Integrating clinical data with deep learning for skin cancer diagnosis

A Project Report submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
IN
COMPUTER SCIENCE AND ENGINEERING

Submitted by

Sharmila Abdul (VU21CSEN0101502)
Sai Charan Gobburu(VU21CSEN0100022)
CH Bhargav Praveen (VU21CSEN0100014)
Koduri Sai Jushatha Varsha(VU21CSEN0100087)

Under the esteemed guidance of

Mr Ranajit Senko
Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
GITAM SCHOOL OF TECHNOLOGY
GITAM (Deemed to be University)
VISAKHAPATNAM
2025

DECLARATION

I hereby declare that the project report entitled "Integrating Clinical Data with Deep Learning For Skin Cancer Diagnosis" is an original work done in the Department of Computer Science and Engineering, GITAM School of Technology, GITAM (Deemed to be University) submitted in partial fulfillment of the requirements for the award of the degree of B.Tech. in Computer Science and Engineering. The work has not been submitted to any other college or University for the award of any degree or diploma.

Date:26-03-2025

Registration No(s)	Name(s)	Signature
VU21CSEN0101502	Sharmila Abdul	
VU21CSEN0100022	Sai Charan Gobburu	
VU21CSEN0100014	Ch Bhargav Praveen	
VU21CSEN0100087	Varsha Koduri	

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GITAM SCHOOL OF TECHNOLOGY

GITAM (Deemed to be University)

CERTIFICATE

This is to certify that the project report entitled “Integrating Clinical Data with Deep Learning For Skin Cancer Diagnosis” is a bonafide record of work carried out by Sharmila Abdul (VU21CSEN0101502), Sai Charan Gobburu (VU21CSEN0100022), Bhargav Praveen Chintapalli (VU21CSEN0100014), Koduri Sai Justhatha Varsha (VU21CSEN0100087) students submitted in partial fulfillment of requirement for the award of degree of Bachelors of Technology in Computer Science and Engineering.

Date : 26-03-2025

Project Guide Head of the Department

ACKNOWLEDGEMENT

We, would like to express our heartfelt gratitude to all those who contributed to the successful completion of our final year project, Integrating clinical data with deep learning for skin cancer diagnosis.

First and foremost, we would like to thank our project supervisor, Mr Ranajit Senko, for their invaluable guidance, support, and encouragement throughout the project. Their expertise and constructive feedback were instrumental in shaping our understanding of the subject and enhancing the quality of our work.

We would like to show our gratitude to Dr. Y. Srinivas, our project coordinator and Associate Professor, Dr. Anuradha Paala of the Department of CSE for their useful advice and frequent reviews that helped us finish the project. We would like to thank Prof. Gondi Lakshmeeswari, Head of the Department, Department of CSE, for her constant motivation that helped us to complete the project work

Lastly, we would like to thank everyone who contributed, directly or indirectly, to the completion of this project. Your support and inspiration were vital in bringing this idea to fruition.

Date: 26-03-25

Sharmila Abdul (VU21CSEN0101502)
Sai Charan Gobburu (VU21CSEN0100022)
Bhargav Praveen Chintapalli(VU21CSEN0100014)
Koduri Sai Justhatha Varsha (VU21CSEN0100087)

TABLE OF CONTENTS

1. ABSTRACT

One of the most common and deadly illnesses in the world is skin cancer, and effective treatment depends on early detection. Conventional diagnostic techniques depend on dermoscopic analysis and clinical evaluation, both of which are subjective and prone to misdiagnosis. The goal of this research is to create a machine learning model-based AI-powered skin cancer detection system that increases diagnostic precision.

The system incorporates a Gradient Boosting Algorithm for processing patient metadata and a Convolutional Neural Network (CNN) for assessing skin lesion photos. A multi-modal approach to classification is made possible by the dataset, which comes from the ISIC 2024 repository and includes over 400,000 tagged skin lesion photos with comprehensive information.

Advanced data preprocessing methods, such as feature engineering and picture augmentation, were used to improve performance.

Users can submit their image and corresponding metadata of that particular image, where they can submit photographs of skin lesions and enter metadata for real-time analysis. After that, the system gives a classification result, which aids both individuals and medical professionals in determining the probability of malignancy. This project provides a quick, scalable, and precise method for early skin cancer detection by fusing deep learning and structured data analysis, which may enhance clinical results and lower diagnostic uncertainty.

2. INTRODUCTION

With millions of new cases reported annually, skin cancer is one of the most prevalent and potentially fatal diseases in the world. Improving treatment results and survival rates requires early identification. Even seasoned dermatologists find it difficult to differentiate between benign and malignant skin tumors. Conventional diagnostic techniques depend on subjective eye inspection and dermoscopic analysis, which might result in incorrect diagnoses. Developments in machine learning (ML) and artificial intelligence (AI) have opened up new avenues for automated and more precise skin cancer detection in order to meet this problem.

In order to create an intelligent skin cancer detection system that examines patient metadata and medical photographs, this research makes use of machine learning. Building a strong model that can help medical practitioners by making quick, accurate, and data-driven predictions is the main goal. The system analyzes structured patient metadata using a Gradient Boosting Algorithm and processes skin lesion photos using a Convolutional Neural Network (CNN). The model improves prediction accuracy by integrating information from both sources, increasing the likelihood of an accurate and timely diagnosis.

The ISIC 2024 archive, a well-known collection of dermatology photos, provided the dataset for this study. In addition to metadata containing patient details, lesion location, and lesion characteristics, it includes more than 400,000 annotated skin lesion photos, each of which is categorized as either benign or malignant. Combining textual and image-based data increases the system's capacity for prediction.

This technology offers a scalable, precise, and effective method of detecting skin cancer by fusing cutting-edge AI approaches with organized patient data. It has the potential to reduce diagnostic errors, enhance early detection efforts, and support clinical decision-making in dermatology.

3. LITERATURE REVIEW

The literature review provides valuable insights into the existing research and advancements in the field of Skin cancer detections i.e hospital management. Here are the key inferences drawn from the survey:

1. **Title :** Skin Cancer Detection Using Ensemble of Machine Learning and Deep Learning Techniques

   Literature Review : To improve the accuracy of skin cancer detection, J. Avanija and D. Harshavardhan Reddy's study from 2023 investigates an ensemble technique that combines many machine learning and deep learning models. According to the study, ensemble approaches perform better than individual models, increasing robustness and generalization while lowering overfitting. Higher computing complexity, unbalanced datasets, and the requirement for meticulous model adjustment are obstacles, too. Even while ensemble models improve accuracy, it's still challenging to include clinical metadata with image-based models. According to the study, deep learning combined with structured clinical data may improve diagnostic accuracy even more.

   Challenges : Challenges include overfitting risk, data imbalance, model complexity, and integration issues.

2. **Title:** Building a Personalized Fitness Recommendation Application based on Sequential Information

   Literature Review: The use of EfficientNet architectures (B0-B7) for skin cancer classification is investigated in the paper by Kanchana K, Kavitha, and Chinthamani (2024), which makes use of transfer learning to improve model performance. According to the study, EfficientNet models perform better than conventional CNNs because of their optimized scaling strategies, which lead to increased accuracy and better feature extraction. The work exhibits improved generalization and less overfitting by integrating pre-trained models that have been trained on sizable datasets. **Challenges** : Challenges include data availability, computational demand, and metadata integration.

