

Skin Lesion Classification using Fuzzy-based Ensemble and Decentralised Federated Learning Approaches

submitted by

Siddhant Majumder

Class Roll No. - 002210503029

Exam Roll No. - MCA244028

Master of Computer Application

under the supervision of

Prof. Ram Sarkar

in the

Department of Computer Science and Engineering

Faculty of Engineering and Technology

Jadavpur University

Faculty of Engineering and Technology
Jadavpur University

CERTIFICATE OF RECOMMENDATION

To Whom It May Concern

I hereby recommend that the project report entitled “**Skin Lesion Classification using Fuzzy-based Ensemble and Decentralised Federated Learning Approaches**” has been carried out by Siddhant Majumder (Reg. No.: 163621 of 2022 - 2023, Roll No.:002210503029, Exam Roll: MCA244028), under my guidance and supervision and be accepted in partial fulfilment of the requirement for the degree of MASTER of COMPUTER APPLICATION in DEPARTMENT of COMPUTER SCIENCE and ENGINEERING, JADAVPUR UNIVERSITY during the academic year 2023-2024.

Prof. Ram Sarkar
Project Supervisor
Dept. of Computer Science and Engineering
Jadavpur University

Prof.(Dr.) Nandini Mukhopadhyay
Head of the Department
Dept. of Computer Science &
Engineering
Jadavpur University

Prof.(Dr.) Dipak Laha
Dean, Faculty Council of
Engineering & Technology
Jadavpur University, Kolkata-032

Faculty of Engineering and Technology
Jadavpur University

CERTIFICATE OF APPROVAL

This is to certify that the project report entitled “**Skin Lesion Classification using Fuzzy-based Ensemble and Decentralised Federated Learning Approaches**” is a bonafide record of work carried out by SIDDHANT MAJUMDER, in fulfilment of the requirements for the award of the degree of Master of Computer Application in the Department of Computer Science and Engineering, Jadavpur University. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein but approve the project report only for the purpose for which it has been submitted.

Signature of Examiner 1

Date:

Signature of Examiner 2

Date:

Faculty of Engineering and Technology
Jadavpur University

**DECLARATION OF COMPLIANCE WITH ACADEMIC
ETHICS**

I hereby declare that the project report entitled "Skin Lesion Classification using Fuzzy-based Ensemble and Decentralised Federated Learning Approaches" is an outcome of my genuine and independent work, as part of my Master of Computer Application program.

All information in this document has been obtained and presented in accordance with academic rules and ethical conduct.

I also declare that, as required by these rules and conduct, all materials and results that are not original to this work have been fully cited and referenced.

Siddhant Majumder

Registration No.: 163621 of 2022 - 2023

Examination Roll No.: MCA244028

Signature with Date

Acknowledgements

I want to express my profound gratitude and thanks to Prof. Ram Sarkar for his valuable guidance, suggestions and feedback throughout the working of this project. His expertise and insights have been instrumental in shaping the direction and outcome of this project .

I am also profoundly grateful to Dr. Nandini Mukhopadhyay, HOD of Department of Computer Science and Engineering, for providing the necessary resources and a conducive environment for working on this project.

I sincerely thank Prof. Mahantapas Kundu, Coordinator of the CMATER Research Lab , for ensuring all the technical and logical requirements were effectively met.

My heartfelt thanks to the research scholars, Sujan Sarkar and Diptarka Mandal, and every faculty member and scholar of the CMATER Lab for their valuable inputs, constant collaboration and willingness to share their knowledge and expertise. Their contributions have been extremely helpful and enriching.

I would also like to thank the Department of Computer Science and Engineering faculty for imparting the fundamental knowledge essential for the inception of this project.

Lastly I express my gratitude to my friends and family members for their constant support.

Siddhant Majumder

Examination Roll No.: MCA244028

Registration No.: 163621 of 2022 - 2023

Department of Computer Science and Engineering,

Jadavpur University.

Contents

Table of Contents	i
List of Figures	ii
List of Tables	iv
Abstract	v
1 Introduction	1
1.1 Challenges of Skin Cancer Detection	2
1.2 Motivation and Contributions	3
1.3 Organization of Project Work	4
2 Related Work	5
2.1 HAM10000	5
2.2 ISIC 2016	7
3 Working Methodologies	10
3.1 Fuzzy Based Ensemble of CNN Models	10
3.1.1 Ensembling	10
3.1.2 Ensemble of Models Containing A Self Attention Mechanism	13
3.2 Decentralised Federated Learning	16
4 Results and Discussion	19
4.1 Experimental Setup	19
4.2 Dataset Description	19
4.2.1 ISIC 2016	20
4.2.2 HAM10000	20
4.3 Dataset Preparation	21
4.3.1 ISIC 2016	21
4.3.2 HAM10000	23
4.4 Evaluation Metrics	24
4.4.1 Accuracy	24
4.4.2 Weighted Precision	24
4.4.3 Weighted Recall	25
4.4.4 Weighted F1-Score	26
4.5 Results on the ISIC 2016 Dataset	26
4.5.1 Selecting Models for Ensemble	27

4.5.2	Model Results	28
4.5.3	Results Analysis	29
4.6	Results on the HAM10000 Dataset	31
4.6.1	Selecting Models for Ensemble	31
4.6.2	Model Results	32
4.6.3	Results Analysis	33
5	Conclusion and Future Scope	36
5.1	Conclusion	36
5.2	Scope for Future Work	37
	Bibliography	39

List of Figures

1.1	Basal Cell Carcinoma	1
1.2	Squamous Cell Carcinoma	2
1.3	Melanoma	2
3.1	Overview of Transfer Learning	10
3.2	Overview of the Proposed Model 1	11
3.3	Probability Scores for each Test Image generated by a Single Classifier	13
3.4	Sugeno Ensemble of Probability Scores of a Single Class for One Image	13
3.5	Overview of the Proposed Model 2	14
3.6	Scaled Dot-Product Attention	15
3.7	Get Probability Scores From Models With Attention	16
3.8	Overview of the Decentralised Federated Learning Approach	16
4.1	Pie Chart Distribution of the Original HAM10000 Data	21
4.2	Pie Chart Distribution of the ISIC 2016 Training Dataset	22
4.3	Pie Chart Distribution of Client Training Data for the ISIC 2016 Clients	22
4.4	Pie Chart Distribution of Training Data of the HAM10000	23
4.5	Pie Chart Distribution of Client Training Data for the HAM10000 Clients	24
4.6	Confusion Matrix of 1st Model on the ISIC 2016 Dataset	29
4.7	Confusion Matrix of 2nd Model on the ISIC 2016 Dataset	29
4.8	Confusion Matrix of 3rd Model on the ISIC 2016 Dataset	29
4.9	A benign skin lesion	30
4.10	A malignant skin lesion	30
4.11	Confusion Matrix of 1st Model on the HAM10000 Dataset	33
4.12	Confusion Matrix of 2nd Model on the HAM10000 Dataset	33
4.13	Confusion Matrix of 3rd Model on the HAM10000 Dataset	33
4.14	A basal cell carcinoma skin lesion	34
4.15	A melanoma skin lesion	34

List of Tables

4.1	Source of Datasets Used in the Project	19
4.2	Class-wise Distribution of the HAM10000 Dataset	20
4.3	Class-wise Breakup of the ISIC 2016 DataSet	21
4.4	Individual Classifiers on the ISIC 2016 for selecting base models for Ensemble (bold models are selected for ensemble)	27
4.5	Individual Classifiers on the ISIC 2016 for selecting base models with Self Attention for Ensemble (bold models are selected for ensemble)	27
4.6	Performance Comparison of Proposed Models with some past methods on the ISIC 2016 dataset	28
4.7	Individual Classifiers on the HAM10000 for selecting base models for Ensemble (bold models are selected for ensemble)	31
4.8	Individual Classifiers on the HAM10000 for selecting base models with Self Attention for Ensemble (bold models are selected for ensemble)	31
4.9	Performance Comparison of Proposed Models with some past methods on the HAM10000 dataset	32

Abstract

Early treatment is crucial for recovery from skin cancer. Detection of malignant skin lesions is required for timely diagnosis of skin cancer and to reduce mortality from this disease.

This project explores convolutional neural networks(CNN) to classify skin lesions by leveraging Sugeno Integral for aggregation of CNN classifier prediction scores and self attention mechanism for improving base model performance. The use of Sugeno Integral allows for effective aggregation of multiple CNN models, capturing diverse features and improving classification accuracy. Self attention assigns weighted importance to each region of the image. This allows the model to focus on critical areas of the image leading to better capture of critical patterns. The attention mechanism is integrated with the Sugeno ensemble for improved accuracy.

Due to privacy concerns, skin cancer data cannot be moved to a central location for model training. In addition to the centralized model training, we introduce a collaborative peer-to-peer decentralised federated learning approach to model training without using a central server. This method transfers the model updates between clients while the data remains stored locally.

The proposed models were tested on the ISIC 2016 and the HAM10000 datasets. Ensemble of prediction scores provided better results compared to the base models for both datasets. The enesemble of models with self attention applied provided the best results due to attention improving base performances. The federated learning model was the least performing but still comparable to the ensembles making it an approach worth considering for distributed data.

Keywords: Skin Cancer, Deep Learning, Convolutional Neural Network, Fuzzy measures, Sugeno Integral, Ensemble, Self Attention, Decentralized, Federated Learning

CHAPTER 1

Introduction

Skin cancer is an abnormal growth of cells in skin tissues. It occurs when DNA damage to skin cells triggers mutations which lead the skin cells to multiply rapidly and form malignant tumors.

Skin cancer is the leading cause of cancer globally.[1] Skin cancer's rate of occurrence is increasing worldwide and mortality from it has more than doubled over the decades from 1990 to 2010.[1]

While more common in fair skin populations due to their lower content of melannin, people of all races are vulnerable to skin cancer. It is estimated that melanin in people of color filters out twice as much UV compared to fair skin.[1]

There are three common types of skin cancer:[2]

- Basal cell carcinoma: Usually occurs in sun exposed areas of the body such as neck or face. May appear as a pearly or waxy lump, or a flat, flesh-colored or brown scar-like lesion.

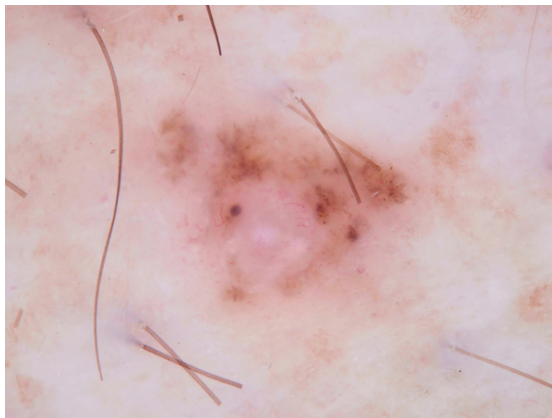


Figure 1.1: Basal Cell Carcinoma

- Squamous cell carcinoma: Occur most often on sun exposed areas of the body such as face, ears and hands. Darker skin more likely to develop squamous cell carcinoma on areas that are not often exposed to the sun. May appear as a firm, red module or a flat lesion with a scaly, crusted surface.



