21]. This dataset contains 1359 samples which in turn contain multiple sequence information and other information like the source of the samples. Although the dataset contains 1359 samples. We did not include the samples which have diagnosis as 'control' this is because very few of these are present in the dataset.

Datasets Used

The information contained ranges from diagnosis, age, sex etc., to more advanced information like where the location of the sample like information habits like smoking. We used the metadata to get information about the dataset. This information allows for knowing what kind of data we are dealing. It also gives about the habits of people. The Figure 8 shows the chart of diagnosis of diseases, where we can observe that it's split under CD, Ulcerative Colitis, and Indeterminate Colitis that all come under different forms of IBD.

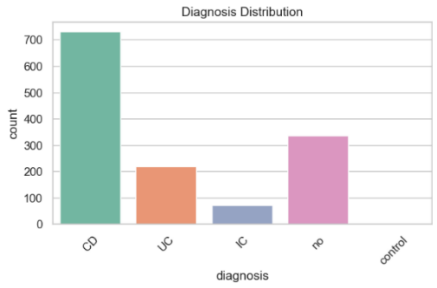


Figure 8. Contains the plot of count vs diagnosis of bowel disease.

It also contains no which means any diagnosis of IBD. It is to note that in the dataset there is also control in the diagnosis status, but it occurred very minutely. This perhaps is one of the limitations of the dataset that there are a smaller number of controls present in the input data set.

The Figure 9 contains visual representation of smoking status in the dataset. The count of the smoking status of the people who fall under the three categories such as never, current, former is plotted for each status. We see that many of the people where the samples were taken have never done any smoking. There were previously studies were made to understand the effect of smoking and IBD. It is observed that in the dataset there are less number of people who smoke or are former smokers.

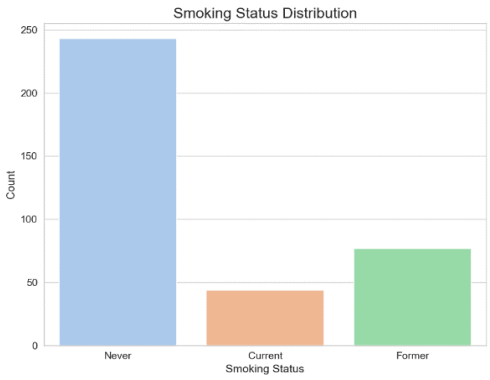


Figure 9. Distribution of smoking status in the dataset.

The Figure 10 is of medical usage correlation. It gives information about how medications are related to each other. The value of 1 means strong positive correlation that tells us that the medications are used at the same time together more often. If the value is 0 it tells us that there is no clear relation between them. If the value is -1, then it shows negative correlation. The more nearer to one of -1, 0, 1 the value is, the more it means that the type of correlation is present.

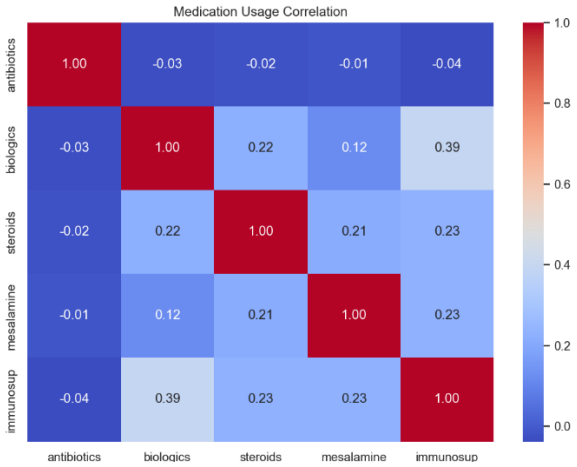


Figure 10. Medical usage correlation.

The above Figure 11 gives the age distribution of people those were diagnosed with IBD (CD, UC or IC) this shows that younger people were diagnosed with IBD in the dataset. We observe bulk of the diagnosis is on people with age less than 20 that may give information about its prevalence in younger people. As the age increases, we observe that the count takes a hard drop. We cannot really say much about this because it may just be that samples are taken from less people who have more age.

Figure 11. Distribution of age of individuals that have IBD in the dataset.

From the above distribution of disease extent vs count for the possible disease extents present in the dataset Figure 12. Disease extent is classified according to the Montreal Classification system, which standardizes the anatomical involvement and clinical behavior of CD. The L* values give us information about the location where the disease is present. The E* values give information about extent of disease. Here j-pouch refers to the surgical procedure hence may be the inability to classify into the Montreal classification.

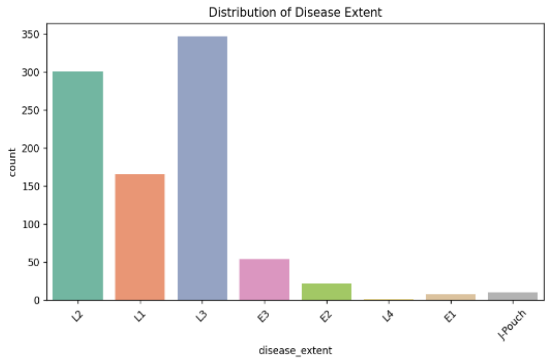


Figure 12. Distribution of disease extent against the count.

Figure13 contains the information of disease duration. It presents the count of the diseases for each of the values in years with disease timeline. We can observe that most of the samples are from people that have had the diseases in the range of 0 to 5 years only. It is seen that after this range, the counts drop dramatically. This can be a form of bias of the dataset which will inevitably affect the ability of the models trained on this dataset. Hence it is important to understand the data on which we are training our models.

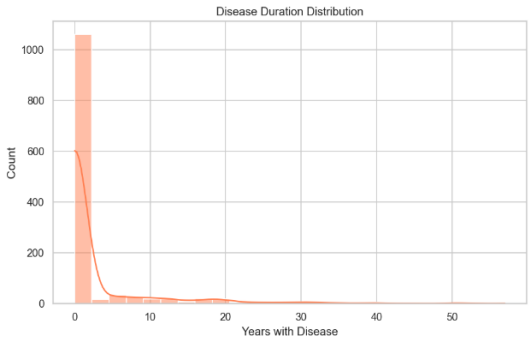


Figure 13. Distribution of disease extent against the count.

In age distribution with respect to diagnosis plot Figure 14 we see the age distribution with respect to different diagnosis such as CD, UC, IC and no. There is a high number of people with no diagnosis compared to the diseases. We can see that from the three CD, UC and IC.

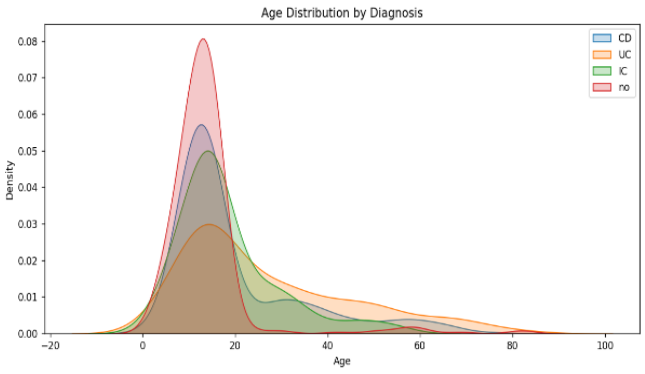


Figure 14. Age distribution with respect to each kind of diagnosis.

The one with highest diagnosis count is CD then followed by UC and IC. It is no surprise that this distribution is similar to that of age distribution of individuals with IBD with the bulk in 0-20 years range.

Similarly, the Figure 15 gives sex information against diagnosis and gives counts. The distributions of diagnosis among both males and females are similar as we see that the order of the counts of the diseases is same with highest being CD followed by UC and then IC with the rest coming under the category of being diagnosed with no IBD. It has been observed that in the dataset, there are more males with CD than any other disease. It is to note that we have excluded control from the figure due to its presence being very minute.

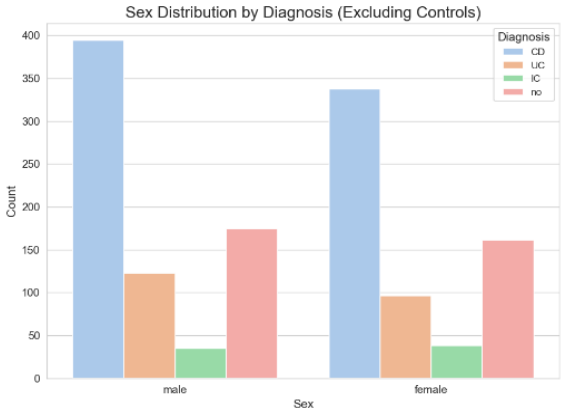


Figure 15. Count of diagnosis based on diagnosis status and sex.