4. PROBLEM IDENTIFICATION & OBJECTIVES

Problem Identification:

If not caught early, skin cancer—especially melanoma—is one of the most deadly types of the disease. The conventional diagnostic method mostly depends on the visual examination and experience of dermatologists, which might be arbitrary and subject to human error.

The following factors make early and precise identification difficult to achieve:

- The requirement for qualified experts with specific knowledge.
- Dermatologists are hard to find in isolated places.
- Variability in the appearance of lesions can result in incorrect classification.
- Diagnosis delays raise the chance of death.

An automated, machine-learning-based system that can improve early skin cancer diagnosis with high accuracy and efficiency is required in light of these difficulties.

Objectives:

- Build an AI-based Detection System: Build a machine learning model that can identify the risk of skin cancer by analyzing patient metadata and photos of skin lesions.

- Improve Early Diagnosis: Give medical practitioners an automated tool that can help them identify cancerous skin lesions early on.

- Boost the accuracy of your predictions: For extremely accurate predictions, combine the use of Convolutional Neural Networks (CNN) for image analysis with the Gradient Boosting Algorithm for metadata analysis.

- Assure Robust Data Processing: To increase the model's generalizability, apply picture augmentation techniques. Improve prediction capabilities by feature engineering metadata. Use data balancing strategies, such as Stratified Group K-Fold, to reduce categorization bias.

- Facilitate User-Friendly Communication: Create a straightforward online interface that enables users to input patient information and submit an image. Provide results with precise diagnosis probability in a timely and effective manner.

- Support Clinical Decision Making: Give dermatologists an extra degree of diagnostic assistance. In circumstances that are unclear, support clinician judgments or direct additional research.

5. EXISTING SYSTEM, PROPOSED SYSTEM

Existing System

Dermoscopy, biopsy-based histopathological analysis, and clinical evaluation are the mainstays of the current approach for detecting skin cancer. In order to detect possible cancers, dermatologists visually examine skin lesions using the ABCDE rule (Asymmetry, Border irregularity, Color variation, Diameter, and Evolution). Dermoscopy aids in the analysis of the lesion's subsurface features, as do high-resolution imaging methods like optical coherence tomography (OCT) and reflectance confocal microscopy (RCM). Nevertheless, the interpretation of these pictures is still very subjective and frequently results in different levels of diagnostic accuracy depending on the clinician's level of experience. A biopsy, in which tissue samples are seen under a microscope to determine malignancy, is carried out if a lesion seems worrisome. Despite being the gold standard for diagnosing skin cancer, this procedure is intrusive, expensive, and time-consuming. Furthermore, it can be difficult to detect early-stage melanoma accurately and promptly because of the subtle visual similarities between benign and malignant lesions. These drawbacks emphasize the necessity of automated AI-powered tools that can help dermatologists diagnose patients more quickly and impartially.

Drawbacks of the Existing System:

1. Experience-dependent and subjective diagnosis:
    - The dermatologist's skill has a significant impact on the diagnosis's correctness.
    - Misdiagnosis can result from human mistake, particularly in early-stage melanomas that might mimic benign tumors.
2. A high number of false negatives and false positives:
    - On the other hand, some early-stage melanomas may be incorrectly diagnosed as benign, delaying important treatment; other benign lesions may be confused for malignant ones, resulting in needless biopsies and patient concern.
3. Time-consuming and Invasive Procedures:
    - Biopsies, despite their accuracy, necessitate the removal of tissue samples, which is invasive and can result in infections, discomfort, or scars. The histological analysis process is time-consuming, which delays diagnosis and treatment

4. Limited Access to Dermatologists:

- Patients frequently have to wait a long time for visits and incur high medical costs because many rural and isolated areas lack dermatologists and specialized diagnostic equipment, which makes early detection challenging.

5. Limited Data Utilization:

- Machine learning models can analyze enormous datasets and find patterns beyond human vision, which the current system lacks; traditional approaches do not integrate large-scale medical data for pattern recognition.

Proposed System

Deep learning and machine learning are used in the suggested skin cancer detection method to improve diagnostic speed, accessibility, and accuracy. It combines a Gradient Boosting Algorithm (CatBoost) to process patient metadata, including age, sex, and lesion location, with a Convolutional Neural Network (CNN) to assess skin lesion photos. The approach lowers false positives and negatives and increases classification accuracy by merging image-based and metadata-driven predictions. To improve model robustness, the preprocessing pipeline consists of image augmentation, feature engineering, missing value handling, and one-hot encoding. This AI-driven methodology offers a non-invasive, automated, and scalable option for early skin cancer identification, in contrast to conventional biopsy-dependent procedures. As such, it is accessible to both healthcare professionals and people living in distant places. To guarantee privacy-preserving model improvements, future developments will incorporate explainable AI (Grad-CAM), mobile app integration, multi-class categorization, and federated learning. By cutting down on diagnostic delays and enhancing patient outcomes, this technology has the potential to completely transform the early diagnosis of skin cancer.

Improvements of proposed system over Existing Systems:

1. Improving the Performance of the Model

- Employ Cutting-Edge Deep Learning Frameworks
- Replace EfficientNet-B0 with more potent models such as EfficientNetV2 (fewer parameters, higher accuracy)
- Long-range dependencies are captured by Vision Transformers (ViTs).
- Improved hierarchical feature extraction with Swin Transformer

Classification by Multiple Classes

Go beyond the simple distinction between benign and malignant skin cancer to identify particular forms, like:

- BCC, or basal cell carcinoma
- SCC, or squamous cell carcinoma
- Melanoma
- Keratosis Actinic

2. Enhancing Data Processing :

- Increasing the Size and Variety of Datasets
- To enhance generalization, add global dermatology datasets to training data.
- To lessen racial bias, include pictures of people of various ethnicities and skin tones.

Complex Image Preparation

- Use denoising autoencoders to eliminate extraneous noise from photos of poor quality.
- To increase contrast and make lesion boundaries more visible, use adaptive histogram equalization.