Figure 1.2: Squamous Cell Carcinoma

- Melanoma: Develops anywhere in the body. It starts in the melanocytes which are cells that make the pigment called melanin which give the skin its color.[3]

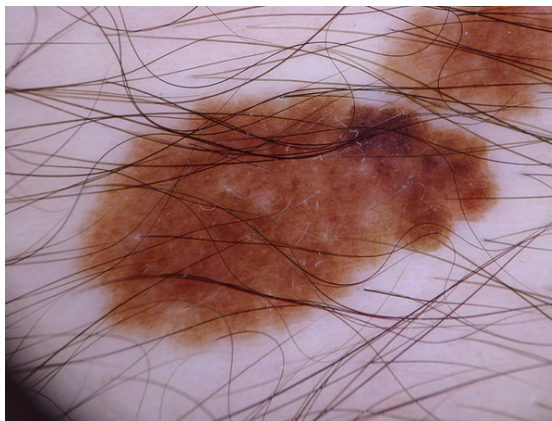


Figure 1.3: Melanoma

Even though melanoma accounts for only 1% of the skin cancer cases, it is responsible for the large majority of skin cancer deaths.[4] This is due to its higher risk of metastasis, i.e, melanoma cancer cells are more likely to break from the original tumor and travel through blood or lymph system and cause new tumor in other organs or tissues of the body.[5]

1.1 Challenges of Skin Cancer Detection

Skin cancer classification is a complex task due to the diverse and intricate nature of skin lesions. Different types of lesions can exhibit visual similarities which causes problems in identifying the correct type of the lesion. Some lesions

can have very subtle differences making classification difficult. Sometimes even within the same type of cancer, lesions can vary widely in appearance leading to miss classification and incorrect diagnosis. Appearance of a lesion is influenced by various factors such as skin color, lesion size, texture and location. This variability proves a challenge in developing a one-size-fits-all classification algorithm.

Any machine learning classification model will have to account for the inter-class similarities and intra-class differences exhibited by the skin lesions to achieve worthwhile accuracy.

1.2 Motivation and Contributions

Early detection is crucial in treatment of skin cancer, especially melanoma. Deep Learning models due to their automatic feature learning ability can be adept at detecting malignant lesions. This will facilitate early detection of malignant lesions.

Deep learning models, especially Convolutional Neural Networks(CNN), have shown remarkable accuracy in image classification tasks. They are adept at learning intricate patterns and features from images. These models automatically learn and extract relevant features from raw images. This why we use CNNs in this project. To create a well performing deep learning neural network, ample and properly labelled data is needed. This can prove a challenge in medical field because patient health data is highly sensitive and highly regulated. To collect and use patient data from hospitals can be unfeasible due to risks of data leaks and threats to patients' privacy. This can be alleviated if instead of sending patient data to research centers, each clinic is given the model architecture to be trained. Once the model is trained on one set of data, its weights can be sent to the other data centers which can fine tune the weights using their own data. Thus the model will be trained on the data of all the centers but without the data being moved to other places and its security intact. This approach is called Federated Learning.

Contributions:

- Fine tuned state of the art pre-trained CNN models to work with skin cancer detection.
- Self attention is used to improve individual model performance.
- Improve the performance of base CNN models by ensemble of prediction scores.
- Decentralised and continual approach to federated learning is explored for peer-to-peer privacy preserving model training by distributed clients.

1.3 Organization of Project Work

This project report is composed of multiple parts. Name and purpose of each part is as follows:

- Chapter 1: Introduction
We give an overview of skin cancer and its most common types. Some challenges in detecting skin cancer are discussed. Motivation for the project is provided.
- Chapter 2: Related Work
This chapter describes some of the past work done on skin cancer classification especially those using the HAM10000 and the ISIC 2016 datasets.
- Chapter 3: Working Methodologies
Theoretical concepts of transfer learning, fuzzy measures, sugeno integral, ensemble using sugeno integral, attention, decentralised federated learning mechanisms are explained. An overview of the algorithms used for implementation is provided.
- Chapter 4: Results and Discussion
This chapter is about the experiments done for the project and the results of these experiments. The dataset preprocessing steps are described and class-wise breakup of images provided. The metrics used are explained and proposed models' performance are compared with related work. Each model's confusion matrix is provided.
- Chapter 5: Conclusion and Future Scope
This chapter concludes the project work and the future scope is discussed

CHAPTER 2

Related Work

In this chapter, a literature survey on two skin cancer datasets, namely HAM10000 and ISIC 2016, has been conducted.

2.1 HAM10000

Ardan Adi Nugroho et al.[6](2019) defined a custom convolutional neural network(CNN) to detect skin cancer using dermatoscopic images. The HAM10000 dataset was used to test the model. Accuracy of train and test images obtained were 80% and 78% respectively.

Agyenta Charity Lariba et al.[7](2022) used transfer learning on InceptionV3, ResNet50 and DenseNet201 to classify and diagnose 7 types of skin liasons using the HAM10000 dataset. The images were pre processed by cropping, color space transformation and removal of hair. The images were resized to 224*224. Densenet201 achieved the best performance with classification accuracy of 99.12% for training and 86.91% for testing.

Muhammad et al.[8](2019) used transfer learning on pre-trained ResNet deep neural network for features extraction then fused and the best features selected using kurtosis controlled principle component (KcPCA). The selected features are fed to a supervised learning classifier such as SVM or RBF. Accuracy of 89.8% was achieved on the HAM10000 dataset.

Emrah ÇEVİK et al.[9](2019) used VGGNET architecture with K-fold cross validation to achieve 85.62% accuracy on the HAM10000 dataset.

Rahman et al.[10](2020) performed weighted ensemble of ResNet, Xception and

DenseNet models trained on the HAM10000 dataset. The individual models achieved balanced accuracy of 78%, 82% and 82% respectively. After ensemble, balanced accuracy of 85.8%, precision of 81%, f1-score of 83%, specificity of 97%, accuracy of 90% and ROC-AUC of 0.94.

Chaturvedi et al.[11](2021) used MobileNet architecture pre-trained on ImageNet and fine-tuned with the HAM10000 images. Accuracy of 83.1%, top2 and top3 accuracies of 91.36% and 95.34% respectively. Also weighted precision of 89%, weighted recall of 83% and weighted f1-score of 83% were obtained.

Khan et al.[12](2024) proposed a framework using 2 main functional blocks: segmentation and classification. Each block has a performance booster which works on principle of information fusion. For segmentation, a hybrid framework using 2 CNN architectures were used. Resultant binary images fused using joint probability distribution and marginal distribution function. For classification a 30-layer CNN architecture was designed and trained on the HAM10000 dataset. A novel summation discriminant correlation analysis technique fused the extracted features from the 2 fully connected layers. “Regular Falsi” feature reduction method was developed to avoid feature redundancy. The selected features were classified using an extreme learning machine classifier. Classification metrics on the HAM10000 dataset were: 87.02% accuracy, 86.98% recall, 13.02% FNR.

Arshad et al.[13](2021) proposed an automated framework using two CNN models ResNet-50 and ResNet-101 and fusing the extracted features using a modified serial-based approach. The fused vector is further enhanced by selecting the best features using the skewness-controlled SVR approach. Final selected features are classified using several machine learning algorithms and the best selected based on accuracy value. The HAM10000 dataset is augmented by performing rotate 90, right-left flip, and up and down flip operations and then used to fine-tune the CNN models. The best results are obtained using Ensemble Subspace Discriminative(ESD) algorithm as classifier with the following results: 91.70% accuracy, 92.00% precision, 91.85% recall, 91.92% f1-score, 8.142% FNR and 0.99 AUC.

Chaturvedi et al.[14](2020) performed multi-class skin cancer classification with five pre-trained CNNs: InceptionV3, ResNetXt101, InceptionResNetV2, Xception, NASNetLarge and fine-tuned using the HAM10000 dataset and four ensemble methods: InceptionV3+Xception, InceptionResNetV2+Xception, InceptionResNetV2+ResNetX101 and InceptionResNetV2+ResNetXt101+Xception. Best results were obtained with the ensemble of InceptionResNetV2+ResNetX101 with

the following results: 92.83% accuracy, 84.00% f1-score, 83.00% precision and 84.00% recall.

Sae-Lim et al.[15](2019) proposed a modified MobileNet architecture for skin cancer classification using the HAM10000 dataset. The last 5 layers of MobileNet architecture were replaced by a Dropout Layer and Fully Connected Layer which used Softmax activation function. Two experiments were performed on the model: first experiment trained the model without using up sampling and data augmentation, second experiment trained the model with up sampled and augmented data. First experiment had better accuracy of 83.93% but rest of the metrics were better in the second experiment with specificity of 87%, sensitivity of 85% and f1-score of 82%.

2.2 ISIC 2016

Mohammed A. Al-masni et al.[16](2020) proposed an integrated diagnostic framework that combined a skin lesion boundary segmentation stage with a multiple skin lesions classification stage. Full resolution convolutional network (FrCN) was used to segment the skin lesion boundaries from the entire dermoscopy images. A convolutional neural network classifier (CNN) was applied on the segmented skin lesions for classification. Inception-v3, ResNet-50, Inception-ResNet-v2, and DenseNet-201 have weighted prediction accuracies of 77.04%, 79.95%, 81.79%, and 81.27% respectively.

Jianpeng Zhang et al.[17](2019) designed a Synergic Deep Learning Model(SDL) which enables multiple deep convolutional neural networks(DCNNs) to mutually learn from each other. Each pair of DCNNs had their learned image representations concatenated as input of the SDL network. SDL has a fully connected structure which predicts whether the input belongs to the same class. Mismatch in the predictions of the DCCN pair causes a synergic error causing the model to update. Model trained end-to-end under supervision of classification errors from DCNNs and synergic errors from DCNN pairs. When using 2 DCNN models for SDL on the ISIC 2016 dataset, average precision, accuracy, Area Under the Curve (AUC) of 0.6536, 0.8522, 0.8139 were obtained without data augmentation respectively. With data augmentation, 0.6644 average precision, 0.8575 accuracy and 0.8179 AUC was obtained. For SDL with 3 models, 0.6608 average precision, 0.8549 accuracy and 0.8149 AUC without data augmentation were obtained. With data augmentation, 0.6713 average precision, 0.8602 accuracy and 0.8371 AUC were obtained. When 4 models were used for SDL, 0.6704 average precision,

0.8575 accuracy and 0.8287 AUC obtained without data augmentation and 0.6810 average precision, 0.8628 accuracy and 0.8224 AUC obtained with augmentation.

Himanshu K. Gajera et al.[18](2022) proposed a method using patch-based local deep feature extraction. A pre trained CNN extracts features from different patches of a dermoscopic image. The extracted features are fused to preserve fine details. Kernel principal component image analysis is employed to select prominent features. Finally, feed forward neural network detects skin cancer. VGG19 being used as CNN, the proposed method yields 86.28% accuracy, 46.67% sensitivity and 96.05% specificity.

Demyanov et al.[19](2016) used an 8-layer deep convolutional neural network on the ISIC 2016 dataset. Using the subset of images divided on segments of various shapes and sizes and labelled on the presence or absence of typical networks or regular globules dermoscopic patterns, two datasets of segments(networks or regular globules) were constructed. Accuracy on network examples was 88% and 83% for globules

Guo et al.[20](2018) proposed a multiple convolutional neural network (MCNN) strategy to train separate models using an additive sample learning strategy. An accuracy of 85.22% and area under the curve(AUC) of 0.81 was achieved for the test set.

