Mitigating Domain Shift in Skin Cancer Recognition with Squeeze-Excitation Attention and Dropout-Consistent Federated Averaging

by

Hasnat Rafi Uddin
24341277
Abu Hossain Mohammad Abed Hasan Tias
21301295
Raian Islam
22101777

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

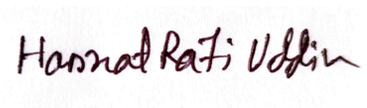
Department of Computer Science and Engineering
Brac University
January 2026

Declaration

It is hereby declared that

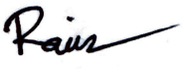
1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:



Hasnat Rafi Uddin

24341277



Raian Islam

22101777

Abu Hossain Mohammad Abed Hasan Tias

21301295

Approval

The thesis/project titled "Mitigating Domain Shift in Skin Cancer Recognition with Squeeze-Excitation Attention and Dropout-Consistent Federated Averaging" submitted by

1. Hasnat Rafi Uddin (24341277)
2. Raian Islam (22101777)
3. Abu Hossain Mohammad Abed Hasan Tias (21301295)

of Spring, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science in 2026 Year.

Examining Committee:

Supervisor:
(Member)

Md. Ahasanul Alam

Lecturer
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Dr Md Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi

Associate Professor & Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

Skin lesion classifiers can achieve high in-domain accuracy yet fail in clinical deployment, where hospitals differ in imaging devices, protocols, and class distributions and patient data cannot be centralized. This thesis first reproduces three state of the art baselines (RegNetY-32GF, VGG16, and SkinLesNet) on three widely used datasets: HAM10000, ISIC (9-class), and PAD-UFES-20. We then propose DermaNet, an EfficientNet-B3 based model augmented with squeeze–excitation attention and trained with focal loss, realistic augmentation, and structured dropout. In centralized training, DermaNet achieves 87.83% on HAM10000, while performance on ISIC and PAD-UFES-20 remains substantially lower due to domain shift. To quantify real world generalization, we evaluate cross-dataset transfer and observe catastrophic collapse. For example, DermaNet trained on HAM10000 drops to approximately 40% on ISIC and 20% on PAD, while the ISIC-trained DermaNet falls to approximately 14% on HAM. We address privacy and heterogeneity via federated learning across three clients and benchmark all frameworks under a FedAvg baseline (FedSE), which improves robustness relative to cross domain centralized deployment (84.54% / 69.55% / 65.29% on HAM / ISIC / PAD). However, FedAvg with stochastic dropout suffers from zero dilution. Structurally dropped weights are averaged as true zeros, weakening attention and slowing convergence. Finally, we propose FedMD, a mask-aware aggregation strategy that averages only active (non-dropped) parameters using deterministic client masks, yielding consistent gains over standard FedAvg, On HAM it achieved 85% accuracy, on ISIC it achieved 71% accuracy and in case of PAD-UFES it achieved 75% accuracy.

Keywords: Federated Learning, Skin Cancer Detection, Domain Shift, Squeeze-Excitation Networks, Privacy-Preserving, Dropout Regularization, Dermoscopy, Heterogeneous Data, Medical Image Analysis.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

α	Focal loss weighting factor
η	Learning rate
γ	Focal loss focusing parameter
$\mathbf{w}_k$	Local model parameters of client k
$\mathbf{w}_{\text{global}}$	Global model parameters
K	Number of clients in the federation
k	Client index, $k \in \{1, \ldots, K\}$
M_k	Binary dropout mask of client k
$M_k[i]$	Binary mask value (0/1) for client k at parameter index i
N	Total number of samples across all clients ($N = \sum_k n_k$)
n_k	Number of samples at client k
p	Dropout probability
r	Reduction ratio in Squeeze-Excitation block
w	Model weight parameters
Acc	Accuracy
AdamW	Adam optimizer with decoupled weight decay
AI	Artificial Intelligence
CNN	Convolutional Neural Network
DNN	Deep Neural Network
Domain shift	Distribution mismatch between training and deployment data
Dropout	Stochastic regularization via random neuron deactivation
F1-score	Harmonic mean of precision and recall