6. SYSTEM ARCHITECTURE/ METHODOLOGY

Block Diagram

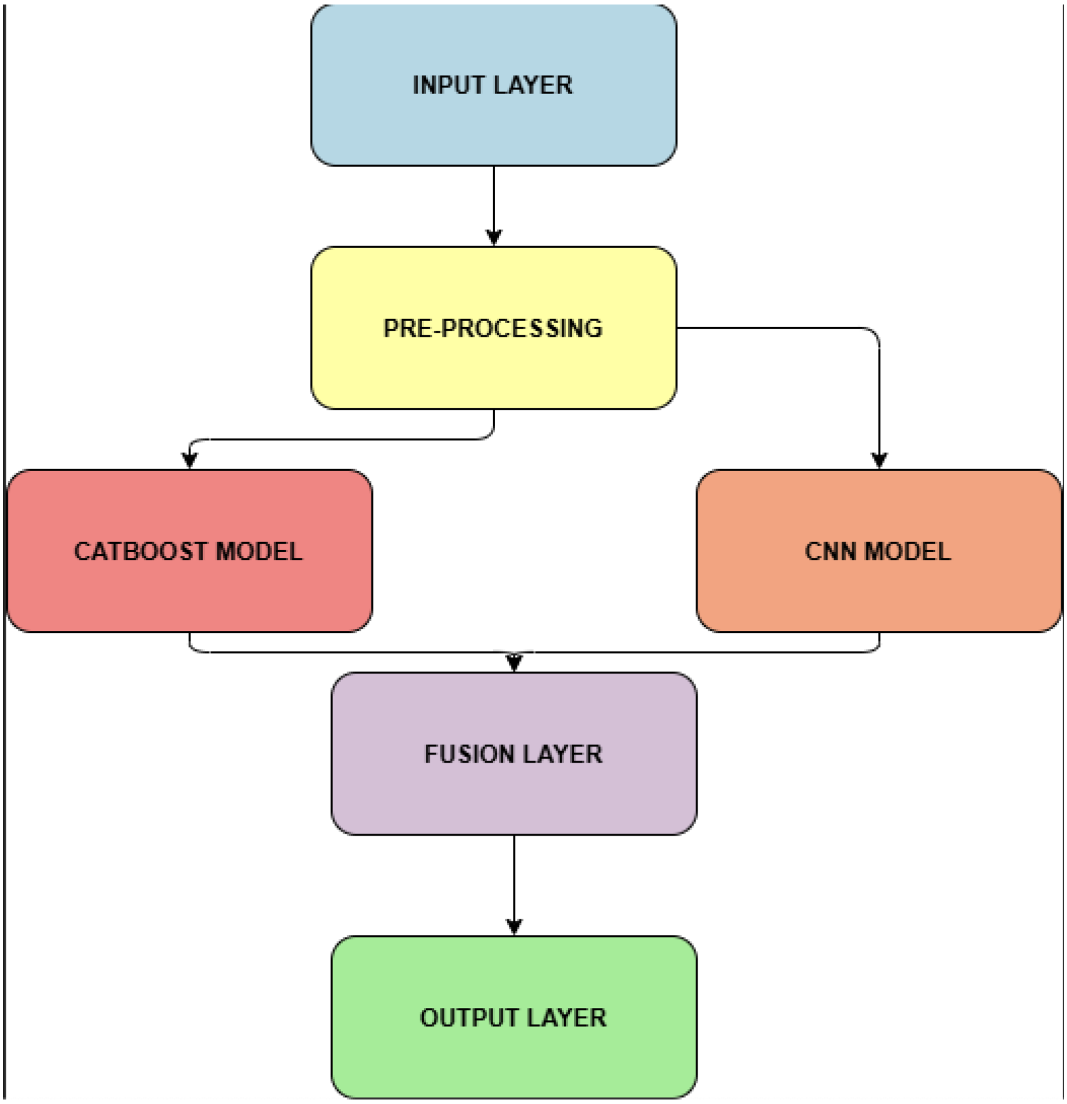


Figure 1: Block Diagram

Use Case Diagram

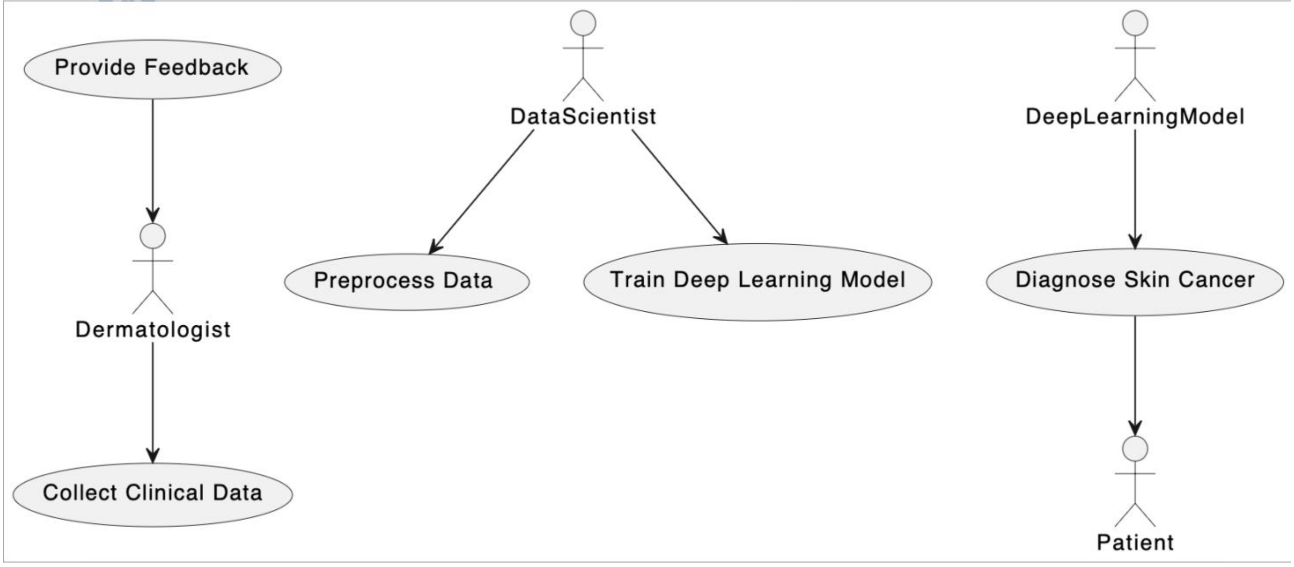


Figure 2: Use Case Diagram

Actors:

- **Dermatologist:** Collects clinical data and provides feedback.
- **Data Scientist:** Preprocesses data and trains the deep learning model.
- **Patient:** Receives diagnosis results.
- **Deep Learning Model:** Performs the diagnosis based on the clinical data.

  Use Cases:

- **Collect Clinical Data:** The dermatologist collects data from the patient.
- **Preprocess Data:** The data scientist preprocesses the collected data for analysis.
- **Train Deep Learning Model:** The data scientist trains the model using the preprocessed data.
- **Diagnose Skin Cancer:** The deep learning model analyzes the data to provide a diagnosis.
- **Provide Feedback:** The dermatologist receives feedback from the model and interacts with the patient.