Zhang et al.[21](2018) proposed a synergic deep learning model(SDL) which enables dual deep convolutional neural networks(DCNN) to learn from each other. The image representation learned by the 2 DCNNs are concatenated and made the input of a synergic network which has a fully connected structure and predicts whether the pair of input images belong to the same class. The SDL achieved average precision of 66.44%, accuracy of 85.75% and AUC of 0.8179

Ding et al.[22](2021) proposed a Deep Attention Branch Network (DABN) model, which introduced the attention branches to expand the conventional deep convolutional neural networks. A novel Entropy-guided Loss Weighting (ELW) strategy was used to counter class imbalance influence. Average precision of 71.90 was obtained on the ISIC 2016 dataset and average area under the ROC curve (AUC) of 0.922 was obtained on the ISIC 2017 dataset.

Yi et al.[23](2018) proposed a categorical generative adversarial network to automatically learn the feature representation of dermoscopy images in an unsu-

pervised and semi-supervised manner. The ground truth segmentation mask was applied to the ISIC 2016 images to extract the smallest square region containing the lesion and then resized to 256*256 using bilinear interpolation. The images in the training set were further augmented with rotation (in the range of $[-180^\circ, 180^\circ]$), horizontal and vertical flipping, and elastic transform. PH2 dataset was used for validation. Dataset for unsupervised training had 20k images. 70 images were randomly selected for semi-supervised training. The best results obtained on the ISIC 2016 were: average precision of 42.40% for semi-supervised, AUC of 0.690 for semi-supervised, accuracy of 81.20% for unsupervised.

He et al.[24](2022) proposed a deep metric attention learning CNN (DeMAL-CNN). In DeMAL-CNN a triplet-based network (TPN) is designed based on deep metric learning, which consists of three weight-shared embedding extraction networks. TPN adopts a triplet of samples as input and uses the triplet loss to optimize the embeddings. A mixed attention mechanism considering both the spatial-wise and channel-wise attention information was designed and integrated into the construction of each embedding extraction network. On the ISIC 2016 dataset, this model achieved accuracy of 87.30%, sensitivity(recall) of 65.30%, specificity of 0.947, AUC of 0.858 and average precision of 69.70% were obtained.

Burdick et al.[25](2017) proposed extending segmentation border to include pixels surrounding the target lesion. These segmented regions were extracted and used as input for two CNNs: VGG16 and InceptionV3. For segmentation, dilated masks were created through a morphological dilation using the original raw images and binary masks provided by the ISIC 2016 dataset. InceptionV3 gave the best results at 75 dialtion with accuracy of 69.30%, sensitivity(recall) of 76.00%, precision of 72.30% and AUC of 0.739

Working Methodologies

This section delves into the theoretical concepts used in the project. Transfer learning, fuzzy measures based ensemble, attention mechanism, decentralised federated learning are explained. Pre trained torchvision models are reused for the skin lesions datasets. This is a common approach in computer vision tasks since the pretrained models have been trained with millions of images. The general features which have been extracted by the models are reused for training with other datasets which do not have as many images. This approach of retraining pre-trained deep learning models using new data is called **Transfer Learning**. The final layer of the model is replaced by a new dense layer which outputs softmax values for each class of the new dataset.

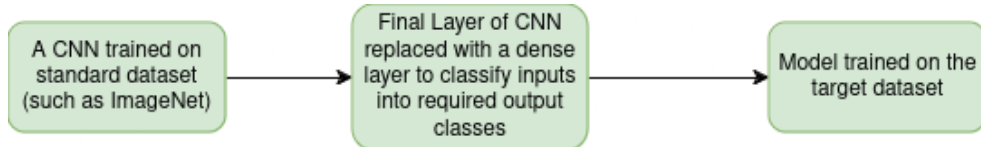


Figure 3.1: Overview of Transfer Learning

3.1 Fuzzy Based Ensemble of CNN Models

3.1.1 Ensembling

Of the base models we trained, three with the highest test accuracy are selected and combined. This is done to get a collective model which will have better overall performance. In this project we will ensemble the prediction scores for each test image by all three models using Sugeno Integral. This type of integral uses fuzzy measures. Fuzzy measures are used for the ensemble process to adapt the weights of each classifier's prediction scores based on their importance while testing instead of assigning pre defined weights to each classifier's predictions.

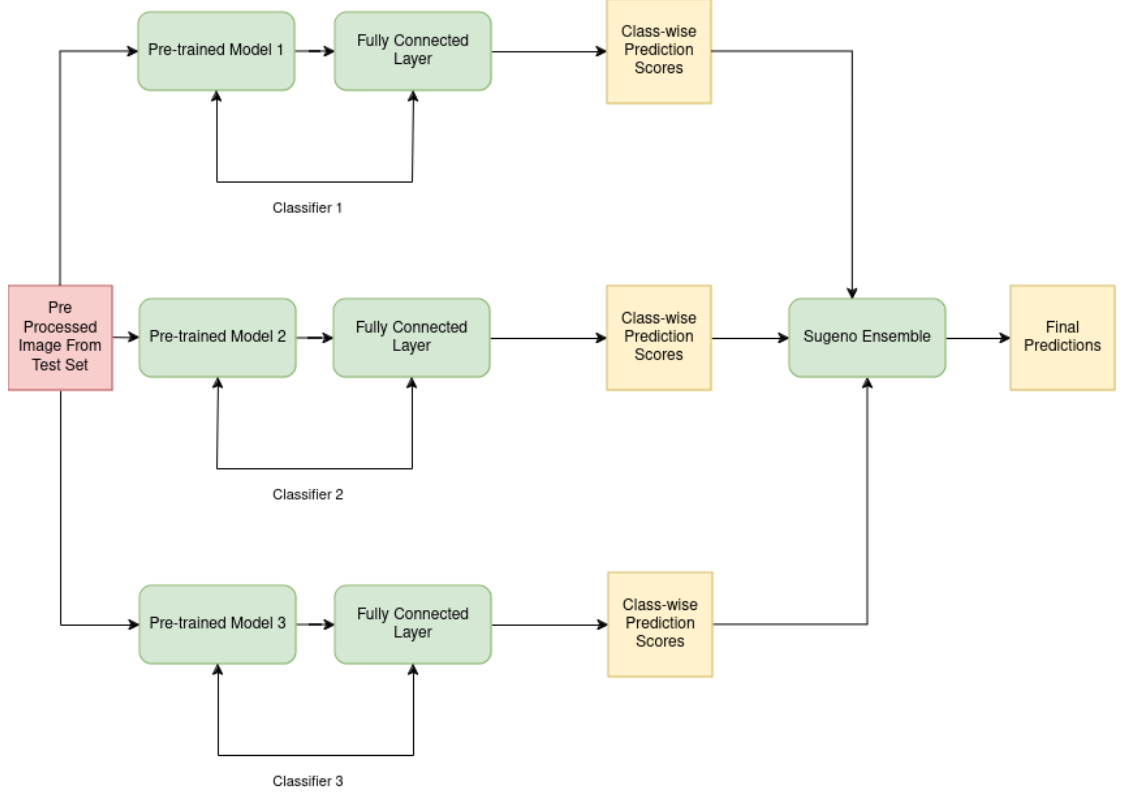


Figure 3.2: Overview of the Proposed Model 1

A. Fuzzy Measures

A fuzzy measure is the set function f that satisfies the following properties[26]:

1. $f(\phi) = 0, f(X) = 1$
2. $A, B \in \beta$ and $A \subseteq B$, implies $f(A) \leq f(B)$
3. If $C_n \in \beta$, $C_1 \subseteq C_2 \subseteq C_3 \subseteq \dots \subseteq C_n$, then $\lim_{n \rightarrow \infty} f(C_n) = f(\lim_{n \rightarrow \infty} C_n)$

where β is a Borel field of an arbitrary set X .

B. Sugeno fuzzy- λ measures

Tahani et al.[27] described Sugeno fuzzy- λ measures using a set of scores $S = \{s_1, s_2, s_3, \dots, s_n\}$ where N is the number of information sources and $e \in S$. The Sugeno- λ measure is the function $f_\lambda : 2^S \rightarrow [0, 1]$ such that it satisfies the following conditions:

1. $f_\lambda(S) = 1$
2. if $e_i \cap e_j = \phi$, then $\exists a\lambda > -1$, such that, Equation 3.1 holds true.

$$f_\lambda(e_i \cup e_j) = f_\lambda(e_i) + f_\lambda(e_j) + \lambda \cdot f_\lambda(e_i) \cdot f_\lambda(e_j) \quad (3.1)$$

Therefore λ is the real root of the equation as given in Equation 3.2

$$\lambda + 1 = \prod_{n=1}^N (\lambda \cdot f(e_n) + 1) \quad (3.2)$$

C. Sugeno Integral

Sugeno Integral is a fuzzy measure integral defined as follows:[26]

Let A, μ be a measurable space and $f : X \rightarrow [0, 1]$ be a μ -measurable function.

Then the Sugeno Integral of the measurable function f with respect to a fuzzy measure ω is as follows:

$$\int f(x) d\omega = \max_{1 \leq i \leq n} (\min (f(x_i), \Omega(A_i))) \quad (3.3)$$

where $\Omega(A_i) = \Omega(\{x_i, x_{i+1}, x_{i+2}, \dots, x_n\})$ and $f(x_1), f(x_2), f(x_3), \dots, f(x_n)$ are the ranges defined as $f(x_1) \leq f(x_2) \leq f(x_3) \leq \dots \leq f(x_n)$

After fine-tuning, predictions from n classifiers with the highest accuracy on test set are ensembled. This project used three classifiers for ensemble. This is achieved using the Sugeno Integral defined previously. Each image from the test set is passed through each of the three classifiers one-by-one. Each classifier outputs a set of probability scores indicating the likelihood of the image belonging to each output class. Probability scores from all classifiers are stored in different csv files where each row is a set of probability scores for an image from the test set.

Fuzzy measures are calculated based on cardinality, each measure is calculated as:

$$\text{measure}[x] = (x/N)^p \quad (3.4)$$

Where:

- N : number of base learners
- measure : A list of fuzzy measures. Length equal to N
- x : an integer in range $[N, 1]$
- p : parameter which defines scaling factor for the measures. 2 by default

For every test sample, we iterate through each class and store the probability score of that class for the test image from each classifier into a list. Let the list be X . So the list is of length of X is equal to N . X is sorted then aggregated along with measure using minimum function. Then the maximum value along the specified axis is selected as the probability score of the test image belonging to that specific

class by the ensemble.

When scores for all the classes are calculated, the final prediction is computed by selecting the class which has the maximum prediction score for the specific test image.



Figure 3.3: Probability Scores for each Test Image generated by a Single Classifier

This process is repeated for all the images in the test set. After classifying all the images, we calculate the metrics using the final predictions.