FedAgg Adaptive Federated Learning with Aggregated Gradients

FedAvg Federated Averaging

FedMD Mask-aware federated aggregation method (proposed)

FedNova Normalized Federated Averaging

FedProx Federated Proximal

FedRolex Rolling Sub-Model Extraction for model-heterogeneous FL

FedSE Baseline federated system using standard FedAvg with stochastic dropout

FL Federated Learning

FLuID Federated Learning using Invariant Dropout

FPR False Positive Rate

GDPR General Data Protection Regulation

HAM10000 Human Against Machine with 10000 training images dataset

HIPAA Health Insurance Portability and Accountability Act

IID Independent and Identically Distributed

ImageNet Large-scale visual recognition dataset

ISIC International Skin Imaging Collaboration dataset

kNN k-Nearest Neighbors

MBConv Mobile Inverted Bottleneck Convolution

Non-IID Non-Independent and Non-Identically Distributed

PAD-UFES-20 Smartphone-based clinical skin lesion dataset

PFL Personalized Federated Learning

ReLU Rectified Linear Unit

RGB Red–Green–Blue color space

RMSE Root Mean Squared Error

SE Squeeze-and-Excitation

SGD Stochastic Gradient Descent

Softmax Softmax activation function

TPR True Positive Rate

Zero dilution Aggregation error where dropout-induced zeros weaken learned weights

Chapter 1

Introduction

Early detection of skin cancer is the biggest challenge in today's dermatology, as an early diagnosis can significantly benefit the patient. Deep learning has shown great potential in automated classification of skin lesions, where convolutional neural networks have achieved dermatologist level accuracy on benchmark datasets[1]. There is a gap between success in controlled experiments and reality in clinical application. Models that reach 85-90% accuracy on their training dataset drop to 10-40% accuracy when deployed at other institutions. This is due to differences in imaging devices, acquisition protocols, lighting conditions, and patient demographics. The term domain shift refers to a fundamental problem that hinders the uptake of AI assisted diagnosis in dermatology[2].

The domain shift issue is prominent across typical dermoscopy datasets. A model trained on high quality dermoscopic images created with standardized clinical equipment could perform excellently on similar data; however, when applied to images from ISIC's[3] multi institutional collection or on smartphone photographs from PAD-UFES-20[4], performance is poor. The variation exists in several dimensions. For instance, while HAM10000[5] consists of controlled dermoscopy, which means consistent magnification and lighting, ISIC is a mix of images acquired by different institutions with varying protocols. PAD-UFES-20 consists of smartphone captures with varying resolution and ambient lighting. Class distributions differ from one dataset to another. The rate of melanoma ranges from 5% to 30% from different sources.

This failure across datasets is a clinical crisis, not just an engineering issue. A skin cancer detection system that has been trained at a well resourced dermatology center in Europe cannot be deployed at a rural clinic in South America without total retraining on local data. Despite very complex architectural tricks like attention mechanisms, transfer learning and advanced regularization, the distributional mismatch is not resolved[6, 7]. Our experiments show that we can achieve an accuracy of 87.83% on HAM10000[5], using DermaNet, which is EfficientNet-B3[8] architecture with squeeze-excitation attention blocks. However, using DermaNet on ISIC leads to a drop to 40% accuracy. Similarly, the same model achieves 20% accuracy on PAD-UFES-20. Results across all studied architectures where it influences, for instance, with RegNetY-32GF[9], VGG16 and SkinLesNet[10], we see drops 30-70 percentage points. The mechanisms intended for focusing on diagnostically relevant features learn to fit, rather than clinical transferability, dataset specific patterns.