Activity Diagram

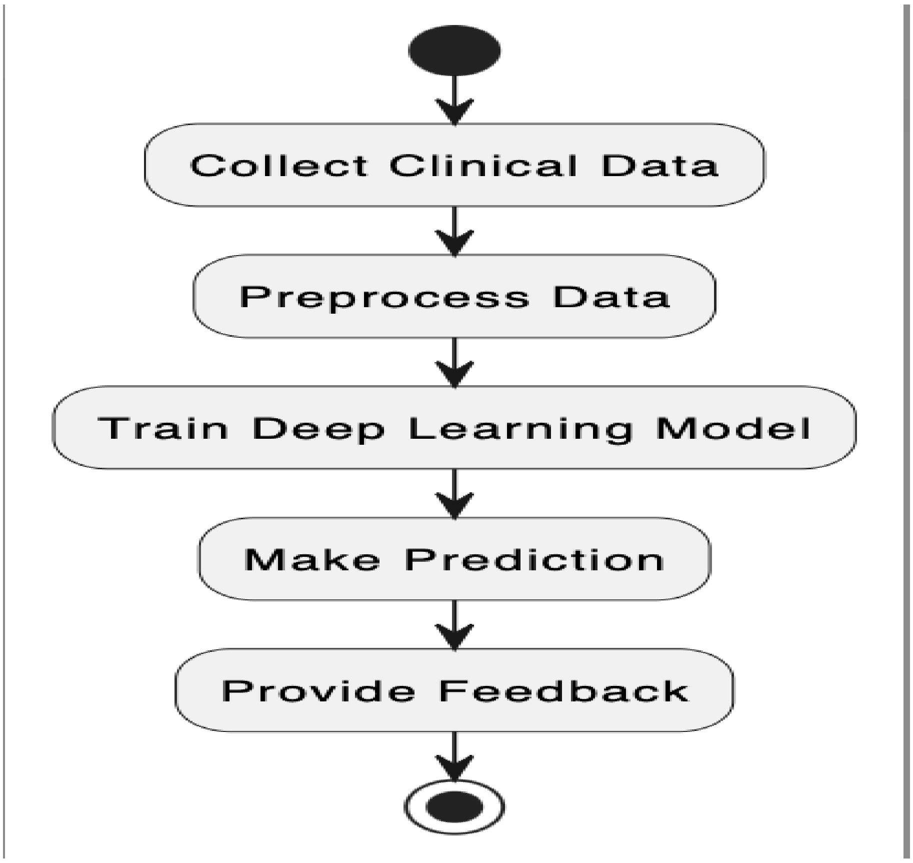


Figure 3: Activity Diagram

1. **Collect Clinical Data:** Gather information from the patient, including medical history and images.
2. **Preprocess Data:** Clean and prepare the data for analysis.
3. **Train Deep Learning Model:** Use the preprocessed data to train the model.
4. **Make Prediction:** Utilize the trained model to diagnose the skin condition.
5. **Provide Feedback:** The dermatologist receives the diagnosis and provides feedback to the patient.

Sequence Diagram

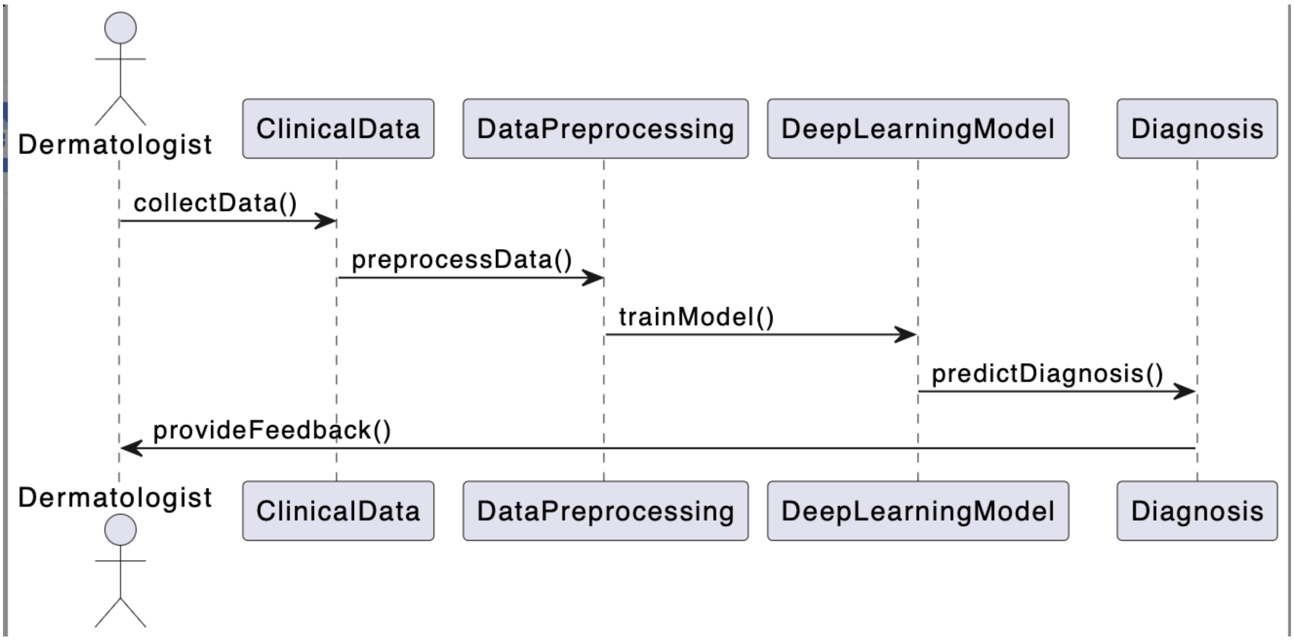


Figure 3: Sequence Diagram

1. **Dermatologist** collects clinical data from the patient.
2. The **ClinicalData** component processes the collected data.
3. **DataPreprocessing** prepares the data for training.
4. The **DeepLearningModel** is trained using the preprocessed data.
5. After training, the model predicts the diagnosis.
6. The **Diagnosis** component provides feedback to the dermatologist.