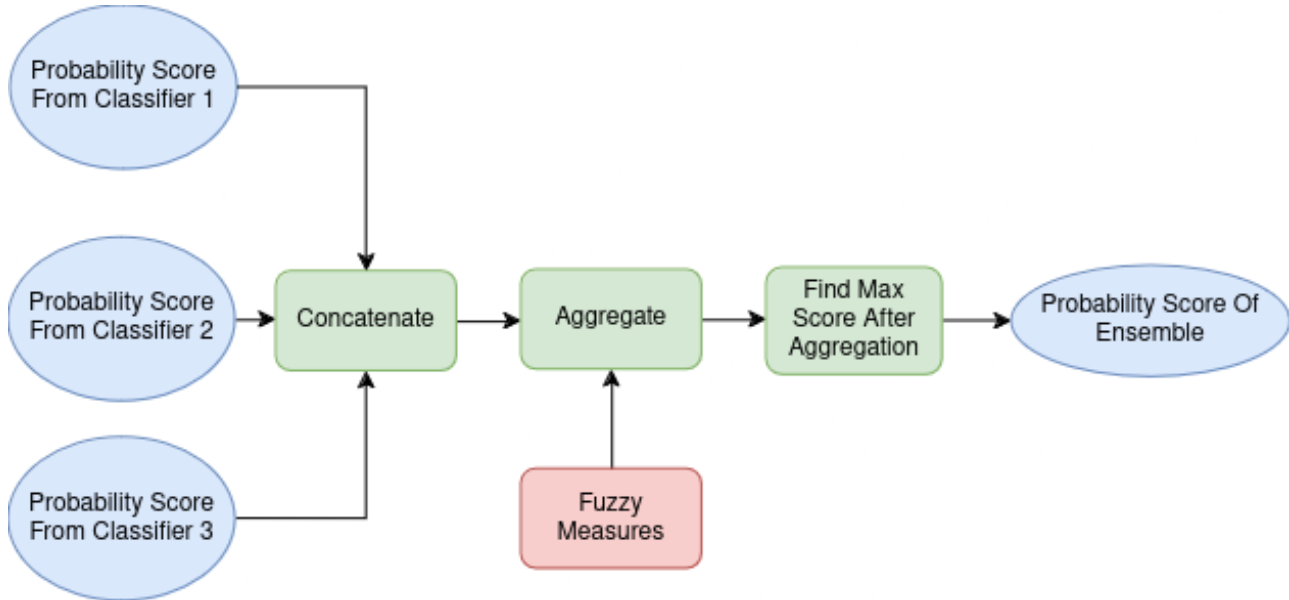


Figure 3.4: Sugeno Ensemble of Probability Scores of a Single Class for One Image

3.1.2 Ensemble of Models Containing A Self Attention Mechanism

While attention mechanism was originally created for natural language processing (NLP) problems, attention finds hearty use in computer vision. Attention methods enable the model "to pay attention" to certain parts of input data, i.e, give increased weightage to certain portions of the data. Attention method is useful for image classification since it can increase the weights of those portions of the image which actually determine the output class, like give more weightage to the lesion area in the dermoscopic image compared to the surrounding skin.

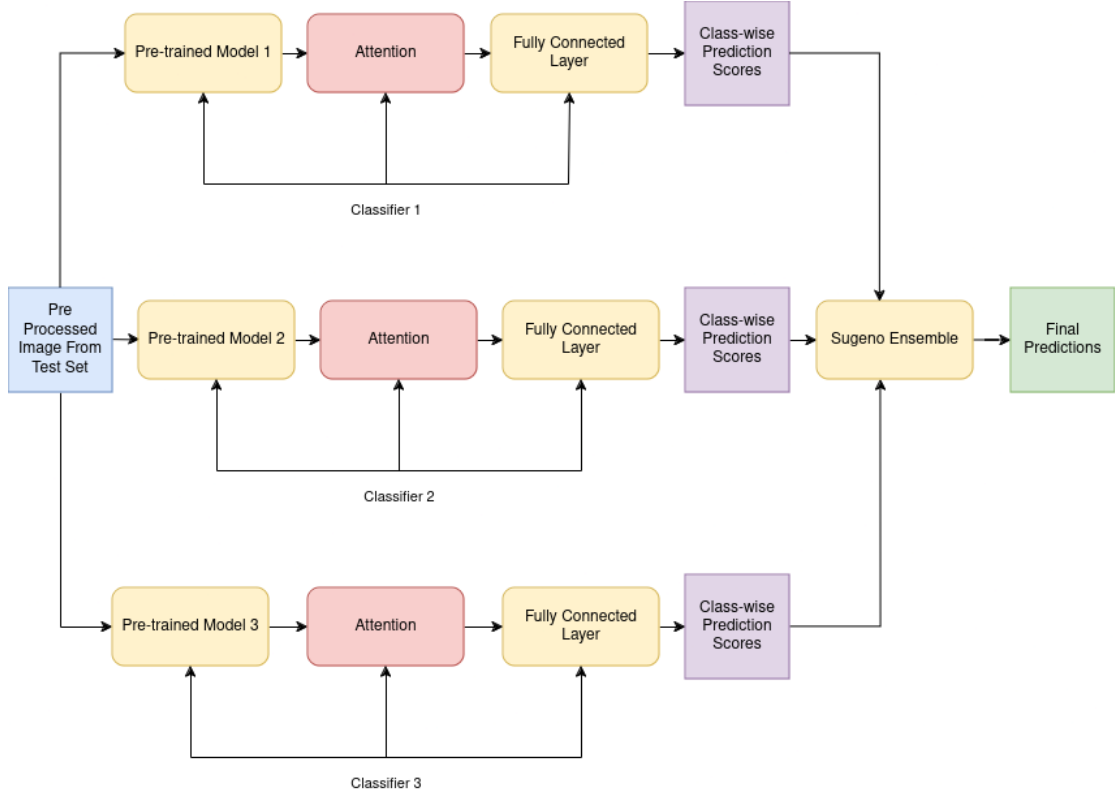


Figure 3.5: Overview of the Proposed Model 2

A. Self Attention

Self Attention allows the model to weigh the importance of different parts of the input data when making predictions. Instead of applying fixed filters to local regions of the input, self attention computes attention scores for each pair of elements in the input. The attention scores reflect how much one element should "pay attention" to another element.

The self attention mechanism implemented in this project is Scaled Dot Product Attention.

B. Scaled Dot-Product Attention

The paper *Attention Is All You Need*[28] introduced Scaled Dot-Product Attention. The formula is as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V \quad (3.5)$$

where:

- Q: Query matrix
- K: Key matrix
- V: Value matrix

- d_k : dimensionality of key vectors

In the context of this project, we perform image classification which is a computer vision task. To calculate the Query, Key and Value matrices, we use the image features extracted from the CNN. This is done because the extracted features capture hierarchical representations of the input image.

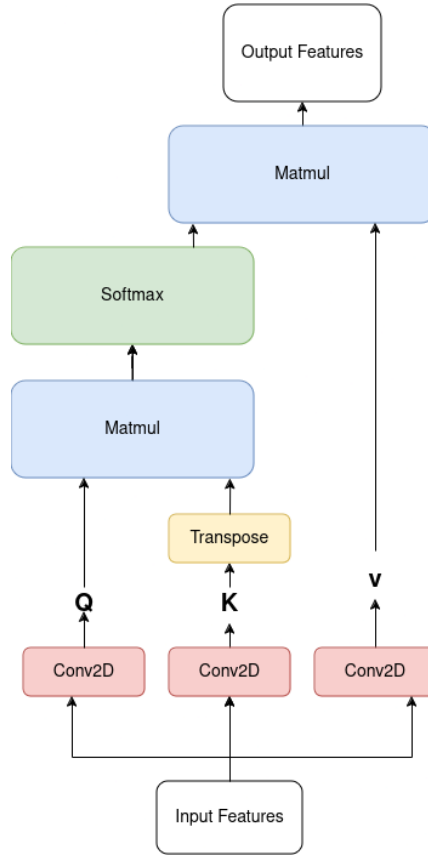


Figure 3.6: Scaled Dot-Product Attention

Depending on the shape of features extracted, they may need to be reshaped to facilitate matrix multiplication. Three separate convolutional layers are defined which will yield the query, key and value matrices when they are applied to the input features.

Dot product of the query and key vectors followed by a softmax operation gives the attention weights. This set of attention weights indicate the importance of different spatial locations in the input.

Multiplying the attention weights with the value vector yields the attended feature representations.

To produce the output of the attention layer, we multiply the attended feature representation with a learnable parameter gamma.



Figure 3.7: Get Probability Scores From Models With Attention

3.2 Decentralised Federated Learning



Figure 3.8: Overview of the Decentralised Federated Learning Approach

Federated Learning was first introduced in 2016 by Google Researchers McMahan et. al[29]. It was applied on Google Keyboard to build a language model collaboration framework to learn contextual information and whether people are clicking on the recommendations.

Since then federated learning has found use in several areas such as healthcare, internet of things, manufacturing, agriculture, remote sensing and more. Federated Learning allows breaking of geographical boundaries allowing better collaboration and access to more diverse training data since data does not need to be transferred to different locations.

Originally, the Federated Learning approach introduced by McMahan et. al [29] used a central server and multiple clients. The clients receive the model architecture from the server. The weights from the local models of the clients are aggregated in the server using federated averaging(FedAvg). Training the model consists of several communication rounds. Each client individually train their local models on their own data. When completed, the clients upload their local models to the server. The server weighted averages all the local models. The weights are set based on the number of client samples.

The aggregated model weights are then used as initial weights for the clients for next round. This process is repeated for a fixed number of rounds. Figure ?? demonstrates the communication between client and server.

The approach implemented in this project does not make use of a single server. Rather, we set up communications between the clients to allow model weights to be transferred from one client to another.

This approach is called **Decentralised Federated Learning**.. The advantage of

this type of Federated Learning is to create a fully autonomous network between the clients for model training without any need of intervention from a central server.

We can use various network topology for connecting the clients[30]. The topology used in this project is line. One client is only connected to a single client resulting in a line like connection being formed. The first client trains the local model using its local data. After training, the model weights are sent to second client. Model weights of local model of second client are sent to 3rd client after training and so on. After the final client is trained, the weights are sent to each of the previous models one-by-one through the network. The line topology has the advantage of low communication overhead since only direct communication between 2 clients need to be handled. Scaling is also much easier in a line topology since new client only needs connection to be set up with one other client.

The training could be done using either one of two paradigms: *Continual* or *Aggregate*.

Aggregate decentralised federated learning aggregates the models of all clients much like centralised federated learning but without the use of a server. Each client receives model updates of all the other clients which have undergone training before it. The client aggregates these models like a server would in centralised federated learning. Then the weights of the aggregated models are used as initial weights for client's local model and fine-tuned on the current client's data. The weights formed after training are sent to the next client along with the weights from all previous models.

We use a continual paradigm for training the clients. A client receives model weights from the previous client. These weights are set as the initial weights of the current client's local model. The model is then fine-tuned on the client's local data. The client sends the model weights to the next client. The paradigm used here is sequential since the training ends when the final client finishes training. If the final client were to send its weights back to the 1st client forming a ring, the paradigm would be cyclical.

Algorithm for the sequential continual decentralised federated learning is as follows:

Input: Client set (C), training epoch (E), initial model (ω_0), loss function (L), learning rate (η)

Output: Local models $\{\omega_c | c \in C\}$

for c **in** C **in sequence do**

 Copy the model from the previous client : $\omega_c \leftarrow \omega_{c-1}$

for $e = 1$ **to** $E - 1$ **do**

Backpropagate and update the local model: $\omega_{e+1}^c \leftarrow \omega_e^c - \eta \nabla L$

end for

Update the local model: $\omega_c \leftarrow \omega_E^c$

Client c sends ω_c to the next client

end for

The above algorithm shows that no data is transferred between clients. Only model weights are transferred. Patients' privacy is preserved since it stays with the client.