The Figure 16 gives us the visual representations of samples are taken from each location. We see as many (600+) are from the terminal ileum which is the end of small intestine which is towards the large intestine apart from the locations are different parts of the colon which is a part of our large intestine. We also see that there are also stool samples that were taken. There are also samples taken from the sigmoid which is at the end of large interesting and it is present right before reaching the rectum.

The following part presents the results obtained by trying various ML models and comparing each of them with the best performing model of light GBM model and their corresponding bar plots for visualization based on the findings.

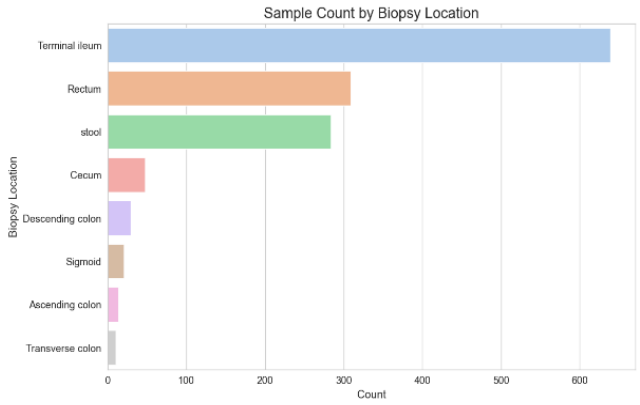


Figure 16. Contains the visual representation of counts for biopsy locations.

1) ResNet

It was introduced in 2015, where researchers have made use of skip connections to address the problem of deep neural networks being difficult to train [3].

Table 1. LGBM vs ResNet comparison based on metrics.

Metric	LGBM	ResNet
Accuracy	0.76	0.85
Precision	0.77	0.90
Kappa	0.53	0.69
AUC	0.82	0.92

From the Table 1 and corresponding plot below Figure 17 we can see that Res Net outperformed the existing model which is LGBM in all the metrics, the greatest difference is seen in kappa values (0.16).

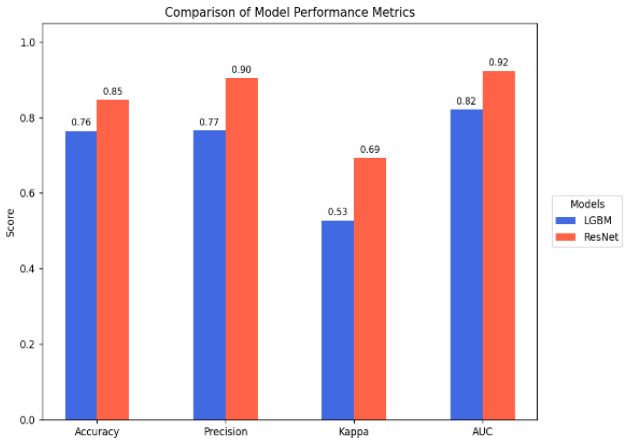


Figure 17. LGBM vs ResNet comparison based on accuracy, precision, kappa, and AUC scores.

Although it was used for image classification, we have tried using a ResNet -like model for sequence classification in this study. In comparison with existing model Figure 17 we observe that our ResNet model outperforms the existing LGBM model. This can be seen in all four of the metrics that we used as part of this study.

2) EMAT-Enhanced Multithreaded Attention Transformer

Madhu *et al*. [8] first presented the transformer which is a transduction model. It is based on attention concept in natural language processing.

From the Table 2 and its corresponding plot below Figure 18 we can see that Transformer outperformed the existing LGBM model in all the metrics with a good margin, the greatest difference is seen in kappa value of (0.23).

Table 2. LGBM vs. Transformer comparison based on metrics.

Metric	LGBM	Transformer
Accuracy	0.76	0.88
Precision	0.77	0.92
Kappa	0.53	0.76
AUC	0.82	0.95

The Figure 18 shows the comparison of this model with LightGBM model based on accuracy, precision,

kappa and AUC scores of both the models. We observe that our Transformer based model outperformed the existing LGBM model on all four of the metrics.

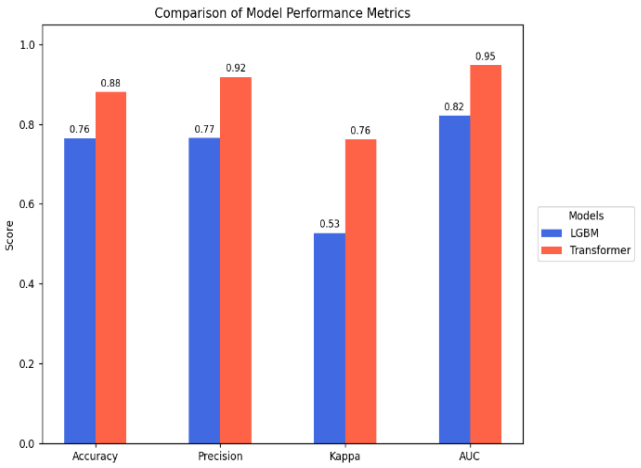


Figure 18. LGBM vs. transformer comparison based on accuracy, precision, kappa, and AUC scores.

3) Recurrent Neural Network (RNN)

RNN is for the purpose of processing data that is sequential. Uses internal loops (recurrence) to maintain a "memory" of past inputs, allowing it to understand context.

Table 3. LGBM vs RNN comparison based on metrics.

Metric	LGBM	RNN
Accuracy	0.76	0.76
Precision	0.77	0.89
Kappa	0.53	0.53
AUC	0.82	0.79

From the Table 3 and corresponding plot below Figure 19 We can see that RNN performed almost equally compared to the existing model in all the metrics but its accuracy is slightly greater than LGBM, while achieving greater precision (0.12) and less AUC score (0.03) to LGBM.

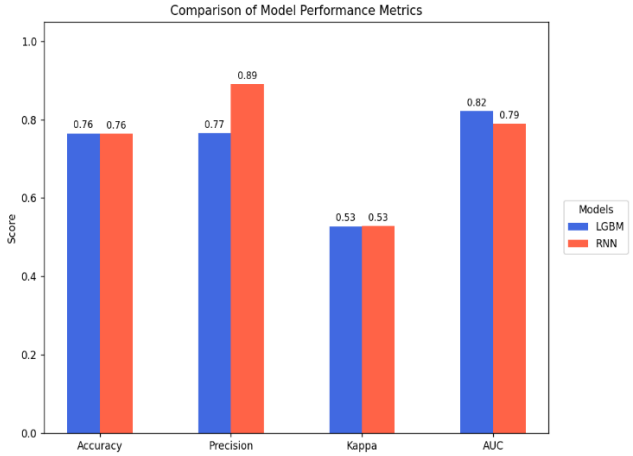


Figure 19. LGBM vs RNN comparison based on accuracy, precision, kappa, and AUC scores.

They have a hidden state to capture information about the previous inputs in the sequence given. Output of the current step is given as an input to the next step. RNNs struggle with long term dependencies present in the sequences. This is addressed by architectures like LSTM and BiLSTM that can be considered more advanced form of RNN. Figure 19 shows the result that we obtained using RNN.

4) Convolution Neural Network (CNN)

The foundation for CNN is by Marlicz *et al.* [9] which used it for handwritten letters recognition in which the MNIST dataset was used.

Table 4. LGBM vs CNN comparison based on metrics.

Metric	LGBM	CNN
Accuracy	0.76	0.79
Precision	0.77	0.83
Kappa	0.53	0.58
AUC	0.82	0.87

Figure 20. LGBM vs CNN comparison based on accuracy, precision, kappa, and AUC scores.

From the Table 4 and corresponding plot below Figure 20 we can see that CNN performed better compared to the existing model LGBM in all the metrics used as a part of this study. A notable score difference is seen in the precision of both the models (0.06) compared to traditional neural networks, CNNs reduce the number of parameters making them easier to train. The result obtained when we used CNN is shown in the plot Figure 21. The proposed CNN Model achieved the scores 0.79, 0.83, 0.58 and 0.87 for accuracy, precision, kappa and AUC scores respectively outperforming LGBM.

5) Long Short-Term Memory (LSTM)

LSTM was introduced for handling issues when training RNN with backpropagation through time. It was developed because of standard RNN facing vanishing and exploding gradient problems making them not much effective when it comes to capturing long-range dependencies.

Table 5. LGBM vs LSTM comparison based on metrics.

Metric	LGBM	LSTM
Accuracy	0.76	0.81
Precision	0.77	1.00
Kappa	0.53	0.62
AUC	0.82	0.72

The comparison between LGBM and LSTM models from Table 5 and corresponding plot below Figure 21 reveals that LSTM excels in accuracy, precision, and kappa, while LGBM outperforms in AUC. LSTM achieves higher accuracy (0.81) and near perfect precision.