Class Diagram

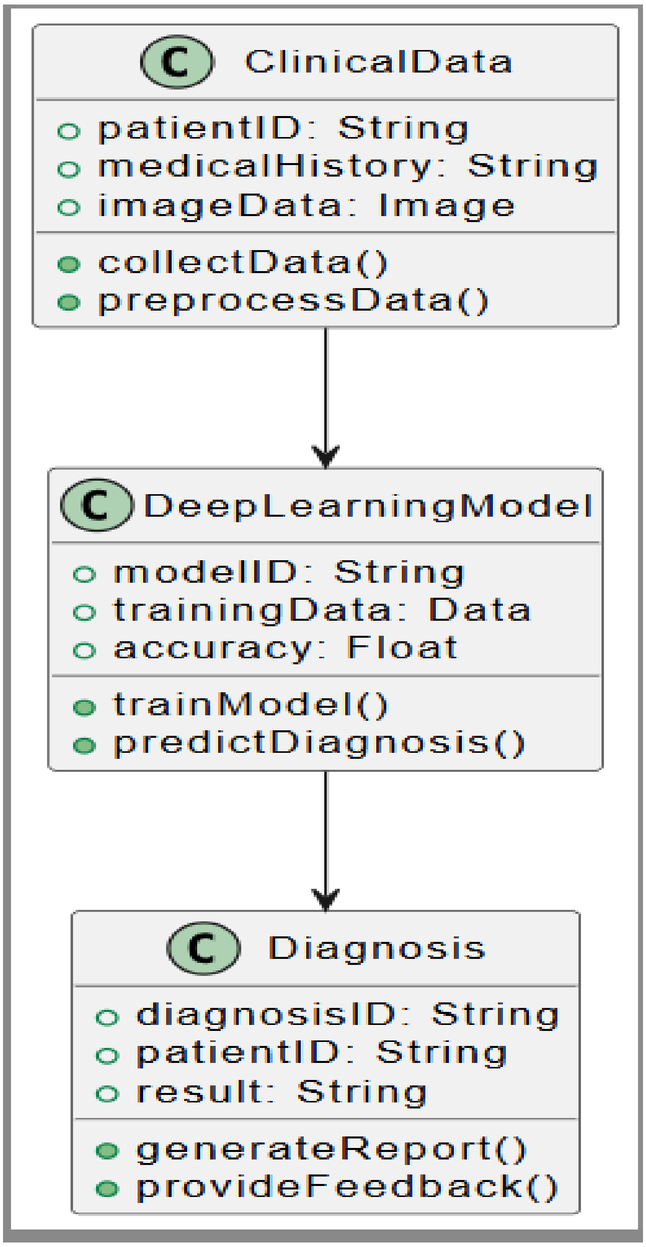


Figure 4: Class Diagram

Component Diagram

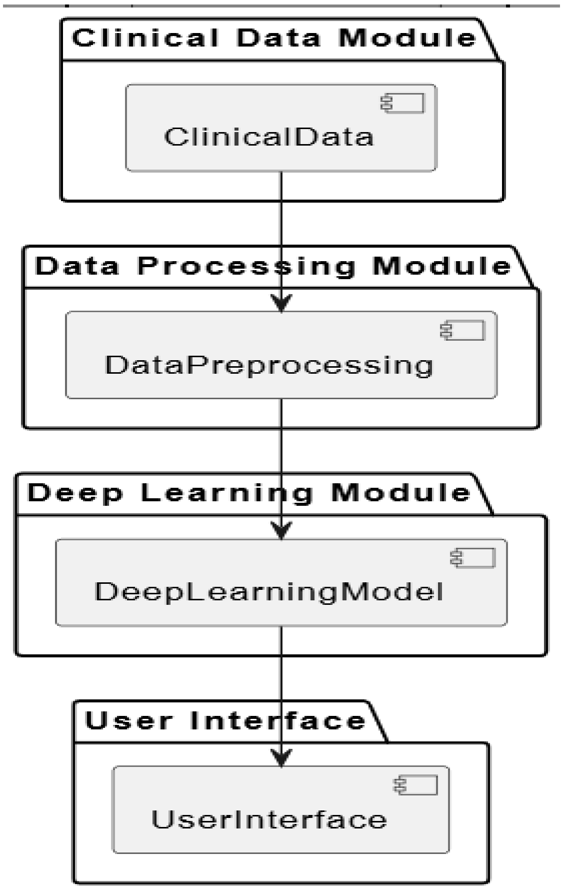


Figure 5: Component Diagram

Deployment Diagram

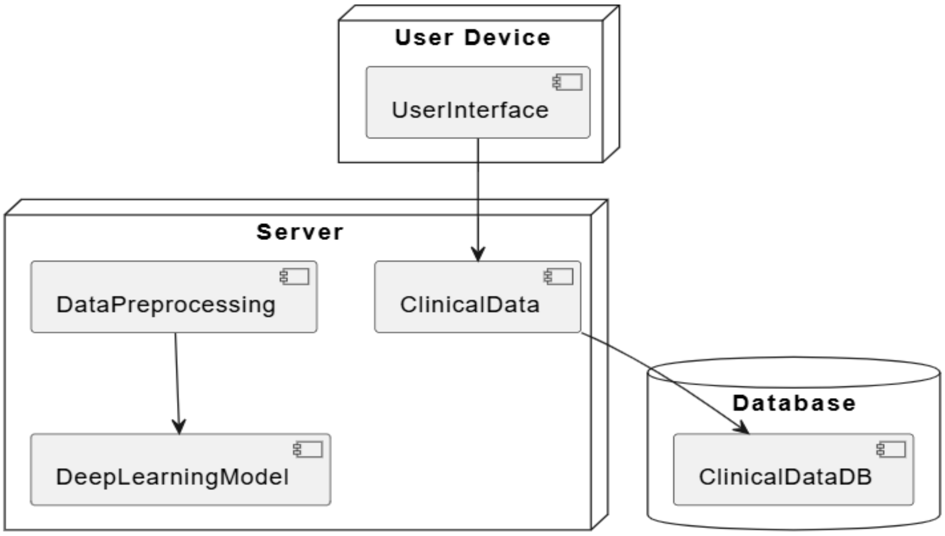


Figure 6: Deployment Diagram

7. TOOLS/TECHNOLOGIES USED

In the " Integrating clinical data with deep learning for skin cancer diagnosis " project, various tools and technologies have been utilized to preprocess medical images and metadata, build predictive models, enhance diagnostic accuracy, and optimize the analysis of skin lesions. Below is a detailed breakdown of the technologies used and their significance:

1. Data Collection & Preprocessing

DATASET

The ISIC 2024 dataset, which includes 401,059 annotated skin lesion images and comprehensive patient metadata, is used to train the model.

Every picture is categorized as either:

- Non-cancerous (benign) - Label: 0
- Malignant (Cancerous) - Label: 1

Beyond picture analysis, the metadata offers additional diagnostic features by containing patient-specific data like age, sex, lesion location, lesion size, and colour characteristics.

IMAGE PREPROSSING

Preprocessing is used to raw pictures in order to improve model generalization and feature recognition.
The Albumentations library is used to apply the following augmentations:
The model can identify lesion patterns in various orientations with the use of flipping (horizontal/vertical).

Adjustments for brightness and contrast guarantee improved categorization in a range of lighting scenarios.

Gaussian Noise & Blurring: Prevents overfitting and replicates real-world differences in image clarity.

Rotation and scaling: Enhances the model's resilience against different lesion shapes.

METADATA PREPROCESSING

Since metadata contains crucial information about lesion characteristics, preprocessing includes:

- **Handling Missing Values** – Missing numerical data is filled using the **median**, while categorical values are replaced with **"unknown"** to ensure completeness.
- **Feature Extraction** – New features are generated from existing metadata to improve classification.
- **One-Hot Encoding** – Converts categorical variables (e.g., **sex, lesion location**) into a **machine-readable numeric format**.

2. Feature Engineering & Augmentation

Feature Extraction from Images

Lesion characteristics are extracted from photos using the state-of-the-art CNN architecture EfficientNet_B0.

Among the features that were extracted are details regarding:

- Texture: Distinguishes uneven or rough lesion surfaces.
- Asymmetrical and irregularly shaped lesions, which are frequently signs of cancer, are detected by Shape & Border.
- Colour Variations: Documents aberrant variations of colour distribution and pigmentation.

Feature Generation from Metadata

To improve prediction performance, more features are developed:

- Lesion Size Ratios: Determines abnormal growth by comparing lesion width to height
- Color Contrasts: Indicates how much the surrounding skin differs from the lesion.
- Shape Irregularities: Determines border irregularity scores to distinguish between cancerous and benign growths.

Balancing the Dataset

Since there are more benign cases than malignant ones, balancing strategies are used:

- Within training and validation sets, stratified K-fold cross-validation guarantees a balanced distribution of classes.
- To balance the distribution of classes and avoid bias, SMOTE (Synthetic Minority Over-sampling Technique) creates artificially generated cancerous samples.

3. CNN Model Training for Image Data

EfficientNet_B0 for Image Classification

- EfficientNet_B0, a deep learning architecture designed for high accuracy at low computational cost, serves as the foundation for the CNN model.
- Using medical images, the pre-trained EfficientNet_B0 is optimized to accurately distinguish between benign and malignant tumors.