This project uses three clients for the demonstration but more clients can be added.

Results and Discussion

4.1 Experimental Setup

The proposed models have been trained and evaluated on the ISIC 2016 and the HAM10000 datasets. Pre-trained CNNs with the ImageNet default weights have been used and trained with a dense layer added to classify the skin lesions. The models have been trained for 30 epochs with initial learning rate of 0.0001 and SGD optimizer except for federated learning where each client is trained for 20 epochs. The 16GB P100 GPU provided by Kaggle has been used for training the neural network.

4.2 Dataset Description

The two datasets used in the project are mentioned in Table 4.1

Dataset Name	URL	Location of the dataset
ISIC 2016	https://challenge.isic-archive.com/data/	isic-archive
HAM10000	https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/DBW86T	Harvard Data-verse

Table 4.1: Source of Datasets Used in the Project

We describe the original breakdown of the ISIC 2016 and HAM10000 datasets as obtained from the provided locations in Table 4.1.

4.2.1 ISIC 2016

The ISIC 2016 dataset has 5 different sets of skin lesion images for the particular tasks which were present in the ISIC 2016 challenge. For this project we will use the images for task 3. It has 900 dermoscopic lesion images in JPEG format for training and 379 images of same format for testing.

There are 2 different classes of images: benign and malignant. The training set has 726 benign images and 173 malignant images. The test set has 303 benign images and 75 malignant images.

4.2.2 HAM10000

There are total 10,015 dermoscopic images in the HAM10000 dataset. This dataset was used for 2018 ISIC classification challenge. The images are split into 7 classes representing important diagnostic categories of pigmented lesions: Actinic keratoses and intraepithelial carcinoma / Bowen’s disease (akiec), basal cell carcinoma (bcc), benign keratosis-like lesions (solar lentigines / seborrheic keratoses and lichen-planus like keratoses, bkl), dermatofibroma (df), melanoma (mel), melanocytic nevi (nv) and vascular lesions (angiomas, angiokeratomas, pyogenic granulomas and hemorrhage, vasc). Number of images in each class is given in Table 4.2

Class	Number of Images
akiec	327
bcc	514
bkl	1099
df	115
mel	1113
nv	6705
vasc	142

Table 4.2: Class-wise Distribution of the HAM10000 Dataset

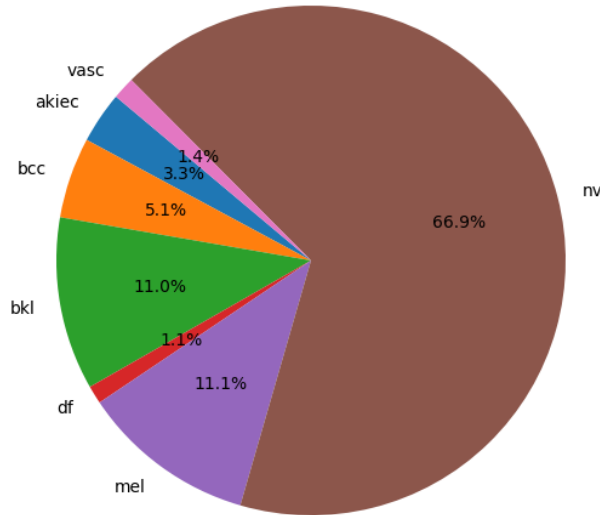


Figure 4.1: Pie Chart Distribution of the Original HAM10000 Data

4.3 Dataset Preparation

This section provides an overview of the pre-processing done on the two datasets and gives a class-wise breakup of the images used for training and testing.

4.3.1 ISIC 2016

The dataset is split into train and test folders with their labels in csv file. We read the csv to organize the images into 2 folders based on their class label. Benign images go into "benign" folder and malignant images into "malignant" folder. This is done inside both train and test folders. For validation 20% of the train images are used. The remaining 80% of images in train folder are used for training the model.

Class-wise breakup of images is given in Table 4.3

Class	Train	Test
benign	726	303
malignant	173	75

Table 4.3: Class-wise Breakup of the ISIC 2016 DataSet

For decentralised federated learning model, we need the clients to each have their own separate data to simulate real world scenario. For this we split the images in train folder between each client. We need 3 different sets of training

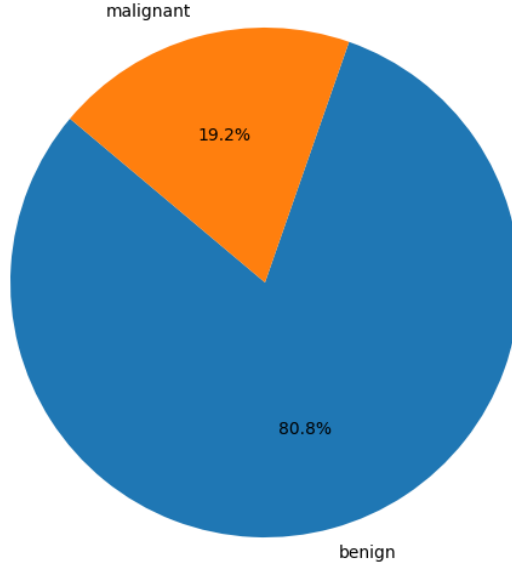


Figure 4.2: Pie Chart Distribution of the ISIC 2016 Training Dataset

images since there are 3 clients. Malignant images have been augmented due to their relatively low number to benign images (173 malignant images and 726 benign images). The number of malignant images is increased to 495 and each client gets 165 malignant images and 239 benign images. These are used for training the clients. Testing is done on same test folder for each client so that final result can be compared with the other models. When loading the images, they are first

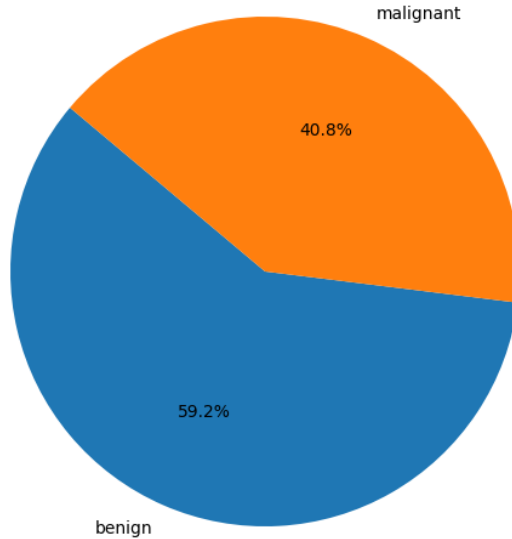


Figure 4.3: Pie Chart Distribution of Client Training Data for the ISIC 2016 Clients

resized to 224x224, converted to tensors and then normalised using mean and std arrays of values: [0.485, 0.456, 0.406], [0.229, 0.224, 0.225]. Normalization process for an image tensor is performed channel-wise on each pixel value. Mathematically,

it is represented in Equation 4.1

$$\text{Normalized Value}_c = \frac{\text{Original Value}_c - \text{Mean}_c}{\text{Standard Deviation}_c} \quad (4.1)$$

4.3.2 HAM10000

The dataset is split into train and test folders in a 80:20 ratio. The test folders had some duplicate images which were moved to train folder. The number of images in the train folder is increased through augmentation to correct the class imbalance and get more input images for training. The total number of images are 53,115 with 52,012 images used for training and 1,103 images used for testing.

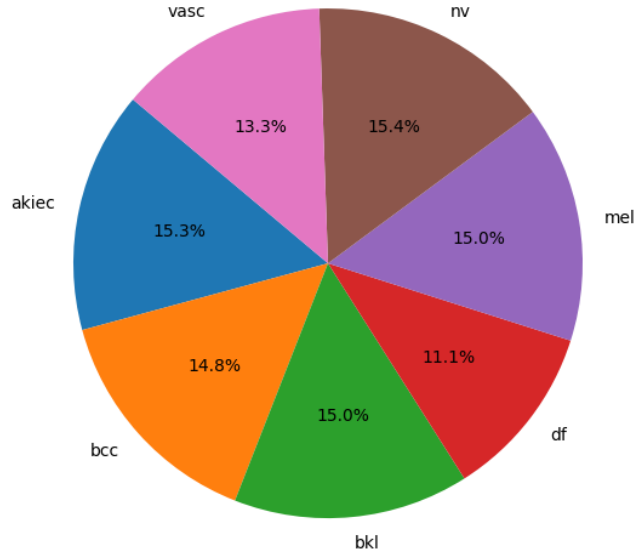


Figure 4.4: Pie Chart Distribution of Training Data of the HAM10000

Three clients are used for decentralised federated learning. The train images are split evenly between the clients. For testing the original test set is used to compare performance with other proposed models.

Class-wise breakup of train data images of each client:

- akiec: 2630
- bcc: 2545
- bkl: 2566
- df: 1904
- mel: 2580
- nv: 2647

- vasc: 2289

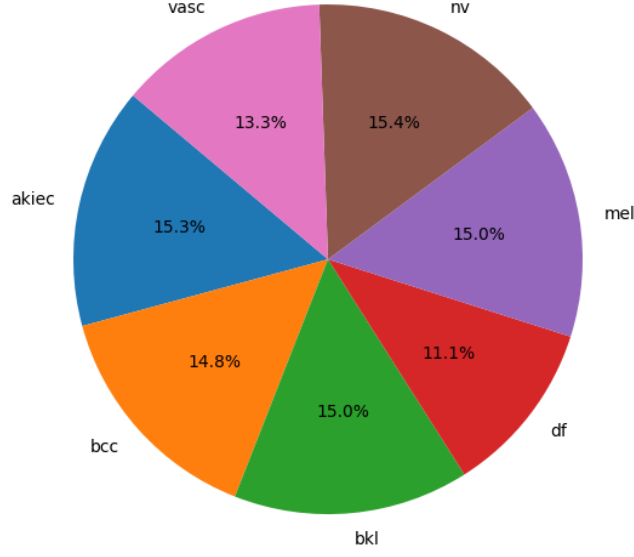


Figure 4.5: Pie Chart Distribution of Client Training Data for the HAM10000 Clients

The images before being fed to the model are resized to 224x224, converted to tensors and then normalized using mean and std arrays with values: [0.485, 0.456, 0.406] and [0.229, 0.224, 0.225] respectively. The normalization process is as described in Equation 4.1

4.4 Evaluation Metrics

To analyse model performance in this project, we will use Cross-Entropy Loss, Accuracy, Weighted Precision, Weighted Recall and Weighted F1-Score.

4.4.1 Accuracy

This is a performance metric which measures the proportion of correctly classified instances out of the total number of instances in context of classification. Mathematically, accuracy is calculated as the ratio of the number of correctly predicted instances to the number of instances.

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Instances}} \quad (4.2)$$

4.4.2 Weighted Precision

Precision measures the proportion of true positive predictions (correctly predicted positive instances) out of all instances predicted as positive by the classifier.

It quantifies the classifier's ability to make correct positive predictions while minimizing false positives.