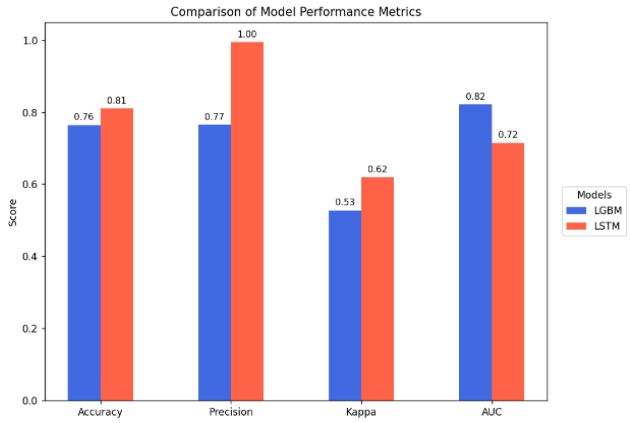


Figure 21. LGBM vs. LSTM comparison based on accuracy, precision, kappa, and AUC score.

In the Figure 21 we have observed that it outperforms the existing LGBM model in metrics like accuracy and precision and kappa.

6) Bidirectional Long Short-Term Memory

BiLSTM was meant as an optimization on LSTM networks to capture the context in both directions of sequences. This means is that for elements in sequence the model has information about both the past and the future context.

Table 6. LGBM vs BiLSTM comparison based on metrics.

Metric	LGBM	BiLSTM
Accuracy	0.76	0.86
Precision	0.77	0.98
Kappa	0.53	0.72
AUC	0.82	0.86

The comparison between LGBM and BiLSTM models from Table 6 and corresponding plot below Figure 22 shows that BiLSTM excels in all metrics. BiLSTM achieved a high accuracy (0.86) and AUC scores (0.86) with near perfect precision (0.98).

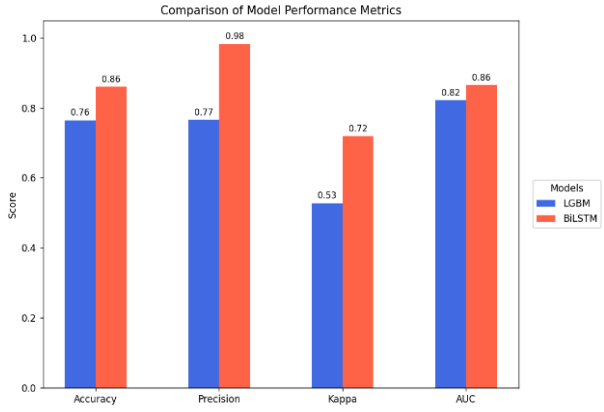


Figure 22 LGBM vs. BiLSTM comparison based on accuracy, precision, kappa, and AUC score.

This makes it good for text classification and other NLP tasks. It also used in DNA sequence classification [2] in studies used a BiLSTM model for differentiating and classifying monkeypox virus DNA sequences and HPV DNA sequences. As per its performance, we can see in the chart Figure 22, it outperforms the existing model in all of the metrics that we have used.

7) Gated Recurrent Unit

GRU are another kind of RNN. They are similar to LSTM but are simpler in structure. In comparison they can be trained in an easier manner due to less calculations.

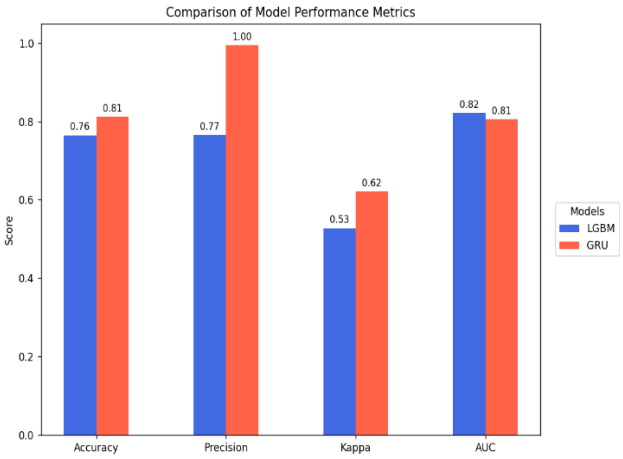


Figure 23. LGBM vs. GRU comparison based on accuracy, precision, kappa, and AUC scores.

Table 7. LGBM vs. GRU comparison based on metrics.

Metric	LGBM	GRU
Accuracy	0.76	0.81
Precision	0.77	1.00
Kappa	0.53	0.62
AUC	0.82	0.81

From the Table 7 and corresponding plot below Figure 23 we can visualize that GRU performed better compared to LGBM model in almost all metrics, except AUC where its score is only 0.81 which is slightly lesser than LGBM's score while excelling at precision (1.00) at the same time.

8) Multilayer Perceptron

MLP is a feedforward neural network that contains multiple layers with each layer consisting of nodes. Each node calculates a weighted sum of inputs and then applies an activation function.

Table 8. LGBM vs MLP comparison based on metrics.

Metric	LGBM	MLP
Accuracy	0.76	0.81
Precision	0.77	0.87
Kappa	0.53	0.61
AUC	0.82	0.87

It contains Input layer after which there are hidden layers and at the end an output layer. From the Table.8 and corresponding plot below Figure 24 We can see that MLP performed slightly better compared to the existing

model LGBM in all the metrics such as accuracy, precision, kappa and AUC.

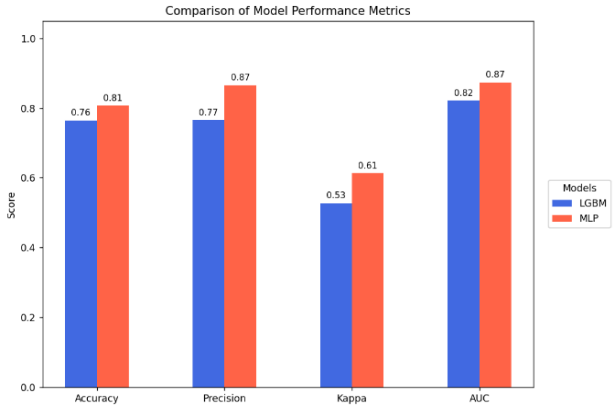


Figure 24. LGBM vs. MLP comparison based on accuracy, precision, kappa, and AUC scores.

9) Variational Autoencoder+Feedforward Neural Network

Variational AutoEncoder (VAE) is similar to the standard Autoencoder but instead of learning fixed encoding of the data, it learns distribution of it over latent space.

From the Table 9 and corresponding plot below Figure 25 we can see that VAE+FNN performed slightly better compared to the existing model LGBM in all the metrics, showing its capability in the task.

Table 9. LGBM vs VAE+FNN comparison based on metrics.

Metric	LGBM	VAE+FNN
Accuracy	0.76	0.82
Precision	0.77	0.93
Kappa	0.53	0.63
AUC	0.82	0.85

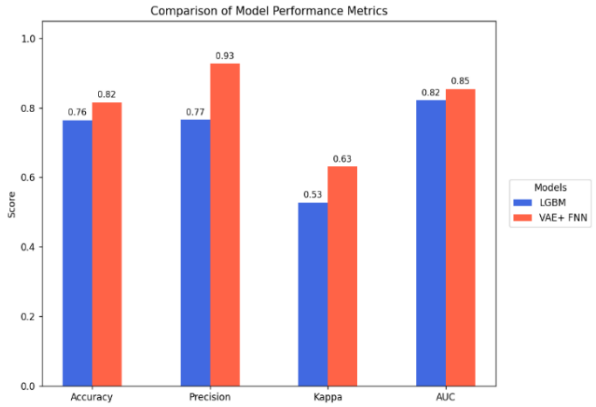


Figure 25. LGBM vs. VAE+FNN comparison based on accuracy, precision, kappa, and AUC scores.

We can make use of these latent representations and give it to a Feedforward Neural Network for classifying. In this case the VAE is used in similar way as feature extraction layer. The result we obtained is plotted Figure 26 showing better performance than existing LGBM model.

10) Decision Tree

DT is a supervised ML algorithm. From the Table 10 and corresponding plot below Figure 26 we can see that DT performed better compared to the existing model LGBM in all the metrics. This is seen in its scores in all of accuracy, precision, kappa and AUC. It is notable that it achieved AUC of 0.95.

Table 10. LGBM vs DT comparison based on metrics.

Metric	LGBM	DT
Accuracy	0.76	0.82
Precision	0.77	0.82
Kappa	0.53	0.64
AUC	0.82	0.95

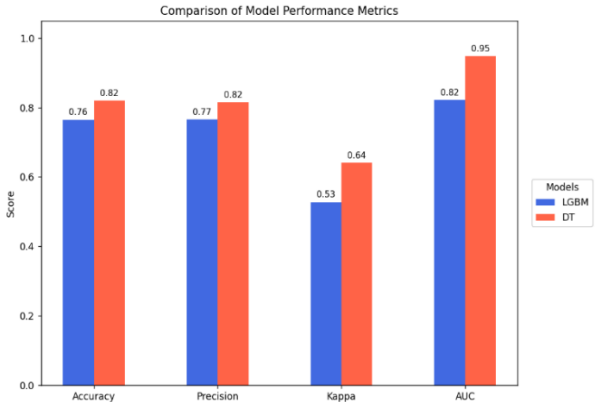


Figure 26. LGBM vs. DT comparison based on accuracy, precision, kappa, and AUC scores.