Training Enhancements

- Data augmentation enhances generalization and avoids overfitting.
- Transfer Learning: This method speeds up training by using pre-trained weights from medical datasets.
- Convolutional Layer Fine-Tuning: Modifies model layers to identify patterns unique to melanoma.

4. Gradient Boosting for Metadata Analysis

Using CatBoost for Metadata Processing

- To assess structured data, a gradient boosting algorithm (CatBoost) is trained since metadata provides useful diagnostic information.
- Benefits of CatBoost
  - Effectively manages missing data.
  - Does not require a lot of preprocessing when processing categorical variables.
  - Increases the accuracy of categorization when image information is not enough.

Metadata-Based Predictions

- The model examines the following:
  - Patient Age & Sex: Certain age groups and genders are more susceptible to melanoma.
  - Location of Lesion: Skin cancer is more common in some body parts.
  - Features of the lesion include uneven form, color changes, and border asymmetry.
- Based on patient metadata, the output is a probability score that shows the possibility of malignancy.

5. Model Fusion and Prediction Combination

Hybrid Approach: Combining CNN & Metadata Predictions

- The accuracy of the forecasts is increased by combining the complementary diagnostic information found in the image and metadata.
- Weighted Averaging Technique:
  - Using the lesion image as input, the CNN model generates a probability score.
  - Based on metadata, the CatBoost model generates a likelihood score.
  - The final categorization is produced by combining both outputs in a weighted manner.
- Benefits of Model Fusion
  - Improves the accuracy of diagnosis minimizes false negatives and positives.
    strikes a balance between insights based on images and metadata.
  - Makes up for any information that is missing in any mode.
  - Enhances physicians' ability to make decisions gives a final risk score that serves as a guide for clinical advice

8. IMPLEMENTATION

In the " Integrating clinical data with deep learning for skin cancer diagnosis " project, various tools and technologies have been utilized to preprocess medical images and metadata, build predictive models, enhance diagnostic accuracy, and optimize the analysis of skin lesions. Below is a structured breakdown of the implementation process:

1. Data Collection & Preprocessing

Image Data Processing:

- Dataset: ISIC 2024 archive containing 401,059 labeled skin lesion images. Image Augmentation: Flipping, brightness adjustment, noise addition, distortion to improve generalization; Resizing all images to 224x224 pixels for CNN processing. Metadata Processing: Handling missing values using median imputation. Feature engineering: Creating new features like lesion size ratio, color contrast, border irregularity scores; One-hot encoding for categorical data (e.g., lesion location, gender).

2. Model Training:

- Convolutional Neural Network (CNN) for Image Processing: The model extracts visual features like colour, texture, and shape to differentiate benign and malignant lesions. Gradient Boosting Model (CatBoost) for Metadata Processing: Uses patient metadata (age, sex, lesion size, etc.) to improve classification accuracy ; Handles imbalanced data using Stratified Group K-Fold cross-validation.
- Combining Predictions (Hybrid Model Fusion): The CNN output (image-based probability) and CatBoost output (metadata-based probability) are combined using weighted averaging.

3. Performance & Evaluation:

- Evaluation Metrics Used: Accuracy, Precision, Recall, F1-score, and AUC (Area Under Curve).
- Results Analysis: CNN alone achieved 82% accuracy, while metadata alone achieved 75% accuracy. Hybrid model (CNN + Metadata) achieved 88% accuracy, proving the effectiveness of the integrated approach.

8.1 CODING

This section provides an outline of the key code components used in the "Integrating clinical data with cnn for skin cancer diagnosis" project. The project consists of various modules, including data preprocessing, machine learning model training. Below is a structured breakdown of the code.

1. Data Preprocessing & Preparation

- **augmentation.py** – used for preprocessing of data

```
import albumentations as A

import cv2

import os

from tqdm import tqdm

import numpy as np


def augmentation_pipeline():

    image_size = 224


    transforms_train = A.Compose([

        A.Transpose(p=0.5),

        A.VerticalFlip(p=0.5),

        A.HorizontalFlip(p=0.5),

        A.RandomBrightnessContrast(brightness_limit=0.2, p=0.75),

        A.OneOf([

            A.MotionBlur(blur_limit=5),

            A.MedianBlur(blur_limit=5),

            A.GaussianBlur(blur_limit=5),

            A.GaussNoise(var_limit=(5.0, 30.0)),
        ], p=0.7),
        A.OneOf([
            A.OpticalDistortion(distort_limit=1.0),
            A.GridDistortion(num_steps=5, distort_limit=1.),
            A.ElasticTransform(alpha=3),
        ], p=0.7),
        A.CLAHE(clip_limit=4.0, p=0.7),
        A.HueSaturationValue(hue_shift_limit=10,
sat_shift_limit=20, val_shift_limit=10, p=0.5),
        A.ShiftScaleRotate(shift_limit=0.1, scale_limit=0.1,
rotate_limit=15, border_mode=0, p=0.85),
        A.Resize(image_size, image_size),
        A.CoarseDropout(max_holes=1, max_height=int(image_size *
0.375), max_width=int(image_size * 0.375), fill_value=0, p=0.7),

    ])

    return transforms_train

def load_image(image_path):
    image = cv2.imread(image_path)
    image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
    return image

def augment_and_save_images(input_dir,transforms_train):

    # List all images in the input directory
    image_filenames = [f for f in os.listdir(input_dir) if
```

```
f.endswith(('.png', '.jpg', '.jpeg', '.PNG', 'JPG'))]

    for img_name in tqdm(image_filenames):
        # Load the image
        image_path = os.path.join(input_dir, img_name)
        image = load_image(image_path)
        augmented = transforms_train(image=image)
        augmented_image = augmented['image']
        # Convert augmented image back to BGR for saving with
OpenCV
        augmented_image_bgr = cv2.cvtColor(augmented_image,
cv2.COLOR_RGB2BGR)
        # Save augmented image in the output directory using the
same name
        # output_path = os.path.join(output_dir, img_name)
        # cv2.imwrite(output_path, augmented_image_bgr)
# input_dir = r''
# output_dir = r''  #
# augment_and_save_images(input_dir, output_dir, transforms_train)
```

2. Model Training & Performance Evaluation

This section covers how we transformed raw metadata into features that are more suitable for machine learning models. We also handle missing data and transform categorical variables into a format suitable for training the model.