We cannot usually adopt the standard solution of centralizing data from various insti-

tutions into a unifying training set in medical contexts. Laws like HIPAA & GDPR restrict sharing of medical data without identifiers[11]. Several factors including ethical obligations, competition pressures and liability concerns hinder consolidation of data by institutions. Federated learning is the essential framework that tackles the dual challenge of domain shift and privacy protection[12]. Federated learning enables collaborative model training between decentralized hospitals while preventing raw data transfers[12]. This allows hospitals to gain benefits from each other while keeping their data locally governed. When training a model, each hospital conducts training on its own patient population. Only the parameters of the model are shared through a server that aggregates globally[12].

The first version of our federated implementation, FedSE, shows that collaborative training on HAM10000[5], ISIC[3] and PAD-UFES-20[4] clients achieves significant improvements over cross-dataset centralized deployment. The performance of the federated model is 84.54% on HAM, 69.55% on ISIC, and 65.29% on PAD, an improvement over their centralized cross-dataset test baselines. This validates the core value proposition of federated learning leveraging the diversity of multiple institutions results in models with generalization capabilities far beyond that allowed by the data of any single institution.

Nevertheless, the accuracy of FedSE suffers by as much as 3-5% when compared to centralized DermaNet training on each dataset individually. Overall, the Centralized HAM score dropped from 87.83% to 84.54%. ISIC score increased from 67.39% to 69.55% while PAD score decreased from 70.50% to 65.29%.

The performance gap arises from the fundamental incompatibility of stochastic regularization and standard federated aggregation. Dropout is a very popular method in deep learning that prevents overfitting by dropping some neurons randomly[13]. In federated learning, dropout is applied independently by different clients with different random masks, so during each training round different clients have different active neurons[12]. When the FedAvg algorithm aggregates these updates by taking the average of the weights sent from each of the clients, it considers these structurally dropped weights (which are zero due to being regularized rather than learned) to be legit learned values.

To overcome this basic limitation, we present FedMD, a mask-aware aggregation mechanism that takes the structure of zeros induced by dropout into account. FedMD sets all weights of all rounds as the corresponding Client weight that can be sent to the Server by the Client. In the aggregation process, the server averages weights from the clients where the neuron was active in question i.e. it does not compute with structural zeros. It preserves the learned representations on active neurons while also retaining dropout regularization within each client's local training.

As our result shows, FedMD removes zero dilution and restores the standard FedAvg performance loss. Through the three client federation of HAM10000[5], ISIC[3] and PAD-UFES-20[4], FedMD achieves final test accuracies of 85.59% on HAM, 71.92% on ISIC and 75.29% on PAD. This corresponds to gains of 1.05, 2.37, and 10.00 percentage points versus the FedSE baseline. The F1-scores also approach 0.8626, 0.7301 and 0.6852 with PAD showing the most dramatic recovery from zero dilution impact. The squeeze-excitation channel weights regain their original dynamic range, with mean parameter coverage stabilizing at 2.10 clients, and only 2.76% of SE parameters experience zero coverage per client in a given global round. With detection of the minority class improved significantly, the melanoma recall on the

HAM increased to 81% and that of actinic keratosis on the ISIC achieved 92% recall. FedMD allows attention-based architectures to work correctly in federated settings, thereby unleashing the power of collaborative training across multiple institutions for robust and privacy preserving skin cancer detection.

1.1 Motivation

Cancer is one of the leading causes of death globally, accounting for almost 10 million deaths in 2022 alone[14]. Other chronic conditions, such as cardiovascular and neurological diseases, also require diagnostics stemming from accurate and data informed methods. Artificial intelligence (AI) as a progressive approach relies on deep learning, providing advanced capabilities from tumor detection through medical imagery to disease development predictions across clinical trajectories. And one of the most common cancers in the world is skin cancer, where early diagnosis is essential. Deep learning based image classification, an application of artificial intelligence, has shown remarkable promise for the automated diagnosis of skin lesions. In controlled research settings, this technology has matched or surpassed dermatologist level accuracy. Because diagnosis in dermatology relies on visual recognition, pattern recognition of dermoscopy images can directly inform a clinical decision. As we already know, there is a basic disconnect between experimental success and deployment.