It splits the data based on decisions made on features making a structure of a tree. The splitting occurs typically on the basis of things such as information gain. This splitting is done repeatedly creating a branch at every decision taken to split by the model. Results are plotted Figure 26 and compared with existing model.

11) CNN+LSTM

This model falls under the category of Hybrid models. Hybrid models are made by adding two models together, testing.

Table 11. LGBM vs CNN+LSTM comparison based on metrics.

Metric	LGBM	CNN+LSTM
Accuracy	0.76	0.83
Precision	0.77	0.93
Kappa	0.53	0.66
AUC	0.82	0.77

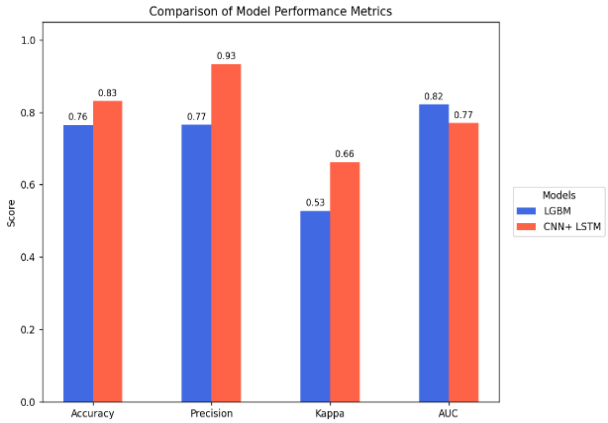


Figure 27. LGBM vs. CNN+LSTM comparison based on accuracy, precision, kappa, and AUC scores.

From the Table 11 and corresponding plot below Figure 27 we can see that CNN+LSTM performed better compared to the existing model LGBM in all the metrics used. Notable that is achieved 0.07 more accuracy compared to existing model.

In this model, the CNN acts like feature extractor that extracts feature from the input and gives it to LSTM as input. The used a hybrid model that used decision tree and RF models in conjunction to form a hybrid model. Our results with this model outperform the existing model.

12) CNN+BiLSTM

Hybrid model model makes use of CNN for the work of extraction of features that are then given to BiLSTM as input. It gave us good results in testing Figure 28, compared with existing model LGBM. It ourperformed it in all of the metrics used in the study.

Table 12. LGBM vs CNN+BiLSTM comparison based on metrics.

Metric	LGBM	CNN+BiLSTM
Accuracy	0.76	0.86
Precision	0.77	0.84
Kappa	0.53	0.71
AUC	0.82	0.93

From the Table 12 and corresponding plot below Figure 28 we can see that CNN+BiLSTM performed moderately better compared to the existing model LGBM in all the metrics, which is observed in its accuracy, precision, kappa and AUC scores obtained. Time-series forecasting or sequence classification is consistent with recent DL research. While LightGBM is highly efficient for tabular data, the CNN-BiLSTM hybrid.

Figure 28. LGBM vs. CNN+BiLSTM comparison based on accuracy, precision, kappa, and AUC scores.

13) Attention BiLSTM

This model combines BiLSTM networks with an attention mechanism and it works by capturing temporal dependencies and stressing key features. Adding attention dynamically assigns weights to highlight crucial input elements.

From the Table 13 and corresponding plot below Figure 29 we can see that attention BiLSTM performed better compared to the existing model LGBM in all the metrics. A notable score is its AUC which was achieved as 0.94.

Table 13. LGBM vs Atten. BiLSTM comparison based on metrics.

Metric	LGBM	Att. BiLSTM
Accuracy	0.76	0.86
Precision	0.77	0.90
Kappa	0.53	0.73
AUC	0.82	0.94

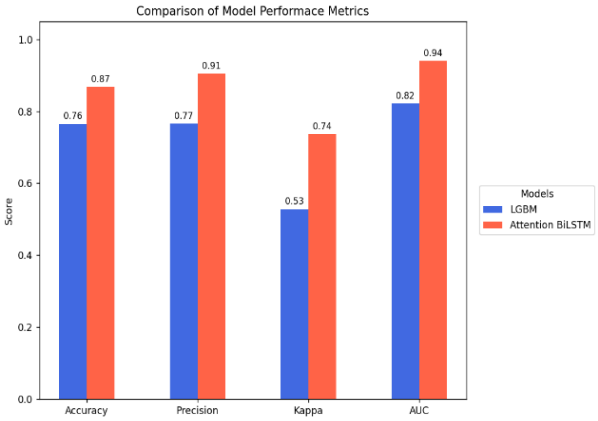


Figure 29. LGBM vs. attension BiLSTM comparison based on accuracy, precision, kappa, and AUC scores.

This combination improves the model's ability to focus on relevant patterns, enhancing accuracy and interpretability in tasks like anomaly detection and NLP. For this task, the performance is plotted Figure 29 and it shows better performance than existing model.

14) CNN+GRU

This model combines CNN for spatial feature extraction and GRUs for sequential data modeling.

From the Table 14 and corresponding plot below Figure 30 we can see that CNN+GRU outperformed the existing model in all of the scores used. It achieved accuracy of 0.85, 0.86 precision, 0.71 kappa and 0.93 AUC score.

Table 14. LGBM vs CNN+GRU comparison based on metrics.

Metric	LGBM	CNN+GRU
Accuracy	0.76	0.86
Precision	0.77	0.87
Kappa	0.53	0.72
AUC	0.82	0.93

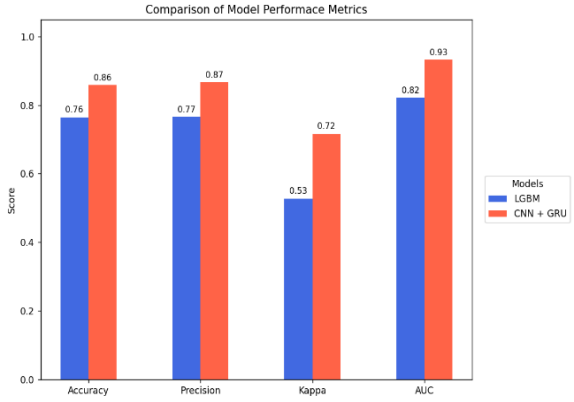


Figure 30. LGBM vs. CNN+GRU comparison based on accuracy, precision, kappa, and AUC scores.

CNNs capture localized patterns, while GRUs address the vanishing gradient problem and efficiently model long-term dependencies with fewer parameters than LSTMs. This hybrid architecture has been used to improve the model performance as seen in its plot Figure 30.

15) VAE+LSTM

VAE combined with LSTM networks leverages VAE's ability to encode data into a latent space while preserving probabilistic structures and LSTM's capability to model temporal dependencies. This architecture is effective for tasks like sequence generation, anomaly detection, and time-series forecasting, as it generates diverse yet coherent sequences by sampling from the latent space. The performance can be seen in the plot Figure 31 showing its ability.

From the Table 15 and corresponding plot below Figure 31 we can see that VAE+LSTM performed better compared to the existing model LGBM in all the metrics. It achieved a notable accuracy score of 0.85, a precision score of 0.90 and 0.93 AUC score.

Table 15. LGBM vs VAE+LSTM comparison based on metrics.

Metric	LGBM	VAE+LSTM
Accuracy	0.76	0.85
Precision	0.77	0.90
Kappa	0.53	0.70
AUC	0.82	0.93

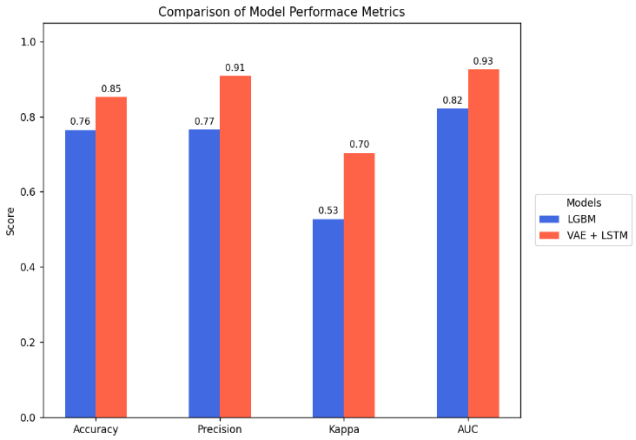


Figure 31. LGBM vs. VAE+LSTM comparison based on accuracy, precision, kappa, and AUC scores.