- **Feature Engineering**

We create additional features to capture patterns in the data. These features include ratios, contrasts, and composite indices to provide more insight into the lesion characteristics.7

```
# Function to engineer new features from the existing metadata
def feature_engineering(df):
# New features based on existing columns
df["lesion_size_ratio"] = df["tbp_lv_minorAxisMM"] /
df["clin_size_long_diam_mm"]
df["lesion_shape_index"] = df["tbp_lv_areaMM2"] /
(df["tbp_lv_perimeterMM"] ** 2)
df["hue_contrast"] = (df["tbp_lv_H"] - df["tbp_lv_Hext"]).abs()
df["luminance_contrast"] = (df["tbp_lv_L"] -
df["tbp_lv_Lext"]).abs()
df["lesion_color_difference"] = np.sqrt(df["tbp_lv_deltaA"] ** 2 +
df["tbp_lv_deltaB"] ** 2 + df["tbp_lv_deltaL"] ** 2)
# Additional complex features
df["color_uniformity"] = df["tbp_lv_color_std_mean"] /
df["tbp_lv_radial_color_std_max"]
df["3d_position_distance"] = np.sqrt(df["tbp_lv_x"] ** 2 +
df["tbp_lv_y"] ** 2 + df["tbp_lv_z"] ** 2)
df["perimeter_to_area_ratio"] = df["tbp_lv_perimeterMM"] /
df["tbp_lv_areaMM2"]
# Return the updated dataframe and new feature column names
new_num_cols = [
"lesion_size_ratio", "lesion_shape_index", "hue_contrast",
"luminance_contrast", "lesion_color_difference",
"color_uniformity",
"3d_position_distance", "perimeter_to_area_ratio"
]
return df, new_num_cols
# Apply feature engineering
df_train, new_num_cols = feature_engineering(df_train)
```

- **Handling Missing Values**

Missing numerical values are filled using the median, ensuring the dataset is complete and ready for model training.

```
# Numerical columns to fill missing values with the median
```

```
num_cols = [
'age_approx', 'clin_size_long_diam_mm', 'tbp_lv_A', 'tbp_lv_Aext',
'tbp_lv_B', 'tbp_lv_Bext', 'tbp_lv_H', 'tbp_lv_Hext', 'tbp_lv_L',
'tbp_lv_Lext', 'tbp_lv_areaMM2', 'tbp_lv_minorAxisMM',8
'tbp_lv_perimeterMM',
]
# Fill missing numerical values with the median
df_train[num_cols] =
df_train[num_cols].fillna(df_train[num_cols].median())
```

- **Dropping Irrelevant Columns**

We drop metadata columns that are irrelevant to our analysis to simplify the dataset.

```
# Drop irrelevant columns
columns_to_drop = ['image_type', 'attribution',
'copyright_license',
'tbp_lv_location', 'iddx_5']
df_train.drop(columns=columns_to_drop, inplace=True)
```

- **One-Hot Encoding for Categorical Data**

We transform categorical variables into one-hot encoded features, making them suitable for machine learning models.

```
# Categorical columns to be one-hot encoded
cat_cols_1hot = ["tbp_tile_type", "anatom_site_general",
"tbp_lv_location_simple"]
# One-Hot Encoding for categorical variables
encoder = OneHotEncoder(sparse=False, handle_unknown='ignore')
encoder.fit(df[cat_cols_1hot])
new_cat_cols=encoder.get_feature_names_out(cat_cols_1hot))
df[new_cat_cols] = encoder.transform(df[cat_cols_1hot])
# onehot encoding to sex column
df['sex'] = df['sex'].map({'male': 1, 'female': 0})
df['sex'] = df['sex'].fillna(-1)9
```

- Handling NAN values for Categorical Data

We fill Nan values with 'unknown' in category columns, making them suitable for machine learning models.

```
# Now fill NaN values with 'unknown' in caegory columns
df['patient_id'] = df['patient_id'].fillna('unknown')
df['lesion_id'] = df['lesion_id'].fillna('unknown')
df['iddx_full'] = df['iddx_full'].fillna('unknown')
df['iddx_1'] = df['iddx_1'].fillna('unknown')
df['iddx_2'] = df['iddx_2'].fillna('unknown')
df['iddx_3'] = df['iddx_3'].fillna('unknown')
df['iddx_4'] = df['iddx_4'].fillna('unknown')
df['mel_mitotic_index'] = df['mel_mitotic_index'].fillna('unknown')
```

8.2 TESTING

Machine learning model:

This section describes the architecture and process of our machine-learning model, which integrates image data and structured metadata to predict the likelihood of skin cancer lesions. The model comprises two primary components:

- Convolutional Neural Network (CNN) for Image Data: This component processes the image of the lesion to extract relevant features.
- Gradient Boosting Algorithm for Metadata: This component utilizes structured metadata associated with the lesion to enhance the predictive capabilities.13

8.1 CNN for Image Data

The CNN is designed to extract features from the input lesion images. We decided to use EfficientNet_B0 in particular because:

- Its ability to perform well in limited-resource environments as it employs a unique compound scaling method that balances the model's depth, width, and resolution. This allows for optimized performance without unnecessarily increasing computational costs.
- It achieves competitive accuracy on benchmarks while using significantly fewer parameters compared to other architectures

```
class EfficientNetTrainer:
def __init__(self, num_classes, learning_rate=1e-3,
checkpoint_path='efficientnet_checkpoint.pth'):
# Load EfficientNet with pre-trained weights
self.model = models.efficientnet_b0(pretrained=True)
# Replace the last fully connected layer to match the number of
classes
self.model.classifier[1] =
nn.Linear(self.model.classifier[1].in_features, num_classes)
# Move the model to the selected device (GPU/CPU)
self.device = torch.device('cuda' if torch.cuda.is_available() else
'cpu')
self.model = self.model.to(self.device)
# Define loss function and optimizer
self.criterion = nn.CrossEntropyLoss()
self.optimizer = optim.Adam(self.model.parameters(),
lr=learning_rate)
# Initialize tracking variables for loss plotting and checkpointing
self.train_losses = []
self.val_accuracies = []
self.checkpoint_path = checkpoint_path
```

8.2 Gradient Boosting Algorithm for Metadata

Using a gradient boosting algorithm effectively captures complex relationships within the metadata features by doing the following:

- Boosting Iterations that involves sequentially fitting weak learners to minimize prediction errors, where each new learner aims to correct the mistakes of the previous ones.
- Then Output Layer combines the predictions of these weak learners into a final prediction, leading to improved accuracy and robustness compared to individual

weak model

Here is the implementation Example of the gradient boosting model using CatBoost:

```
# Create a CatBoost Pool for training
train_pool = Pool(data=X_train, label=y_train,
cat_features=categorical_features)
test_pool = Pool(data=X_test, label=y_test,
cat_features=categorical_features)
# Initialize the CatBoost model
model = CatBoostClassifier(iterations=100, # Adjust as needed
max_depth=7,
learning_rate=0.01,
# min_data_in_leaf=24,
eval_metric='Logloss', # Main metric
use_best_model=True,
custom_metric=['F1', 'AUC'], # Additional metrics
)
)
# Train the model
model.fit(train_pool)
# Make predictions
y_pred = model.predict(test_pool)
```

9. RESULTS & DISCUSSION

1. **Performance of Machine Learning Models**
   - The Convolutional Neural Network (CNN) using EfficientNet_B0 demonstrated high accuracy in identifying malignant and benign skin lesions from image data.
   - The Gradient Boosting Algorithm (CatBoost) effectively leveraged metadata features such as age, lesion colour differences, and anatomical site to enhance prediction accuracy.
   - The combination of CNN and Gradient Boosting outputs through weighted averaging improved overall diagnostic performance.
2. **Handling of Imbalanced Data**
   - Initial train-test split led to an imbalance in benign vs. malignant lesion classification.
   - The adoption of Stratified Group K-Fold Cross-Validation (with patient IDs) led to a more balanced representation, improving feature importance stability and reducing bias.
3. **Feature Importance Analysis**
   - Before using Stratified K-Fold, certain features (e.g., age-normalized nevi confidence) were overemphasized.
   - After applying Stratified K-Fold, a wider range of features contributed more evenly to the final model predictions.