Mathematically,

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positives} + \text{False Positives}} \quad (4.3)$$

Where:

- True Positives(TP): instances correctly classified as positive by the classifier.
- False Positives(FP): instances incorrectly classified as positive by the classifier

Weighted Precision is a variation of the above precision that accounts for class imbalance in multi-class classification tasks. Here precision for each class is calculated individually and a weighted average is taken of them. The weights are proportional to the number of instances in each class. This ensures that classes with more instances have a greater influence on the overall precision calculation while still considering the performance of all classes.

Mathematically,

$$\text{Weighted Precision} = \frac{\sum_{i=1}^C (\text{Precision}_i \times \text{Instances}_i)}{\text{Total Instances}} \quad (4.4)$$

Where:

- C: number of classes
- Precision_i : precision for class i
- Instances_i : number of instances in class i
- Total Instances: total number of instances across all classes

4.4.3 Weighted Recall

Recall, also known as sensitivity or true positive rate, measures the proportion of true positive instances that are correctly identified by the classifier out of all actual positive instances.

Mathematically,

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \quad (4.5)$$

Where:

1. False Negatives(FN): instances incorrectly classified as negative by the classifier

Weighted recall is a variant of recall that accounts for class imbalance by calculating the recall for each class and then computing their weighted average. The weights are proportional to the number of instances in each class.

Mathematically,

$$\text{Weighted Recall} = \frac{\sum_{i=1}^C (\text{Recall}_i \times \text{Instances}_i)}{\text{Total Instances}} \quad (4.6)$$

Where:

- Recall_i : recall for class i

4.4.4 Weighted F1-Score

F1-Score is the harmonic mean of precision and recall.

Mathematically,

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4.7)$$

Weighted F1-Score is a variant of F1-Score that accounts for class imbalance by calculating F1-Score for each class individually and then computing their weighted average.

Mathematically,

$$\text{Weighted F1-Score} = \frac{\sum_{i=1}^C (\text{F1-Score}_i \times \text{Instances}_i)}{\text{Total Instances}} \quad (4.8)$$

Where:

- F1-Score_i : F1-Score for class i

4.5 Results on the ISIC 2016 Dataset

First pretrained torchvision models DenseNet201, ResNet101, VGG16, GoogleNet, WideResNet50_2, VGG19 with final layer replaced with a dense layer for classification. 3 models with highest test accuracy were ensembled using Sugeno Integral. Then a Self Attention Layer was added before the final dense layer of each model and retrained. Again 3 models with highest test accuracy were ensembled. To demonstrate Decentralised Federated Learning, 3 clients were created with the model which had the highest test accuracy. Each client has unique training images from the ISIC 2016 dataset. The final testing is done on same set to compare

results with non FDL results. The clients have been implemented using OOP. Each client is an object.

4.5.1 Selecting Models for Ensemble

We fine tuned pre-trained models DenseNet201, ResNet101, VGG16, GoogleNet, WideResNet50_2 and VGG19 with and without a self attention layer. The self attention layer was used on the final features extracted by the models before being sent to the dense layer.

Model	Loss	Accuracy	Precision	Recall	F1-Score
DenseNet201	0.5408	84.39%	82.90%	84.39%	82.87%
ResNet101	0.8386	83.86%	82.23%	83.86%	82.29%
VGG16	0.4682	82.80%	80.75%	82.80%	80.82%
GoogleNet	0.4732	85.19%	83.94%	85.18%	83.94%
WideResNet50_2	0.5278	83.86%	83.19%	83.86%	80.33%
VGG19	0.4978	83.86%	82.22%	83.86%	81.51%

Table 4.4: Individual Classifiers on the ISIC 2016 for selecting base models for Ensemble (bold models are selected for ensemble)

Model	Loss	Accuracy	Precision	Recall	F1-Score
DenseNet201	0.6924	84.13%	82.55%	84.13%	82.52%
ResNet101	0.7152	84.92%	83.58%	84.92%	83.32%
VGG16	0.402	80.16%	64.25%	80.16%	71.33%
GoogleNet	0.4671	86.24%	85.85%	86.24%	84.17%
WideResNet50_2	0.678	84.66%	83.23%	84.66%	83.01%
VGG19	0.4285	81.22%	80.01%	81.22%	74.57%

Table 4.5: Individual Classifiers on the ISIC 2016 for selecting base models with Self Attention for Ensemble (bold models are selected for ensemble)

We select the 3 models from Table 4.4 with highest test accuracy for ensemble. Since ResNet101, WideResNet50_2 and VGG19 have same accuracy we chose the model with lowest loss: VGG19. So, GoogleNet, DenseNet201 and VGG19 were ensembled.

From Table 4.5 GoogleNet, WideResNet50_2 and ResNet101 are chosen for ensemble.

From tables 4.4 and 4.5 we observe that GoogleNet model with Scaled Dot-Product Attention has the best results out of all the individual classifiers. Hence we use this classifier for decentralised federated learning.

4.5.2 Model Results

Model	Accuracy	F1-Score	Precision	Recall
Mohammed A. Al-masni et al. [16], 2020	Inception-ResNet-v2:81.79%	-	-	-
Jianpeng Zhang et al.[17],2019	86.28%	-	-	-
Himanshu K. Gajera et al.[18], 2022	86.28%	-	-	-
Demyanov et al.[19], 2016	88.00% for networks, 83.00% for globules	-	-	-
Guo et al.[20], 2018	85.22%	-	-	-
Zhang et al.[21], 2018	85.75%	-	66.44%	-
Ding et al.[22], 2021	-	-	71.90%	-
Yi et al.[23], 2018	81.20%	-	42.40%	-
He et al.[24], 2022	87.30%	-	69.70%	65.30%
Burdick et al.[25], 2017	69.30%	-	72.30%	76.00%
Proposed Model 1:	85.19%	87.42%	91.68%	85.19%
Proposed Model 2:	86.51%	88.77%	93.49%	86.51%
Proposed Model 3:	83.07%	81.19%	81.13%	83.07%

Table 4.6: Performance Comparison of Proposed Models with some past methods on the ISIC 2016 dataset

Table 4.6 shows that among our proposed models Ensemble of models with self attention performs the best with accuracy of 86.51% and F1-Score of 88.77%. The Federated Learning approach shows a decrease in performance as expected since each client has their own distinct data which cannot be aggregated.

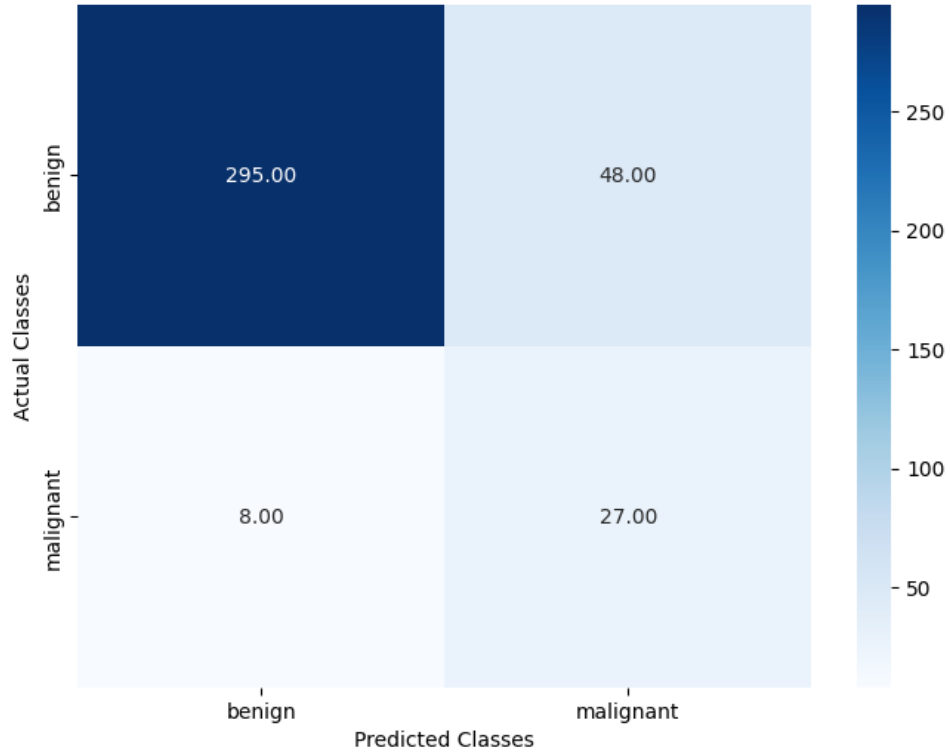


Figure 4.6: Confusion Matrix of 1st Model on the ISIC 2016 Dataset

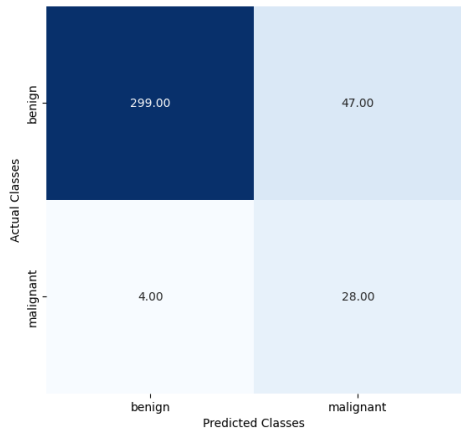


Figure 4.7: Confusion Matrix of 2nd Model on the ISIC 2016 Dataset

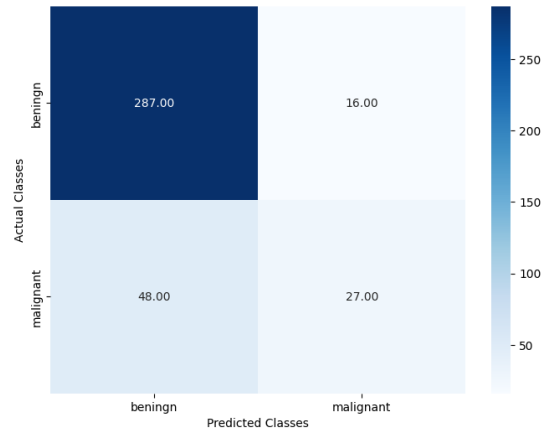


Figure 4.8: Confusion Matrix of 3rd Model on the ISIC 2016 Dataset

4.5.3 Results Analysis

Table 4.6 shows that proposed model 2 outperforms proposed model 1 in all metrics. It has 1.32% higher accuracy, 1.35% higher f1-score, 1.81% higher precision and 1.32% higher recall. This indicates that the attention mechanism has lead to a better performing model on the ISIC 2016 dataset.

The proposed model 3 is the least performing of the models due to the decentralised approach.

The confusion matrices Figure 4.6 and Figure 4.7 show that *malignant* class is more likely to be misclassified by the models.