The failure of this generalization occurs due to the domain shift, where data distributions vary greatly across institutions. Federated Learning (FL) offers a way around this complication, providing decentralized opportunities for model training across hospitals and research centers without ever transferring patient data[12]. In our first federated implementation results show a gain of accuracy percentage points on ISIC and PAD respectively when compared to cross domain centralized baselines. This confirms that federated learning draws on multi institutional diversity to produce models that generalize further than was possible with a single institution dataset.

However, standard federated learning causes a performance drop. Centralized training on individual datasets outperforms our federated models. The gap arises from zero dilution, which is a previously unnoticed incompatibility between stochastic dropout regularization and federated averaging. Dropout randomly deactivates neurons during training, but on different clients in federated settings drop different neurons independently[13]. When FedAvg aggregates updates by averaging weights across clients, structural zeroes from dropout are treated as learned values that indicate feature unimportance.

Algorithms currently in use for federated learning, such as FedAvg[12], FedProx[15], and SCAFFOLD[16], do not tackle this aggregation level corruption of dropout regularized architectures. As such, they tend to ignore communication efficiency, client drift, and non-IID data distributions. The field is becoming saturated due to a growing interdependence between the two and the growing use of an increased number of attention based models in the medical imaging field. Our work tackles the issue of zero dilution through mask aware aggregation, and employs structural properties of dropout induced zeros, yielding a strong, privacy preserving generalizable skin cancer detection, delivering on the promises of federated learning for medical AI.

1.2 Research Objective

The main goal of this study is to formulate a federated learning framework to ensure robust and privacy assured skin cancer detection from different medical institutions while also showing the fundamental incompatibility between dropout regularization and normal federated averaging.
The first goal is to create comprehensive baselines by reproducing and evaluating state of the art skin cancer detection architectures over heterogeneous datasets. Implement RegNetY-32GF[9], VGG16 and SkinLesNet[10] on HAM10000[5], ISIC[3] and PAD-UFES-20[4] datasets to measure their in domain performance and cross dataset generalization failures. This research aims to enhance the baseline approaches by developing a deep learning architecture called DermaNet, which will feature an EfficientNet-B3[8] backbone with squeeze-excitation attention, focal loss to handle class imbalance[17], and structured dropout regularization for dermoscopic image classification. We aim to outperform the existing baseline models while ensuring that the architectural components are fit for federated deployment.
The second objective is to address the issue of cross domain generalization in skin cancer recognition. This involves systematic cross dataset evaluation, where models trained only on one institution's dataset are tested on other institution's datasets without further fine tuning. This quantifies the catastrophic performance drop attributed to domain shift between different imaging devices, acquisition protocols and patient populations. By knowing them one can see the basic motivation behind collaborative multi institution learning methods that can utilize diverse data distributions without centralizing sensitive patient data.
The third objective implements and evaluates federated learning as a privacy-preserving mechanism to facilitate multi institutional collaboration. To deploy federated learning, we will design a federated architecture. HAM10000[5], ISIC[3] and PAD-UFES-20[4] will act as independent clients representing hospitals with heterogeneous dermoscopy datasets. In this architecture, we will implement the standard Federated Averaging algorithm where only model parameters are shared, while raw patient data remain local. We will measure the effectiveness of federated learning in comparison to both cross domain centralized deployment and centralized training where we train the model on individual datasets. The objective of this goal is to investigate whether federated learning can effectively exploit institutional diversity to enhance generalization while benefiting from privacy guarantees.
The fourth objective is to study the limitations of standard federated learning under domain shift and dropout regularization. To this end, we implement FedSE, a baseline federated system deploying DermaNet across heterogeneous clients with standard FedAvg, and show systematic performance degradation despite access to diverse multi institutional data. The goal of the research is to understand the federated models' poor performance which is a few percentage points worse than centralized training, culminating in the discovery and characterization of the zero dilution problem. Due to stochastic dropout creating inconsistent network structures across clients, structural zeros from regularizing the corresponding neural nets get mistakenly considered as expression of importance for features from the aggregation by the server[13]. We aim to measure the harm caused by this integration error to attention mechanisms, specifically squeeze-excitation channel weights that were meant to laser focus on the entire range but instead are collapsing to a narrow band