16) VAE+BiLSTM

The VAE+BiLSTM model enhances sequence modeling by combining VAE's latent space representation with BiLSTMs, which capture dependencies in both forward and backward directions.

From the Table 16 and corresponding plot below Figure 32 we can see that variation autoencoder+BiLSTM performed better compared to the existing model LGBM in all the metrics used. As per accuracy score, it achieved 0.09 more accuracy compared to LGBM and high AUC score of 0.93. This architecture is particularly useful in tasks requiring comprehensive context understanding, such as DNA sequence analysis or sentiment prediction.

Table 16. LGBM vs VAE+BiLSTM comparison based on metrics.

Metric	LGBM	VAE+BiLSTM
Accuracy	0.76	0.85
Precision	0.77	0.88
Kappa	0.53	0.71
AUC	0.82	0.93

Figure 32. LGBM vs. VAE+BiLSTM comparison based on accuracy, precision, kappa, and AUC scores.

17) Stacked BiLSTM

Stacked BiLSTM models have multiple layers of BiLSTM stacked sequentially to capture complex dependencies in sequential data.

From the Table 17 and corresponding plot below Figure 33 we can see that stacked BiLSTM performed better compared to the existing model LGBM in all the metrics. A notable score difference is seen for the metric kappa where Stacked BiLSTM scored 0.21 more compared to LGBM.

Table 17. LGBM vs Stacked BiLSTM comparison based on metrics.

Metric	LGBM	Stacked BiLSTM
Accuracy	0.76	0.87
Precision	0.77	0.87
Kappa	0.53	0.74
AUC	0.82	0.94

Figure 33. LGBM vs. StackedBiLSTM comparison based on accuracy, precision, kappa, and AUC scores.

These models are highly effective due to their ability to process deep contextual information across multiple

layers. Its ability can be seen in the plot Figure 33 as it performed better than the existing model.

18) Efficient Net

Efficient Net is a family of CNN made for image classification, optimizing accuracy and efficiency.

From the Table 18 and corresponding plot below Figure 34 we can see that efficient net model performed better compared to the existing model LGBM in all the metrics. A notable score difference is seen in the precision value of both the models and efficient net has a greater precision score. Its lightweight architecture makes it suitable for resource-constrained environments while maintaining state-of-the-art performance across various computer vision tasks.

Table 18. LGBM vs Efficient Net comparison based on metrics.

Metric	LGBM	EfficientNet
Accuracy	0.76	0.88
Precision	0.77	0.91
Kappa	0.53	0.76
AUC	0.82	0.95

Figure 34. LGBM vs. EfficientNet comparison based on accuracy, precision, kappa, and AUC scores.

We have made use of the architecture for this task by making some adjustments to the inputs so that they can be fed to the model without any issue. It achieved a good performance with respect to all the metrics used Figure 34.

19) VAE+Decision Tree

The VAE+DT model integrates variational autoencoders for dimensionality reduction and latent space representation with decision trees for interpretable classification or regression tasks.

From the Table 19 and corresponding plot below Figure 35 we can see that VAE+DT hybrid model performed slightly better compared to the existing model LGBM. It outperforms LGBM in most metrics, including accuracy (0.80), precision (0.79), and kappa (0.59), while LightGBM slightly leads in AUC (0.82 vs 0.8). By making use of VAE, we get the latent representation of input. This representation is then fed to DT to as an input. This integration balances feature extraction with decision-making. It outperformed existing model LGBM Figure 35 showing good performance.

Table 19. LGBM vs VAE+DT comparison based on metrics.

Metric	LGBM	VAE+DT
Accuracy	0.76	0.80
Precision	0.77	0.79
Kappa	0.53	0.59
AUC	0.82	0.80

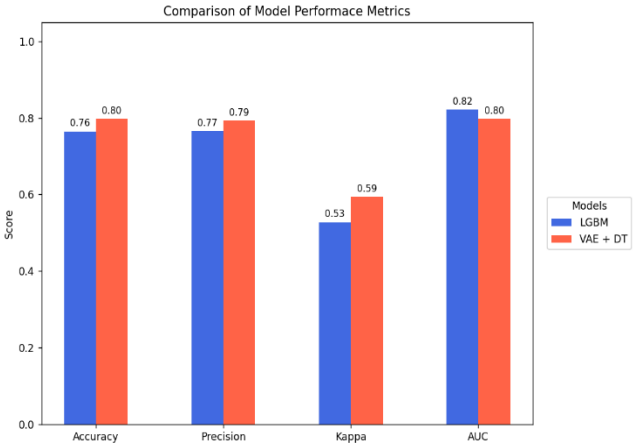


Figure 35. LGBM vs. VAE+DT comparison based on accuracy, precision, kappa, and AUC scores.

20) Temporal Convolutional Networks

Temporal Convolutional Networks (TCNs) are a specialized form of convolutional architecture designed specifically for processing sequential data.

From the Table 20 and corresponding plot below Figure 36 we can see that TCN performed slightly better compared to the existing model LGBM in all the metrics. Notable scores of this model are AUC and precision which are its greatest scores.

Table 20. LGBM vs TCN comparison based on metrics.

Metric	LGBM	TCN
Accuracy	0.76	0.86
Precision	0.77	0.95
Kappa	0.53	0.72
AUC	0.82	0.93

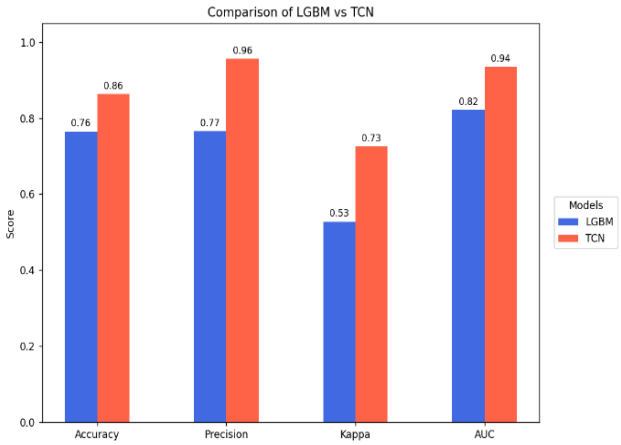


Figure 36. LGBM vs. TCN comparison based on accuracy, precision, kappa, and AUC scores.

From Table 20 and the corresponding plot in Figure 36, it can be observed that the TCN slightly

outperformed the existing LGBM model across all evaluated performance metrics. Although the overall improvement is incremental, the consistency.

TCNs are extensively used in applications like time-series forecasting and speech recognition because they process sequences more quickly than traditional recurrent networks. They utilize dilated convolutions to efficiently capture long-range dependencies while preserving temporal order.

Comparison of all Models

We have the predictions from each of the models and tested them as said in the model evaluation section. We have got the resulting accuracy, AUC, precision and kappa from each of the models. We compared the results with a similar previous study by Zhang *et al*. [23] where the authors have converted the sequences into k-mers of different sizes and represented them in the form of De Bruijn graph. then they have trained and tested seven different ML models.

Table 21. Comparison of results of Unal's study (existing model) with EMAT (proposed study) for k in their study as 3.

Model	Results of	Accuracy	Precision	Kappa	AUC
RF	Existing (Unal's)	0.65	0.65	0.31	0.70
SVM	Existing (Unal's)	0.57	0.57	0.14	0.59
XGB	Existing (Unal's)	0.62	0.67	0.25	0.66
LGBM	Existing (Unal's)	0.67	0.67	0.34	0.70
GNB	Existing (Unal's)	0.61	0.62	0.23	0.64
LR	Existing (Unal's)	0.62	0.67	0.24	0.68
KNN	Existing (Unal's)	0.59	0.59	0.13	0.64
BiLSTM	Proposed (EMAT)	0.85	0.98	0.71	0.86
LSTM	Proposed (EMAT)	0.81	0.99	0.61	0.71
RNN	Proposed (EMAT)	0.76	0.83	0.53	0.77
GRU	Proposed (EMAT)	0.81	0.99	0.62	0.80
VAE+FNN	Proposed (EMAT)	0.81	0.92	0.63	0.85
CNN+LSTM	Proposed (EMAT)	0.83	0.98	0.66	0.76
CNN+BiLSTM	Proposed (EMAT)	0.85	0.84	0.70	0.93
DT	Proposed (EMAT)	0.82	0.81	0.64	0.94
Transformer	Proposed (EMAT)	0.88	0.91	0.76	0.94
MLP	Proposed (EMAT)	0.80	0.86	0.61	0.87
CNN	Proposed (EMAT)	0.79	0.83	0.58	0.87
ResNet	Proposed (EMAT)	0.84	0.90	0.69	0.92
TCN	Proposed (EMAT)	0.86	0.95	0.72	0.93
Attention BiLSTM	Proposed (EMAT)	0.86	0.90	0.73	0.94
CNN+GRU	Proposed (EMAT)	0.85	0.86	0.71	0.93
VAE+LSTM	Proposed (EMAT)	0.85	0.90	0.70	0.92
VAE+BiLSTM	Proposed (EMAT)	0.85	0.88	0.70	0.92
Stacked BiLSTM	Proposed (EMAT)	0.87	0.87	0.74	0.94
EfficientNet	Proposed (EMAT)	0.87	0.91	0.75	0.94
VAE+DT	Proposed (EMAT)	0.79	0.79	0.59	0.79