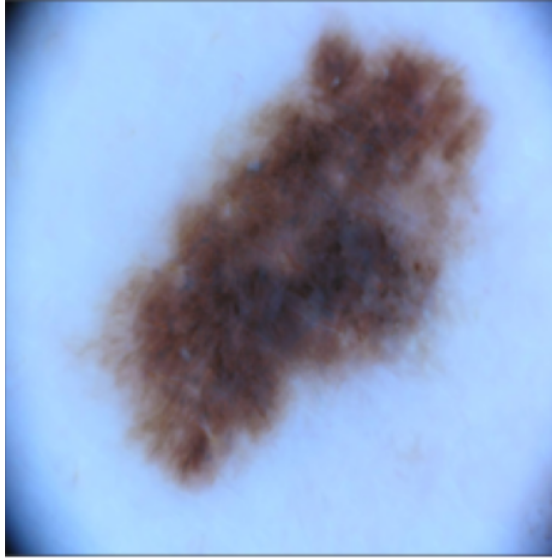


Figure 4.9: A benign skin lesion

As an example of the improved performance due to the ensemble, consider the image Figure 4.9. The image is correctly classified as *benign* by GoogelNet and DenseNet201 but misclassified as *malignant* by VGG19. The correct prediction of *benign* is made after ensemble of the prediction scores.

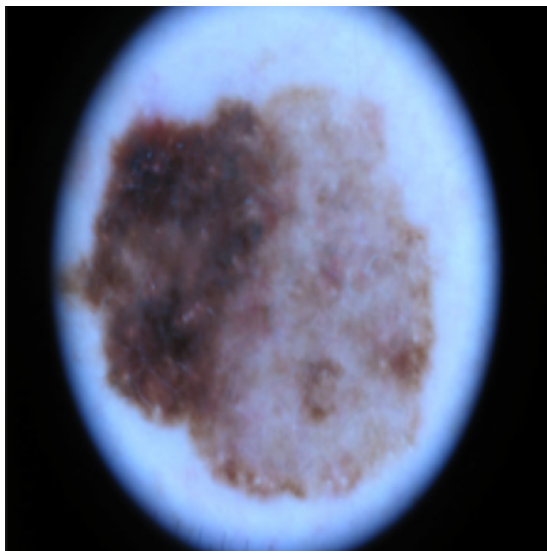


Figure 4.10: A malignant skin lesion

Figure 4.10 is a *malignant* skin lesion from the test dataset. It is misclassified as *benign* by the self attention applied googlenet model. The ensemble corrects this prediction to *malignant*.

4.6 Results on the HAM10000 Dataset

We first select base models with and without using attention mechanisms. Out of the models using attention mechanism, the best one is chosen for federated learning. Three models are selected for each ensemble and their performance measured by evaluation metrics and confusion matrices.

4.6.1 Selecting Models for Ensemble

We fine tuned pre-trained models DenseNet201, WideResNet50_2, SqueezeNet, GoogleNet and VGG11 with and without a self attention layer. The self attention layer was used on the final features extracted by the models before being sent to the dense layer.

Model	Loss	Accuracy	Precision	Recall	F1-Score
DenseNet201	0.3552	92.66%	92.26%	92.66%	92.33%
WideResNet50_2	0.3946	92.11%	91.74%	92.11%	91.79%
SqueezeNet	0.5441	90.30%	90.20%	90.30%	90.12%
GoogleNet	0.401	91.57%	91.45%	91.57%	91.36%
VGG11	0.3576	92.20%	92.09%	92.20%	92.07%

Table 4.7: Individual Classifiers on the HAM10000 for selecting base models for Ensemble (bold models are selected for ensemble)

Model	Loss	Accuracy	Precision	Recall	F1-Score
DenseNet201	0.3952	92.11%	91.83%	92.11%	91.75%
WideResNet50_2	0.4006	92.11%	91.69%	92.11%	91.82%
SqueezeNet	0.6755	90.66%	90.52%	90.66%	90.52%
GoogleNet	0.3792	91.30%	90.95%	91.30%	91.00%
VGG11	0.3046	92.57%	92.21%	92.57%	92.14%

Table 4.8: Individual Classifiers on the HAM10000 for selecting base models with Self Attention for Ensemble (bold models are selected for ensemble)

Based on Table 4.7, we choose DenseNet201, VGG11 and WideResNet50 for ensemble.

Based on Table 4.8, VGG11, WideResNet50_2 and DenseNet201 were chosen for ensemble.

We choose VGG11 with attention as the architecture for decentralised federated learning.

4.6.2 Model Results

Model	Accuracy	F1-Score	Precision	Recall
Ardan Adi Nugroho et al.[6], 2019	78.00%	-	-	-
Agyenta Charity Lariba et al.[7], 2022	86.91%	-	-	-
Muhammad et al.[8], 2019	89.80%	-	-	-
Emrah ÇEVİK et al.[9], 2019	85.62%	-	-	-
Rahman et al.[10], 2020	90.00%	83.00%	81.00%	-
Chaturvedi et al.[11], 2021	83.10%	83.00%	89.00%	83.00%
Khan et al.[12], 2024	87.02%	-	-	86.98%
Arshad et al.[13], 2021	91.70%	91.92%	92.00%	91.85%
Chaturvedi et al.[14], 2020	92.83%	84.00%	83.00%	84.00%
Sae-Lim et al.[15], 2019	83.93%	82.00%	-	85.00%
Proposed Model 1:	93.02%	93.26%	93.67%	93.02%
Proposed Model 2:	93.11%	93.47%	94.11%	93.11%
Proposed Model 3:	92.48%	92.08%	92.02%	92.48%

Table 4.9: Performance Comparison of Proposed Models with some past methods on the HAM10000 dataset

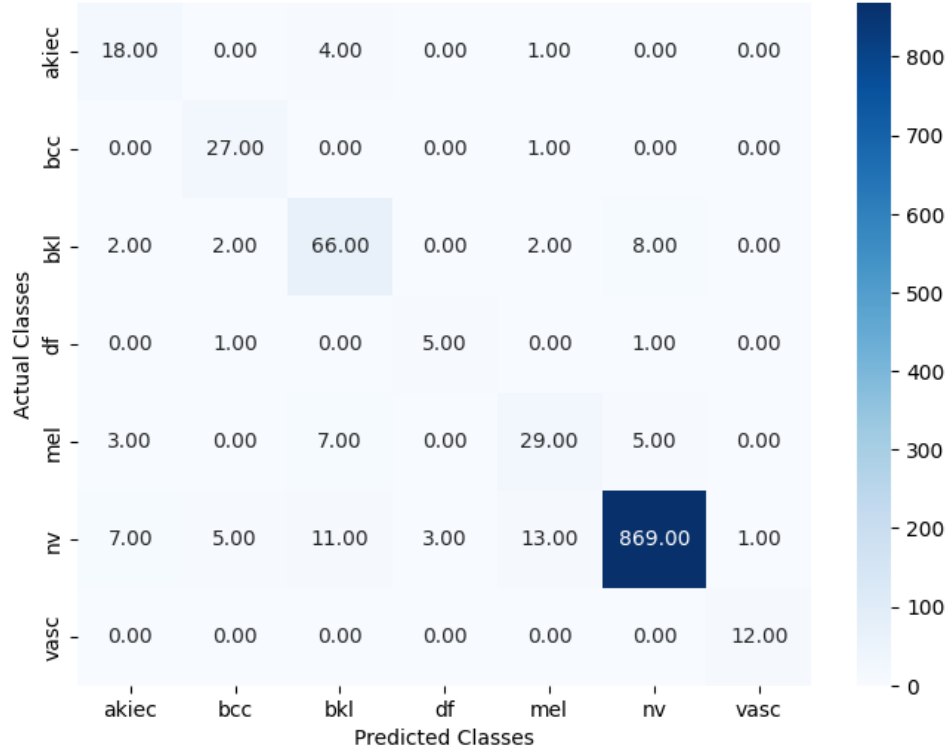


Figure 4.11: Confusion Matrix of 1st Model on the HAM10000 Dataset

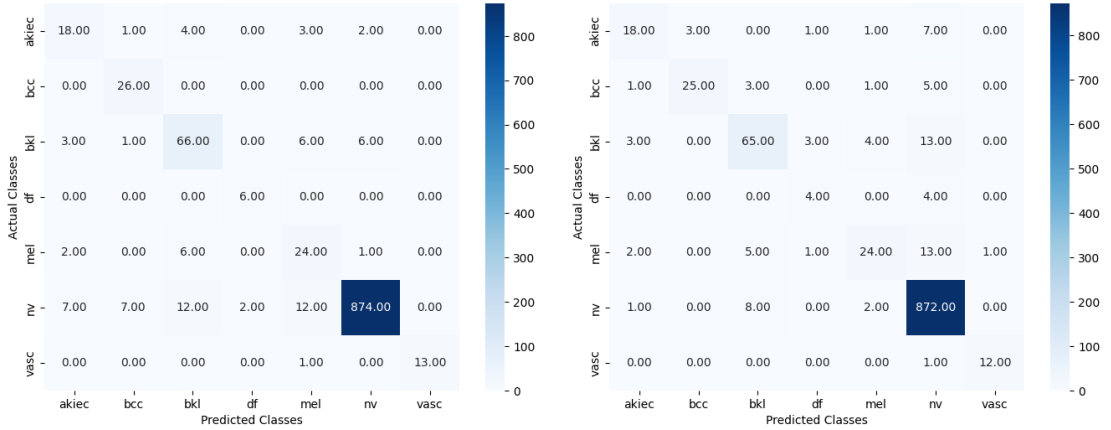


Figure 4.12: Confusion Matrix of 2nd Model on the HAM10000 Dataset

Figure 4.13: Confusion Matrix of 3rd Model on the HAM10000 Dataset

4.6.3 Results Analysis

Table 4.9 shows that proposed model 2 outperforms proposed model 1. It has 0.09% higher accuracy, 0.11% higher f1-score, 0.44% higher precision and 0.9% higher recall. These gains are attributed to the attention mechanism. While proposed model 3 has the least performance among the models, it is still close enough to the other 2 models to make decentralised learning a viable alternative and worth consideration.

The confusion matrices Figure 4.11, Figure 4.12 and Figure 4.13 show that every class is mostly properly classified with no class showing significant miss classification. The federated learning approach Figure 4.13 yields comparable results to centrally trained models Figure 4.11, Figure 4.12.

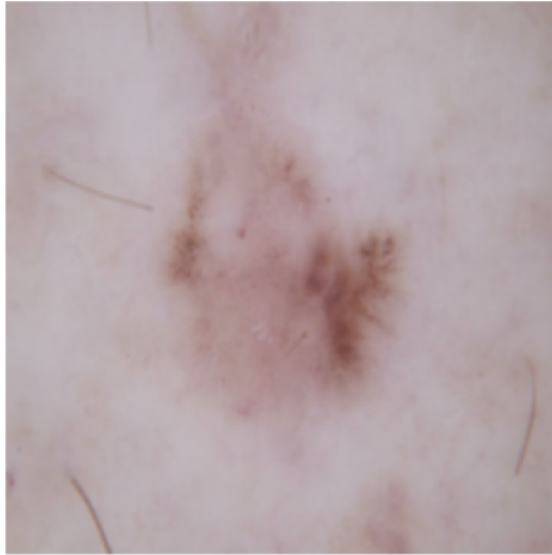


Figure 4.14: A basal cell carcinoma skin lesion

Figure 4.14 is a *basal cell carcinoma* skin lesion from the test set which is misclassified as *actinic keratoses* by the vgg11 model. The ensemble of models without attention outputs the correct prediction.