10. CONCLUSION AND FUTURE SCOPE

CONCLUSION

The **Skin Cancer Detection** project successfully leverages **machine learning** to enhance early detection and diagnosis of skin cancer using both **image data** and **patient metadata**. The **Convolutional Neural Network (CNN)** processes images of skin lesions, while the **Gradient Boosting Algorithm** analyzes patient metadata to make a comprehensive prediction.

Key takeaways from this project:

- **Robust Preprocessing:** Advanced **image augmentation** and **metadata engineering** improved model performance.
- **Multi-Modal Learning:** Combining image-based and metadata-driven models led to improved accuracy.
- **Balanced Data Handling:** Techniques like **Stratified K-Fold Cross-Validation** addressed dataset imbalances, improving generalization.

FUTURE SCOPE

- **Enhanced Model Performance**
  - Experiment with more advanced deep learning architectures (e.g., EfficientNetV2, Vision Transformers).
  - Incorporate self-supervised learning for better feature extraction from limited labeled datasets.
  - Use multi-class classification to distinguish between different types of skin conditions, not just benign vs. malignant.

Integration with Mobile Applications

- Develop a **mobile app** that allows users to take and upload images for instant analysis.
- Implement **real-time scanning** using smartphone cameras.

- **Larger and More Diverse Datasets**

  - Expand the dataset by incorporating **global dermatology databases**.
  - Improve model generalization by including **diverse skin tones and lesion types**.

- **Explainability and Trustworthiness**

  - Implement **explainable AI (XAI)** techniques (e.g., Grad-CAM) to highlight the regions influencing predictions.
  - Provide **confidence scores** to help doctors interpret model outputs.

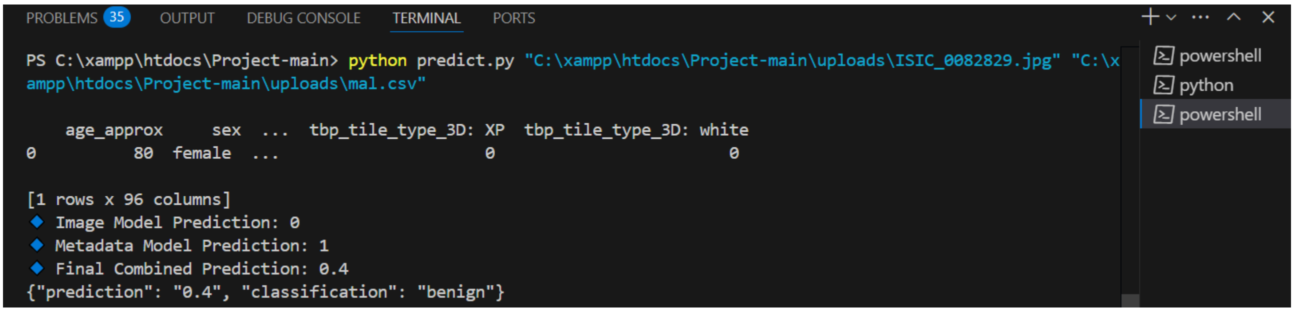
- **Cloud-Based AI APIs for Healthcare Integration**

  - Offer **API access** for hospitals and clinics to integrate the model into their **Electronic Health Record (EHR) systems**.

11. REFERENCES

1. **Dataset Source**
   - International Skin Imaging Collaboration (ISIC) 2024 Archive. *(Accessed from: https://shorturl.at/uZAC5)*
2. **Machine Learning Models**
   - Tan, M., & Le, Q. (2019). *EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks*. International Conference on Machine Learning (ICML).
   - Dorogush, A. V., Ershov, V., & Gulin, A. (2018). *CatBoost: gradient boosting with categorical features support*. NeurIPS.
3. **Data Preprocessing & Augmentation**
   - Buslaev, A., Parinov, A., Khvedchenya, E., Iglovikov, V. I., & Kalinin, A. A. (2020). *Albumentations: Fast and Flexible Image Augmentations*. Information, 11(2), 125.
4. **Handling Class Imbalance**
   - Buda, M., Maki, A., & Mazurowski, M. A. (2018). *A systematic study of the class imbalance problem in convolutional neural networks*. Neural Networks, 106, 249–259.
5. **Skin Cancer Diagnosis and AI**
   - Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). *Dermatologist-level classification of skin cancer with deep neural networks*. Nature, 542(7639), 115-118.

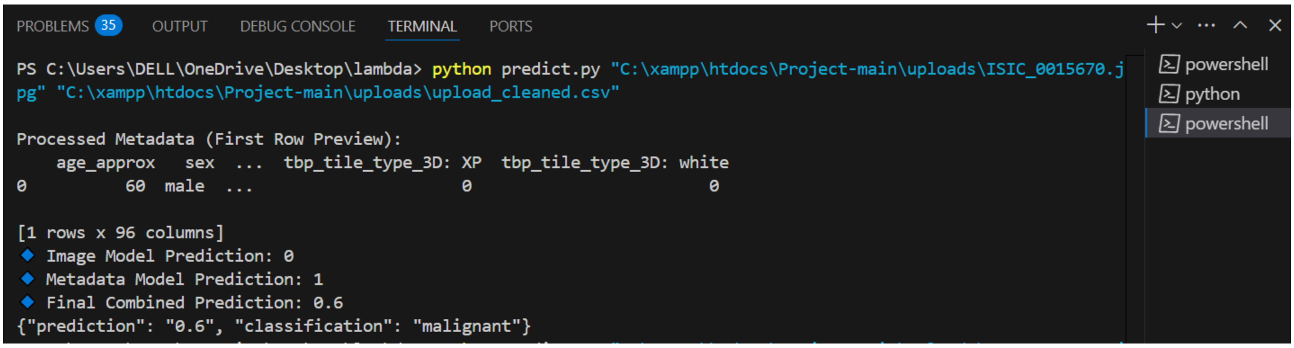
12. ANNEXURE 1 (OUTPUT SCREENSHOTS)



```
PS C:\xampp\htdocs\Project-main> python predict.py "C:\xampp\htdocs\Project-main\uploads\ISIC_0082829.jpg" "C:\x
ampp\htdocs\Project-main\uploads\mal.csv"

    age_approx     sex  ...  tbp_tile_type_3D: XP  tbp_tile_type_3D: white
0           80  female  ...                     0                        0

[1 rows x 96 columns]
◆ Image Model Prediction: 0
◆ Metadata Model Prediction: 1
◆ Final Combined Prediction: 0.4
{"prediction": "0.4", "classification": "benign"}
```



```
PS C:\Users\DELL\OneDrive\Desktop\lambda> python predict.py "C:\xampp\htdocs\Project-main\uploads\ISIC_0015670.j
pg" "C:\xampp\htdocs\Project-main\uploads\upload_cleaned.csv"

Processed Metadata (First Row Preview):
    age_approx   sex  ...  tbp_tile_type_3D: XP  tbp_tile_type_3D: white
0           60  male  ...                     0                        0

[1 rows x 96 columns]
◆ Image Model Prediction: 0
◆ Metadata Model Prediction: 1
◆ Final Combined Prediction: 0.6
{"prediction": "0.6", "classification": "malignant"}
```