Table 21 shows us comparison of various models from our study with Zhang *et al.* [23] study for k-mers with value of k as 3. As per the metrics used for evaluating these models, they have opted to use accuracy, precision, F-score, kappa and AUC scores. While comparing our study with the existing models, we did not use F-score metric as the values of beta used to calculate F-score was not stated in the previous study. Hence, we compare the values of accuracy, precision, kappa and AUC. Existing models used seven ML models. Those seven models being Support Vector Machine (SVM), Gaussian Naive Byes (GNB), XGBoost, RF, LGBM, LR and K-Nearest Neighbors (KNN). It is observed that our models outperform the existing models significantly. For k as 3, the highest accuracy obtained by them was 67.28 % using the model LGBM which is surpassed by all of our models. As per the rest of the metrics, our models surpass the highest in each of the metrics being 0.6746, 0.3441, 0.7076 respectively for precision, kappa and AUC.

Table 22. Comparison of results of Unal's study (existing model) with EMAT (proposed study) for k in their study as 4.

Model	Results of	Accuracy	Precision	Kappa	AUC
RF	Existing (Unal's)	0.71	0.71	0.43	0.75
SVM	Existing (Unal's)	0.68	0.68	0.35	0.72
XGB	Existing (Unal's)	0.65	0.63	0.29	0.70
LGBM	Existing (Unal's)	0.74	0.74	0.49	0.80
GNB	Existing (Unal's)	0.57	0.59	0.16	0.59
LR	Existing (Unal's)	0.65	0.65	0.30	0.72
KNN	Existing (Unal's)	0.64	0.64	0.28	0.70
BiLSTM	Proposed (EMAT)	0.85	0.98	0.71	0.86
LSTM	Proposed (EMAT)	0.81	0.99	0.61	0.71
RNN	Proposed (EMAT)	0.76	0.83	0.53	0.77
GRU	Proposed (EMAT)	0.81	0.99	0.62	0.80
VAE+FNN	Proposed (EMAT)	0.81	0.92	0.63	0.85
CNN+LSTM	Proposed (EMAT)	0.83	0.98	0.66	0.76
CNN+BiLSTM	Proposed (EMAT)	0.85	0.84	0.70	0.93
DT	Proposed (EMAT)	0.82	0.81	0.64	0.94
Transformer	Proposed (EMAT)	0.88	0.91	0.76	0.94
MLP	Proposed (EMAT)	0.80	0.86	0.61	0.87
CNN	Proposed (EMAT)	0.79	0.83	0.58	0.87
ResNet	Proposed (EMAT)	0.84	0.90	0.69	0.92
TCN	Proposed (EMAT)	0.86	0.95	0.72	0.93
Attention BiLSTM	Proposed (EMAT)	0.86	0.90	0.73	0.94
CNN+GRU	Proposed (EMAT)	0.85	0.86	0.71	0.93
VAE+LSTM	Proposed (EMAT)	0.85	0.90	0.70	0.92
VAE+BiLSTM	Proposed (EMAT)	0.85	0.88	0.70	0.92
Stacked BiLSTM	Proposed (EMAT)	0.87	0.87	0.74	0.94
EfficientNet	Proposed (EMAT)	0.87	0.91	0.75	0.94
VAE+DT	Proposed (EMAT)	0.79	0.79	0.59	0.79

Table 22 presents comparison of our results with the existing models for the value of k used as 4. In this the highest accuracy obtained by existing model is 74.65 %. It is to note that this is also obtained by LGBM showing its robustness. The next highest accuracy obtained by RF is also notable, being 71.69 %. But it can be observed that all of our models surpass this accuracy score.

As per the precision scores, highest score obtained by existing model is 0.7469 which is also obtained by LGBM. Similarly, it also obtained the highest kappa from the seven models in the study by Zhang *et al.* [23] being 0.4902 which is considerably higher than the model RF that comes second in kappa score of existing models with a value of 0.4325. In a similar fashion LGBM has also obtained the best AUC among the previous models with a value of 0.8057. It is seen that the previous best model for k value as 4 is LGBM and our models surpass its scores in accuracy.

From the models in this study, the ones underperforming as observed in the Table 21 are RNN which was obtained an accuracy of 76.75 % which is still higher than the accuracy obtained by LGBM. Another model that is slightly underperforming is

regular CNN that got accuracy of 79.04 % on the test set.

But we also observed that when CNN is used along with LSTM/BILSTM/GRU where CNN will act as feature extraction and give its output to the other model, the accuracy score is higher 83.17 and 85.93 respectively for CNN+LSTM and CNN+BiLSTM hybrid models. The improvement of score when CNN is used as a feature extractor shows its ability to perform feature extraction. Apart from these, the other hybrid models used in our study such achieved good accuracy. These models made use of VAE to get latent representation of the data which is given input to the models like FNN/LSTM.

BiLSTM/DT to enhance the understanding about data and as a result performs well. Similarly, Table 22 shows us the comparison between proposed models and the existing models in study [19] with values of k used being 5. This contains the best result of that study begin LGBM. The said model got an accuracy of 76.47%. It is observed that all of the models proposed the study exceed the score. The best model in this study is transformer with accuracy of 88.12% precision score of 0.9187 kappa-0.7621 and AUC as 0.9487.

Similar trends have been reported in prior studies, where CNN-based feature extraction has been shown to effectively capture local patterns, while recurrent models excel at modeling long-range dependencies. The observed improvement in performance further reinforces the ability of CNNs to learn discriminative and robust features when integrated with temporal or sequential learning frameworks.

Addition to CNN-based hybrids, other models evaluated in our study demonstrated competitive accuracy by incorporating representation learning techniques. These approaches employed VAEs to learn compact latent representations of the input data, which were subsequently used as inputs to classifiers such as FNNs and LSTMs. Such VAE-driven hybrid frameworks have been increasingly explored in recent literature, as they enable dimensionality reduction and noise filtering while preserving essential data characteristics. Overall, these findings align with existing research that highlights the effectiveness of hybrid DL architectures in improving classification performance by combining complementary model strengths.

The present study surpassed this performance bench mark. Among them, the transformer-based model achieved the best results, attaining an accuracy of 88.12%, a precision score of 0.9187, a Cohen's Kappa value of 0.7621, and an AUC of 0.9487. These results demonstrate the superior capability of transformer architectures in capturing complex feature interactions and long-range dependencies compared to traditional ML and hybrid DL models reported in earlier studies. Other notable models are stacked BiLSTM with accuracy similar to that of efficient net with a value of 87.15%.

Table 23. Comparison of results of Unal's study (existing model) with EMAT (proposed study) for k in their study as 5.