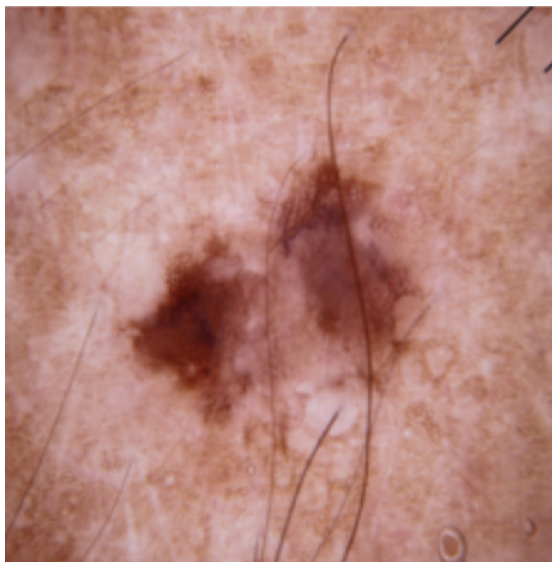


Figure 4.15: A melanoma skin lesion

Figure 4.15 is a melanoma skin lesion in the test dataset which is misclassified as *melanocytic nevi* by self attention applied wideresent50 but is correctly classified

by the ensemble

Conclusion and Future Scope

5.1 Conclusion

This project explored various techniques for improving model performance and privacy-preserving collaboration in distributed settings.

Fuzzy based ensemble of models using Sugeno Integral allowed us to leverage the diverse architectures of multiple models to make more accurate predictions.

Self-attention improved model performance by assigning higher attention weights to critical regions of the skin lesion image.

The decentralised approach to Federated Learning enabled the clients to train a shared model without aggregating model updates in a central server.

For the ISIC 2016 dataset, ensemble of base model predictions while yielding same accuracy as the highest accuracy base model googlenet, highly improved the precision to 91.68% and also improved recall and f1-score to 85.19% and 87.42% respectively. The highest metrics for the base models were: accuracy of 85.19%, precision of 83.94%, recall of 85.18% and f1-score of 83.94%. The ensemble of predictions of attention applied models gave improved results. Accuracy of 86.51%, f1-score of 88.77%, precision of 93.49% and recall of 86.51% were obtained. This is better than the ensemble of base model scores without attention. The highest metrics of the models before ensemble were: accuracy of 86.24%, f1-score of 84.17%, precision of 85.85% and recall of 86.24%. The decentralised federated learning employed googlenet with self attention mechanism as the model architecture for all clients. The metrics obtained were: 83.07% accuracy, 81.19% f1-score, 81.13% precision and 83.07% recall. Metrics obtained by centrally trained model were: 86.24% accuracy, 84.17% f1-score, 85.85% precision and 86.24% recall. It was the least performing of the proposed models but successfully demonstrated that a privacy focused approach can be implemented and made viable.

For the HAM10000 dataset, ensemble of base model predictions resulted in accu-

racy of 93.02%, f1-score of 93.26%, precision of 93.67% and recall of 93.02%. The highest base model metrics were accuracy of 92.66%, f1-score of 92.33%, precision of 92.26% and recall of 92.66%. The ensemble of self attention applied models resulted in 93.11% accuracy, 93.47% f1-score, 94.11% precision and 93.11% recall. This was the best performing of the three models. The base attention models had highest metrics of: 92.57% accuracy, 92.14% f1-score, 92.21% precision and 92.57% recall. For decentralised federated learning, VGG11 with self attention is used which originally had metrics: 92.57% accuracy, 92.14% f1-score, 92.21% precision and 92.57% recall. With decentralised federated learning approach, the obtained metrics were: 92.48% accuracy, 92.08% f1-score, 92.02% precision and 92.48% recall. These are close to the centrally trained model and demonstrate the viability of a privacy focused distributed learning approach.

Limitations: Some limitations of the project work are:

- Federated Learning and Ensemble techniques have not been integrated together.
- The federated learning approach used here trains each client only once.

5.2 Scope for Future Work

The project has several avenues for the improvements. Some are mentioned here.

- If the clients could train different model architectures one-by-one and retain each of the architectures, then they could be aggregated using Sugeno Integral. This would require all the clients to have the final trained weights for all the architectures since the ensemble is done on the predictions from each model. So each client will need all the models to get probability scores for ensemble.
- This project used continual approach to train the decentralised clients. An aggregate approach could also be attempted to increase impact of past knowledge.
- Other network topologies like fully connected, star, node clustering may be used though some of them would increase communication overhead. The line topology used can be extended to a ring by connecting first and last clients making multiple rounds possible.

- Attention layers can be applied to features extracted by the other convolutional layers in the CNN model.
- Attention mechanisms other than scaled dot-product attention can be used.

Bibliography

- [1] M. Dr. Liji Thomas, *Skin cancer stats*, <https://www.news-medical.net/health/Understanding-the-Diversity-of-Skin-Cancers.aspx#:~:text=SCCs%20are%20the%20most%20common,Asians%2C%20among%20whom%20BCC%20leads>, 2023.
- [2] M. Clinic, *Skin cancer*, <https://www.mayoclinic.org/diseases-conditions/skin-cancer/symptoms-causes/syc-20377605>, 2022.
- [3] M. Clinic, *Melanoma*, <https://www.mayoclinic.org/diseases-conditions/melanoma/symptoms-causes/syc-20374884>, 2023.
- [4] cancer.org, *Melanoma*, <https://www.cancer.org/cancer/types/melanoma-skin-cancer/about/key-statistics.html>, 2024.
- [5] N. C. Institue, *Metastasis definition*, <https://www.cancer.gov/publications/dictionaries/cancer-terms/def/metastasis>.
- [6] A. A. Nugroho, I. Slamet, and S. Sugiyanto, “Skins cancer identification system of ham10000 skin cancer dataset using convolutional neural network,” in *AIP conference proceedings*, AIP Publishing, vol. 2202, 2019.
- [7] C. Agyenta and M. Akanzawon, “Skin lesion classification based on convolutional neural network,” *Journal of Applied Science and Technology Trends*, vol. 3, no. 01, pp. 21–26, Apr. 2022. DOI: 10.38094/jastt301121.
- [8] M. A. Khan, M. Y. Javed, M. Sharif, T. Saba, and A. Rehman, “Multi-model deep neural network based features extraction and optimal selection approach for skin lesion classification,” in *2019 international conference on computer and information sciences (ICCIS)*, IEEE, 2019, pp. 1–7.
- [9] E. ÇEVİK and K. ZENGİN, “Classification of skin lesions in dermatoscopic images with deep convolution network,” *Avrupa Bilim ve Teknoloji Dergisi*, pp. 309–318, 2019.

-
- [10] Z. Rahman and A. M. Ami, “A transfer learning based approach for skin lesion classification from imbalanced data,” in *2020 11th International Conference on Electrical and Computer Engineering (ICECE)*, 2020, pp. 65–68. DOI: 10.1109/ICECE51571.2020.9393155.
 - [11] S. S. Chaturvedi, K. Gupta, and P. S. Prasad, “Skin lesion analyser: An efficient seven-way multi-class skin cancer classification using mobilenet,” in *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020*, Springer, 2021, pp. 165–176.
 - [12] M. A. Khan, K. Muhammad, M. Sharif, T. Akram, and S. Kadry, “Intelligent fusion-assisted skin lesion localization and classification for smart healthcare,” *Neural Computing and Applications*, vol. 36, no. 1, pp. 37–52, 2024.
 - [13] M. Arshad, M. A. Khan, U. Tariq, *et al.*, “A computer-aided diagnosis system using deep learning for multiclass skin lesion classification,” *Computational intelligence and neuroscience*, vol. 2021, 2021.
 - [14] S. S. Chaturvedi, J. V. Tembhurne, and T. Diwan, “A multi-class skin cancer classification using deep convolutional neural networks,” *Multimedia Tools and Applications*, vol. 79, no. 39, pp. 28 477–28 498, 2020.
 - [15] W. Sae-Lim, W. Wettayaprasit, and P. Aiyarak, “Convolutional neural networks using mobilenet for skin lesion classification,” in *2019 16th international joint conference on computer science and software engineering (JC-SSE)*, IEEE, 2019, pp. 242–247.
 - [16] M. A. Al-Masni, D.-H. Kim, and T.-S. Kim, “Multiple skin lesions diagnostics via integrated deep convolutional networks for segmentation and classification,” *Computer methods and programs in biomedicine*, vol. 190, p. 105 351, 2020.
 - [17] J. Zhang, Y. Xie, Q. Wu, and Y. Xia, “Medical image classification using synergic deep learning,” *Medical image analysis*, vol. 54, pp. 10–19, 2019.
 - [18] H. K. Gajera, M. A. Zaveri, and D. R. Nayak, “Patch-based local deep feature extraction for automated skin cancer classification,” *International Journal of Imaging Systems and Technology*, vol. 32, no. 5, pp. 1774–1788, 2022.
 - [19] S. Demyanov, R. Chakravorty, M. Abedini, A. Halpern, and R. Garnavi, “Classification of dermoscopy patterns using deep convolutional neural networks,” in *2016 IEEE 13th International symposium on biomedical imaging (ISBI)*, IEEE, 2016, pp. 364–368.

-
- [20] Y. Guo, A. S. Ashour, L. Si, and D. P. Mandalaywala, “Multiple convolutional neural network for skin dermoscopic image classification,” in *2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, IEEE, 2018, pp. 365–369.
 - [21] J. Zhang, Y. Xie, Q. Wu, and Y. Xia, “Skin lesion classification in dermoscopy images using synergic deep learning,” in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16–20, 2018, Proceedings, Part II 11*, Springer, 2018, pp. 12–20.
 - [22] S. Ding, Z. Wu, Y. Zheng, *et al.*, “Deep attention branch networks for skin lesion classification,” *Computer methods and programs in biomedicine*, vol. 212, p. 106 447, 2021.
 - [23] X. Yi, E. Walia, and P. Babyn, “Unsupervised and semi-supervised learning with categorical generative adversarial networks assisted by wasserstein distance for dermoscopy image classification,” *arXiv preprint arXiv:1804.03700*, 2018.
 - [24] X. He, Y. Wang, S. Zhao, and C. Yao, “Deep metric attention learning for skin lesion classification in dermoscopy images,” *Complex & Intelligent Systems*, vol. 8, no. 2, pp. 1487–1504, 2022.
 - [25] J. Burdick, O. Marques, J. Weinthal, and B. Furht, “Rethinking skin lesion segmentation in a convolutional classifier,” *Journal of digital imaging*, vol. 31, pp. 435–440, 2018.
 - [26] R. Kundu, P. K. Singh, S. Mirjalili, and R. Sarkar, “Covid-19 detection from lung ct-scans using a fuzzy integral-based cnn ensemble,” *Computers in Biology and Medicine*, vol. 138, p. 104 895, 2021.
 - [27] H. Tahani and J. Keller, “Information fusion in computer vision using the fuzzy integral,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 3, pp. 733–741, 1990. DOI: 10.1109/21.57289.
 - [28] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, *Attention is all you need*, 2023.
 - [29] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, *Communication-efficient learning of deep networks from decentralized data*, 2023.
 - [30] L. Yuan, L. Sun, P. S. Yu, and Z. Wang, “Decentralized federated learning: A survey and perspective,” *arXiv preprint arXiv:2306.01603*, 2023.