which makes them lose their diagnostic selectivity[7].
The fifth objective is to create FedMD, which is a dropout compatible federated aggregation strategy which achieves zero dilution while retaining regularization benefits and privacy guarantees. The process involves creating a deterministic mechanism for generating masks. As a result, each client gets a distinct but fixed dropout mask. It must obtain this mask through the client identifiers and seed randomization. The server then implements mask-aware aggregation. In this method, only actively trained parameters are averaged. The normalisation here happens with respect to the mask weighted client proportions. This is done instead of the total client counts. Additionally, we apply a mask-aware approach for the squeeze-excitation attention layers. This is while using a standard FedAvg for the backbone network. We aim to design a solution that respects the structural nature of zeros created by dropout, is compatible with existing federated learning systems, and incurs minimal computational or communication overhead.
We will evaluate FedMD on various dimension and the final objective. The metrics to assess the performance of our masked-aware client aggregation strategy will include the following: First, the convergence pace and training stability over global rounds, to compare whether mask-aware aggregation speeds up federated learning as contrasted to standard FedAvg. Second, the final test accuracy and F1-scores over all three heterogeneous clients, to quantify the performance penalty caused by eliminating zero dilution. Third, the behavior of the squeeze-excitation attention mechanism, to check if the channel weights recover their full dynamic range and diagnostic selectivity. Fourth, the federated performance against centralized baselines and FedSE, to establish if mask-aware aggregation achieves performance parity with centralized training. Finally, the improvements in minority class detection, specifically for clinically important melanomas where attention mechanisms must correctly emphasize rare diagnostic features. Through these comprehensive evaluations, we aim to show that proper treatment of dropout induced structural zeros allows attention based architectures to work properly in federated settings, and thus use the full potential of privacy preserving multi institutional skin cancer detection systems.

1.2.1 Research Contribution

The contributions of this thesis on privacy preserving skin cancer detection and federated learning in medical imaging are :

1. Comprehensive Analysis Set up and Assessment. We systematically reproduce and evaluate three state of the art architectures. RegNetY-32GF[9], VGG16, and SkinLesNet[10] across three heterogeneous dermoscopy datasets (HAM10000[5], ISIC 9-Class[3] and PAD-UFES-20[4]). By training on one dataset and testing on other datasets without fine tuning, we assess the catastrophic generalization failures of centralized training through rigorous cross dataset evaluation. As established from our experiments, the performance of the model drops by 30 to 70 percentage points on different dataset pairs. This is the first motivation for multi-institutional collaborative learning.

2. DermaNet is an architecture for dermoscopic image. We present DermaNet, a hybrid architecture that incorporates an EfficientNet-B3 backbone with

Squeeze-Excitation attention mechanisms and a structured dropout regularisation (0.4→0.3→0.2) combined with a focal loss for handling class imbalance DermaNet's performance on HAM10000 yields an impressive accuracy of 87.83% which is 1.35% more than RegNetY-32GF. More notably, DermaNet recovers from extreme baseline failures on difficult datasets, achieving a 26.71% improvement in ISIC performance over VGG16 (40.68%→67.39%) and a 13.02% improvement in PAD-UFES-20 performance over SkinLesNet (57.48%→70.50%). As shown by the results, attention based architectures with clinical realistic training protocols are useful for dermatological image classification, especially in cases characterized by heavy overfitting or limited training data.

3. A unified learning approach towards heterogeneous medical imaging We generalise DermaNet to the privacy preserving federated learning setting, adopting FedSE as a baseline system using standard FedAvg aggregation across three heterogeneous clients mimicking diverse hospital environments. Our federated implementation shows that collaborating training across institutes perform significantly better than a cross domain centralized deployment with accuracies of 84.54%, 69.55%, and 65.29% on HAM, ISIC, and PAD respectively which is a dramatic improvement over 10–40% accuracies when the centralized model faces out of domain data. Federated learning is not just a privacy friendly alternative, but instead might be essential for creating clinically deployable skin cancer detection systems that generalize across imaging protocols and patients.