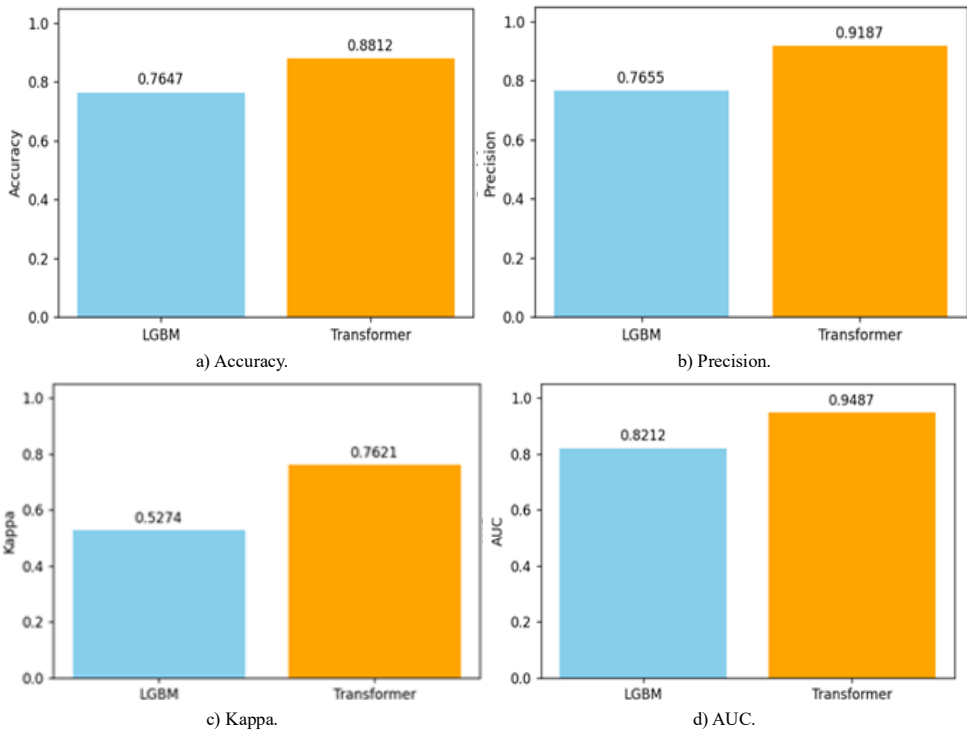
Model	Results	Accuracy	Precision	Kappa	AUC
RF	Existing (Unal's)	0.75	0.75	0.50	0.81
SVM	Existing (Unal's)	0.68	0.68	0.37	0.76
XGB	Existing (Unal's)	0.72	0.72	0.45	0.79
LGBM	Existing (Unal's)	0.76	0.76	0.52	0.82
GNB	Existing (Unal's)	0.61	0.61	0.20	0.60
LR	Existing (Unal's)	0.68	0.68	0.36	0.73
KNN	Existing (Unal's)	0.65	0.65	0.29	0.72
BiLSTM	Proposed (EMAT)	0.85	0.98	0.71	0.86
LSTM	Proposed (EMAT)	0.81	0.99	0.61	0.71
RNN	Proposed (EMAT)	0.76	0.83	0.53	0.77
GRU	Proposed (EMAT)	0.81	0.99	0.62	0.80
VAE+FNN	Proposed (EMAT)	0.81	0.92	0.63	0.85
CNN+LSTM	Proposed (EMAT)	0.83	0.98	0.66	0.76
CNN+Bi LSTM	Proposed (EMAT)	0.85	0.84	0.70	0.93
DT	Proposed (EMAT)	0.82	0.81	0.64	0.94
Transformer	Proposed (EMAT)	0.88	0.91	0.76	0.94
MLP	Proposed (EMAT)	0.80	0.86	0.61	0.87
CNN	Proposed (EMAT)	0.79	0.83	0.58	0.87
Res Net	Proposed (EMAT)	0.84	0.90	0.69	0.92
TCN	Proposed (EMAT)	0.86	0.95	0.72	0.93
Attention BiLSTM	Proposed (EMAT)	0.86	0.90	0.73	0.94
CNN+GRU	Proposed (EMAT)	0.85	0.86	0.71	0.93
VAE+LSTM	Proposed (EMAT)	0.85	0.90	0.70	0.92
VAE+BiLSTM	Proposed (EMAT)	0.85	0.88	0.70	0.92
Stacked BiLSTM	Proposed (EMAT)	0.87	0.87	0.74	0.94
Efficient Net	Proposed (EMAT)	0.87	0.91	0.75	0.94
VAE+DT	Proposed (EMAT)	0.79	0.79	0.59	0.79
Bio bert	Proposed (EMAT)	0.77	0.76	0.51	0.78
Albert	Proposed (EMAT)	0.80	0.80	0.60	0.87
Dense net	Proposed (EMAT)	0.78	0.71	0.43	0.78
Bagging classifier	Proposed (EMAT)	0.76	0.73	0.54	0.75
Ada boost classifier	Proposed (EMAT)	0.82	0.78	0.75	0.82
Voting classifier	Proposed (EMAT)	0.77	0.75	0.64	0.79
Logistic regression	Proposed (EMAT)	0.78	0.79	0.73	0.76
Navie bayes	Proposed (EMAT)	0.74	0.70	0.74	0.59

From the above Table 23 for k=5 we have proposed few more non-NLP models like bio-Bert, albert, densenet, bagging classifiers, ad-boost classifiers, voting classifiers, logistic regression and naive base ML models have been used. From the above Table 23 ad-boost is giving the highest accuracy of 0.82 followed by Albert of 0.80 logistic regressions and dense net of 0.78.

Its accuracy score is followed by efficient net model with 87.83% accuracy. This model is notable because the efficient net architecture is mainly used for images. This works as when we change the shape, the underlying sequence pattern is not disturbed. One of the existing model by 0.1165. All the other metrics are also competitively outperforming the existing model.

Comparison of Various Models

Figure 37 contains the comparison between the previous model and the model that we proposed in this study. accuracy, precision, kappa and AUC. We observe that our proposed model outperforms the existing model in all the metrics used in the study. Our proposed model achieved an accuracy score of 0.8812 which outperformed.



a) Accuracy. b) Precision. c) Kappa. d) AUC.

Figure 37. Comparison of the proposed model with existing model.

The paired t-test is a statistical method used when the same subjects (test samples) are evaluated under two different conditions here, the two models. It has been observed the performance differences between the two models which are logistic regression and AdaBoost not merely due to random chance; a paired t-test was conducted between logistic regression and AdaBoost using their predictions on the same test dataset samples. It measures whether the mean difference in accuracy (or prediction correctness) across all test samples is significantly different from zero.

The test produced between logistic regression and AdaBoost of a t-statistic is -1.5151 and a p-value is0.1311. Since the p-value is greater than 0.05, we conclude that the performance difference is not statistically significant, meaning that while AdaBoost may appear to perform slightly better than logistic regression, which could be due to random variability.

The statistical comparison between logistic regression and AdaBoost yielded a t-statistic of-1.5151 with a corresponding p-value of 0.1311. The p-value represents the probability of observing a performance difference at least as extreme as the one obtained, assuming that there is no true difference between the two models (null hypothesis). Since the p-value (0.1311) is greater than the commonly used significance level of 0.05, we fail to reject the null hypothesis. This indicates that the observed difference in performance between AdaBoost and logistic regression is not statistically significant.

AdaBoost consistently outperforms logistic regression on this dataset.

Although RNNs and GRUs are theoretically well-suited for sequential data like DNA k-mer vectors, their performance in this context may be limited due to several practical factors:

Synthetic Structure of Input Sequences

The input is not a true time series but a fixed-length k-mer vector derived from De Bruijn graphs. These sequences may not contain strong temporal dependencies, reducing the advantage of recurrent architectures.

Noise and Biological Variability

Biological data often contains high intra-class variability and noise. Recurrent models may overfit to noise rather than learn stable patterns without strong regularization.

High Dimensionality with Sparse Information

K-mer vectors are high-dimensional and may include irrelevant or redundant features. Without proper feature selection or dimensionality reduction, RNNs/GRUs may struggle to focus on biologically meaningful patterns. Although RNNs and GRUs are powerful for sequence modeling, their theoretical strengths may not align perfectly with the characteristics of biologically derived k-mer vector sequences, especially under data-limited conditions.

The De Bruijn graph is a specialized graph-based method for DNA sequence assembly, representing sequences as a directed graph of overlapping k-mers. This explicit structure efficiently resolves repeats and sequencing errors a key advantage over neural models like RNNs/Transformers that rely on learned statistical patterns rather than direct overlap analysis. While DL excels at complex pattern recognition (e.g., protein folding), it struggles with genomic-scale sequences,

where the De Bruijn graph's combinatorial efficiency shines. Recent hybrid approaches (e.g., GNN-enhanced De Bruijn graphs) combine their strengths, merging the graph's precision with neural networks' adaptability. This highlights graph-based methods ensure accurate assembly, while DL enables broader sequence analysis, guiding future bioinformatics.

EMAT Integrate into Diagnostic Workflows

- Primary care: EMAT could be used as a rapid, non-invasive initial screening tool for conditions like liver fibrosis, osteoporosis, or cardiovascular disease, reducing reliance on costly imaging.
- Emergency settings: for trauma, EMAT might quickly assess bone integrity or internal bleeding, prioritizing patients for further imaging.
- Longitudinal tracking: EMAT's repeatability could monitor disease progression (e.g., liver stiffness in cirrhosis, plaque stability in atherosclerosis).
- Point-of-care use: portable EMAT devices in clinics could provide real-time data during routine follow-ups for diseases like NAFLD or COPD.
- Biopsy/surgical planning: EMAT could identify tissue abnormalities (e.g., tumors, fibrotic regions) to target biopsies or guide surgical margins, improving precision.

6. Conclusions

We tested and implemented a variety of DL models BiLSTM, LSTM, GRU, RNN, CNN, CNN+BiLSTM, CNN+LSTM, ResNet, transformer, MLP, and VAE+FNN on CD classification using genomic sequence data. All models were compared in terms of accuracy, precision, kappa, and AUC metrics. While all the performances varied, all DL models showed good results compared to traditional ML models discussed with previous paper, with many of them showing more than 80% accuracy and high AUC. Transformer based architecture performed the best among all the models considered with 88.12% accuracy, precision of 0.91, kappa value of 0.76 and AUC score of 0.94. This was followed by CNN+BiLSTM and ResNet which recorded an accuracy of 85% and 84% respectively

The results also indicated that specifically non-linear models performed better than other linear models since.