4. We identify and formally characterize a failure mode in federated learning which is zero-dilution. This occurs when the standard FedAvg aggregation naively averages structural zeros, which come from stochastic dropout, with true learned weights. We show that the aggregation error is responsible for the degradation of the attention mechanism through systematic analyses of this phenomenon. When we apply Squeeze-Excitation on the ResNet architecture, we note the channel weights collapse from their expected dynamic range of 0.1 to 0.9 to a small band of 0.4 to 0.6. As a result, the selectivity of the attention mechanism is rendered ineffective or neutralized. According to our analysis, the presence of zero dilution compounds across layers and training rounds leads to a systematic degradation in performance with respect to centralized training. According to the parameter coverage analysis, applying 30% dropout on three clients results in only 2.76% of the SE attention parameters receiving zero training in a given global round. However, the impact of averaging their structural zeros with active weights throughout the network is harmful. This contribution reveals a basic incompatibility between popular stochastic regularization methods and standard federated aggregation algorithms, which has serious consequences for attention based medical imaging models.

5. We propose FedMD , a new aggregation method that eliminates zero dilution by generating deterministic mask and mask aware averaging. Every client is assigned a specific dropout mask initialized using the same seed. This allows the topology to remain fixed and consistent for each client during federated training, while also providing a regularization benefit. During the aggregating, the server computes an element wise normalized average over the various weight

vectors w_k supplied by the clients, where for every k only those weights w_k are considered for which the parameter was actually trained by the client.
This selective aggregation recovers diluted weights. FedMD consistently surpasses the FedSE baseline, showing improvements of 1.05% on HAM (from 84.54% to 85.59%), 2.37% on ISIC (from 69.55% to 71.92%), and 9.90% on PAD (from 65.39% to 75.29%), especially indicating a recovery for the domain shifted PAD client. Importantly, FedMD involves zero communication overhead masked being generated locally through deterministic seeds. It also has a low computational cost only element wise operations are needed and it is an architectural method which works with any attention based model, thus it can be easily adopted into existing federated learning frameworks beyond skin cancer detection.

This work shows that treatment of structural zeros incurred by regularization enables attention architectures to work correctly in a distributed setting. A proper handling of structural zeros together with attention architectures unlocks the full potential of privacy preserving federated medical imaging and collaboration for robust clinical AI.

Chapter 2

Background Studies and Literature Reviews

2.1 Preliminaries

Federated Learning (FL) is a machine learning paradigm that is designed to train the global model distributedly on multiple clients. And the clients do not share anything between them. The central server communicates with them and shares the updated global model after aggregation. This approach also reduces communication overhead by transmitting model parameters and updated the local model after every round instead of raw data[18].
Here the clients perform local training on their own data and periodically communicate with the central server which then aggregates the received local model updates. Then sends the updated global model back to the clients. This process continues iteratively until convergence.
However there are several problems this approach faces. They are, Non-IID data distribution: every client possesses a different dataset and their dataset may be biased[19]. Thus this results in biased local updates, dropout rate increase and slower convergence. System Heterogeneity: not every clients device can perform the best. They can also face various issues such as network issues, limited bandwidth, availability issues. Thus this results in unbalanced contributions[18]. Client Dropout: due to the similar issues many client may dropout during training rounds, also increasing the bias and slower convergence[12, 15, 20]. Many times certain neurons may become inactive, it affects the models effectiveness. Also some recent research has shown that clients updating in FL can reveal certain info that can be used to model inverse attack and get substantial info about local dataset[21, 22].
Hence, to overcome these obstacles there are many other approaches that reach higher peaks with FL and we will discuss them in later sections.