Genomic and biological sequence data is mostly non-linear and high-dimensional hence linear models (like LR) have trouble modeling these unless they are highly preprocessed. We can conclude that DL models applied on genomic sequences play an important role by speeding the diagnosis, thus aiding many patients by supporting early intervention by alleviating the need of manual testing.

Hybrid models combining feature extraction and sequential learning, as well as transformer-based architectures, showed notable improvements in performance across multiple evaluation metrics. In particular, the proposed transformer-based model effectively captured long-range dependencies and intricate relationships within biological sequences. leading to superior detection accuracy.

References

[1] Bhukya R. and Ashok A., "Gene Expression Prediction Using Deep Neural Networks," *The International Arab Journal of Information Technology*, vol. 17, no. 3, pp. 422-431, 2020. https://doi.org/10.34028/iajit/17/3/16

[2] Bhukya R. and Dasari C., "Inter SSPP: Investigating Patterns Through Interpretable Deep Neural Networks for Accurate Splice Signal Prediction, *Chemometrics and Intelligent Laboratory Systems*, vol. 206, pp. 104144, 2020. https://doi.org/10.1016/j.chemolab.2020.104144

[3] Bhukya R., "Encoding Gene Expression Using Deep Auto Encoders for Expression Inference," *The International Arab Journal of Information Technology*, vol. 18, no. 5, pp. 625-633, 2021. https://doi.org/10.34028/iajit/18/5/1

[4] Con D., Langenberg D., and Vasudevan A., "Deep Learning vs Conventional Learning Algorithms for Clinical Prediction in Crohn's Disease: A Proof-of-Concept Study," *World J Gastroenterol*, vol. 27, no. 38, pp. 6476-6488, 2021. https://doi.org/10.3748/wjg.v27.i38.6476

[5] Dasari C. and Bhukya R., "Explainable Deep Neural Networks for Novel Viral Genome Prediction," *Applied Intelligence*, vol. 52, pp. 3002-3017, 2022. https://doi.org/10.1007/s10489-021-02572-3

[6] Gevers D., Kugathasan S., Denson L., Baeza Y., and et al., "The Treatment-Naive Microbiome in New-Onset Crohn's Disease," *Cell Host Microbe*, vol. 15, pp. 382-392, 2014. https://doi.org/10.1016/j.chom.2014.02.005

[7] Hendrycks D. and Kevin G., "Gaussian Error Linear Units (Gelus)," *arXiv Preprint*, vol. arXiv:1606.08415v5, pp. 1-10, 2016. https://arxiv.org/abs/1606.08415v5

[8] Madhu B., Chari V., Vankdothu R., Silivery A., and Aerranagula V., "Intrusion Detection Models for IoT Networks Via Deep Learning Approaches," *Measurement: Sensors*, vol. 25, pp. 1-14, 2023. https://doi.org/10.1016/j.measen.2022.100641

[9] Marlicz W., Zydecka K., Dabos K., Loniewski I., and Koulaouzidis A., "Emerging Concepts in Non-Invasive Monitoring of Crohn's Disease," *Therapeutic Advances in Gastroenterology*, vol. 11, pp. 1-20, 2018. https://doi.org/10.1177/1756284818769076

[10] Milletari F., Navab N., and Ahmadi S., "V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation," *in*

Proceedings of the 4th International Conference on 3D Vision, Stanford, pp. 565-571, 2016. DOI:10.1109/3DV.2016.79 2024

[11] Nayak D., Routray S., Sahooo S., Sahoo S., and Swarnkar T., "A Comparative Study Using Next Generation Sequencing Data and Machine Learning Approach for Crohn's Disease (CD) Identification," *in Proceedings of the International Conference on Machine Learning, Computer Systems and Security*, Bhubaneswar, pp. 17-21, 2022. https://doi.org/10.1109/MLCSS57186.2022.00012

[12] NCBI BioProject, *The Treatment-Naive Microbiome in New-Onset Crohn's Disease (Gevers_CCFA_RISK)* [BioProject Accession: PRJEB13679], University of California San Diego Microbiome Initiative, https://www.ncbi.nlm.nih.gov/bioproject/PRJEB13679, Last Visited, 2025.

[13] Olivera P. and Silverberg M., "Biomarkers that Predict Crohn's Disease Outcomes," *J Can Assoc Gastroenterol*, vol. 7, no. 1, pp. 59-67, 2023. https://doi.org/10.1093/jcag/gwad024

[14] Pei J., Wang G., Li Y., Li L., and et al., "Utility of Four Machine Learning Approaches for Identifying Ulcerative Colitis and Crohn's Disease," *Heliyon*, vol. 10, no. 1, pp. e23439, 2023. https://doi.org/10.1016/j.heliyon.2023.e23439

[15] Romagnoni A., Jegou S., Steen K., Wainrib G., and Hugot J., "Comparative Performances of Machine Learning Methods for Classifying Crohn Disease Patients Using Genome-Wide Genotyping Data," *Scientific Reports*, vol. 9, no. 1, pp. 1-18, 2019. https://doi.org/10.1038/s41598-019-46649-z

[16] Ruan G., Qi J., Cheng Y., Liu R., and et al., "Development and Validation of a Deep Neural Network for Accurate Identification of Endoscopic Images from Patients with Ulcerative Colitis and Crohn's Disease," *Front Med (Lausanne)*, vol. 9, pp. 854677, 2022. https://doi.org/10.3389/fmed.2022.854677

[17] Rymarczyk D., Schultz W., Borowa A., Friedman J., and et al., "Deep Learning Models Capture Histological Disease Activity in Crohn's Disease and Ulcerative Colitis with High Fidelity," *J Crohns Colitis*, vol. 18, no. 4, pp. 604-614, 2024. https://doi.org/10.1093/ecco-jcc/jjad171

[18] Shu Y., Chen Z., Chi J., Cheng S., and et al., "A Machine Learning Method for Differentiation Crohn's Disease and Intestinal Tuberculosis," *J Multidiscip Healthc*, vol. 17, pp. 3835-3847, 2024. DOI:10.2147/JMDH.S470429

[19] Sravanthi J. and Bhukya R., "Enhanced Nucleus Segmentation with Sobel Edge Detection and Attention Gate with Modified UNET," *The International Arab Journal of Information Technology*, vol. 22, no. 4, pp. 722-729, 2025. https://doi.org/10.34028/iajit/22/4/7

[20] Tsai L., McCurdy J., Ma C., Jairath V., and Singh S., "Epidemiology and Natural History of Perianal Crohn's Disease: A Systematic Review and Meta-Analysis of Population-Based Cohorts," *Inflamm Bowel Dis*, vol. 28, no. 10, pp. 1477-1484, 2022. https://doi.org/10.1093/ibd/izab287

[21] Unal M., Bostanci E., Ozkul C., Acici K., and et al., "Crohn's Disease Prediction Using Sequence Based Machine Learning Analysis of Human Microbiome," *Diagnostics*, vol. 13, no. 17, pp. 2835, 2023. https://doi.org/10.3390/diagnostics13172835

[22] Veauthier B. and Hornecker J., "Crohn's Disease: Diagnosis and Management," *Am Fam Physician*, vol. 98, no. 11, pp. 661-669, 2018. https://www.aafp.org/pubs/afp/issues/2018/1201/p661.html

[23] Zhang Y., Chu X., Wang L., and Yang H., "Global Patterns in the Epidemiology, Cancer Risk, and Surgical Implications of Inflammatory Bowel Disease," *Gastroenterol Rep (Oxf)*, vol. 12, pp. 1-9, 2024. https://doi.org/10.1093/gastro/goae053

Veerender Aerranagula received his B.Tech degree in Information Technology from GEC, Warangal, affiliated to JNTUH University, Hyderabad, India 2011 and M.Tech degree in Software Engineering from BVRIT, Narsapur, affiliated to JNTUH University, Hyderabad, India 2014.Currently, Pursuing PhD in Computer Science and Engineering from NIT, Warangal, India. Her research interests are Bioinformatics, Machine Learning. He has more than ten years of engineering experience and he has guided 8 M.Tech theses and 14 B.Tech projects.

Raju Bhukya has received his B.Tech in Computer Science and Engineering from Nagarjuna University in the year 2003, M.Tech degree in Computer Science and Engineering from Andhra University in the year 2005 and Ph.D. in Computer Science and Engineering from National Institute of Technology (NIT) Warangal in the year 2014. He is currently working as an Associate Professor in the Department of Computer Science and Engineering in National Institute of Technology, Warangal, and Telangana, India. He is currently working in the areas of Bio-Informatics and Data Mining.