2.2 Background Studies

Compared to base machine learning we have advanced a lot. There are many approaches to FL that solve most problems and give excellent results. Here we will discuss some state-of-the-art approaches of FL that mitigates the above mentioned issues and achieve excellent results. Also they will provide a comparative foundation

for our approach, we will see the evaluation analysis in a later chapter.

2.2.1 Federated Learning

Federated Learnings' privacy protection feature makes it highly suitable for not only small cases such as massage text suggestion on smart device keyboards but also in sensitive situations such as healthcare, personal finance, IT intelligence, etc. Our approach hopes to achieve more secure and accurate results. The above mentioned issues of FL can slow convergence and overall performance. So the FedAvg method is the baseline performance for later models[12]. Similar approaches that work on the averaging steps are FedProx[15], FedNova[20] and FedAgg[23]; they focus on getting better results on aggregation steps to get less error. We will also discuss other approaches like FLuID[24] and DReS-FL[25] that emphasize dropout resilient, improving training robustness in varying participating clients scenario.

2.2.1.1 FedAvg

FedAvg was proposed by McMahan et al. [12], they intended for this approach to provide a guideline and performance benchmark for later advancements. It also intended to show the problems of decentralized data from devices. It also combines local stochastic gradient descent (SGD) on each client with centralized model averaging on the server.
Here the authors emphasizes on handling the non-IID data and unbalanced data distribution across clients, along with addressing some communication constraints. They had to use a controlled environment that would be suitable for experiments, because a practical federated optimization system must tackle real-world scenarios, such as dynamic client datasets, variable availability and unreliable participation. As these situations were then beyond the scope of work, the authors still tried to address the non-IID and unbalanced data in this situation.
Similar researches has also done iterative averaging on central server, from the data that has been trained distributively[26], though they do not consider non-IID property of data. Hence, they select fraction of clients in each round and perform local training and update the central server. It continues till convergence.
The proposed Federated Avareging algorithm that is applicable to any finite-sum objective of the form is:

$$\min_{w \in R^d} f(w) = \frac{1}{n} \sum_{i=1}^{n} f_i(w),$$

which, under federated partitioning, is reformulated as:

$$f(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w), \quad \text{where} \quad F_k(w) = \frac{1}{n_k} \sum_{i \in P_k} f_i(w),$$

with P_k denoting the local dataset of client k, and n_k its size.
We know that in data center optimization communication costs are relatively small and computation cost high, in comparison FL communication cost is important, as this determines and constrains the clients communication with the central server[12]. Thus, to achieve better results they focus on reducing communication round by increasing local computation.

The approach the authors took to model this approach are, applied SGD by using randomly selected clients per round. It costs large number of communication round, while being communicatively efficient. Also they increased parallelism, where more clients work independently between each communication round and increase computation on each client. This will result in reducing the communication round helping balance the load.
As mentioned earlier this paper was aimed to provide straightforward guidelines to where further research is needed. Because we can see that this approach has many lacking and limitations, which will be discussed in later subsections, along with its solutions and many new approaches from it. Despite its lacking the authors has provided valuable insights of FL and its limitations.

2.2.1.2 FedProx

FedAvg has been working well to train models while keeping the data locally and privately. However it has one big drawbacks which is it cannot perform well in heterogeneous situations. While fedAvg worked well in case of powerful devices, it failed to impress in case of less powerful devices and specially in case of non-independent and non-identically distributed data[27, 28]. The paper titled "Federated Optimization in Heterogeneous Networks" introduces us with an advanced framework to overcome the challenges of federated learning. This framework is called the FedProx. This works well specially in case of heterogeneous situations. This paper developed FedProx as a general convergence analysis framework. Authors performed experiments on both synthetic and real world datasets to evaluate the results[29].
Federated learning runs the models locally which results in sometimes the model updates gets carried far away from the main global model due to heterogeneous situations. FedProx fixes this issue by adding proximal term which acts like a regularizer. This proximal term restricts models so that they can't get too far away from the global model[29]. Different devices has different capabilities and the datasets on those devices also varies from IID to Non IID. As a result the models may get carried away in some circumstances. Proximal term plays its role here by keeping theme in one range. This provides stability and robustness in training in times of unpredictability. FedProx also accepts partial outputs from local models. If a model cant complete training fully due to various reasons, it still will be counted as FedProx accepts partial work contribution. FedProx also introduces another term called B-local dissimilarities which helps in measuring the difference between datasets. This term helped in proving the effectiveness of FedProx mathematically[29].
Firstly the server selects random devices to sent the global model. Once it reaches the devices it starts running locally on the local datasets available in the device. Local models train with the proximal term which keeps them restricted within a range. After training fully or partially it sends update to the global model where all updates gets aggregated. This process runs in a cycle till it reaches optimal result[12].
To validate the claim this paper used datasets like MNIST for digit recognition with over 1000 clients and 69 samples on an average. They extended MNIST for federated use, took 200 clients from FEMNIST with 18k sample on an average. 143 and 772 clients from Shakespeare and Sent140 with 500k+ and 40 samples on an average. Researcher also use synthetic dataset which were specially designed to simulate heterogeneity in the distribution system of data in local models.

While in case of 50% stragglers and 90% slow devices, fedAvg performed poorly. FedProx on the other hand did a great job in maintaining accuracy and stable convergence which was also significantly faster then fed average. This proves the concept of FedProx in case of system heterogeneity. In case of statistical heterogeneity, fed average was affected very badly as the data were not identically distributed. In these situations where the dataset is usually non IID, FedProx reduces the size of the local model divergence. It runs a relatively small model according to the device and dataset. As a result the model provides a stable local update. All together FedProx gives a lower gradient variance. In different situations where fed average faced challenges and performed poorly, FedProx stepped up and showed great accuracy in these extreme cases. Accuracy went up to 22% from fed average. It improved the federated learning algorithm significantly without compromising with privacy or accuracy.
Moreover, FedProx is not just a upgrade but also a crucial framework which helped federated learning algorithm to tackle the challenges which federated average could not solve.

2.2.1.3 FedNova

In Federated Learning, clients differ in dataset size, computing speed and optimization methods, leading to heterogeneity in the number and quality of local updates[12, 15, 20]. This results in an unbalanced local model across clients, finally inconsistency while aggregation and potentially harming convergence. Also, many non-practical analyses do not consider unbalanced local updates(i.e., $\tau_i = \tau$), which do not portray real world scenarios[12, 15, 20].
These issues can be fixed by taking a target number of local updates for each clients. This approach will solve objective inconsistency, but waiting for the slowest one can significantly increase total time[20]. Then there are more advanced approaches such as FedProx[15], VRLSGD, SCAFFOLD, modeled to handle non-IID data, are more appropriate to use for achieving objective consistency, yet they result in slower convergence or require additional communication and memory.
Considering all these issues the authors of [20] proposed FedNova, a novel federated optimization algorithm that accomplishes objective consistency, while achieving better results than the other state-of-the-art algorithms. The update formula:

$$x^{(t+1,0)} - x^{(t,0)} = -\tau_{\text{eff}} \sum_{i=1}^{m} w_i \cdot \eta \cdot d_i^{(t)},$$

Here $d_i^{(t)}$ is the normalized gradient form client i and w_i are the aggravation weights satisfying $\sum w_i = 1$, τ_{eff} is the effective number of local steps, adjusting the impact of each update.
Each component plays a critical role; Normalized Gradient ($d_i^{(t)}$): A moderation of local stochastic gradients that ensures balance across clients, irrespective of optimizer or the number of local steps[20]. Aggregation Weights (w_i): it controls every client's updates priority, ensuring unbiased updates[20]. Effective Number of Steps (τ_{eff}): also measures the weight of local SGD steps per communication round. The server can scale up or down the effect by updating this term[20].
FedNova has many features, its performance can be further improved by applying momentum on the server side[30]. Or using cross-client variance reduction

such as VRLSGD and SCAFFOLD. Furthermore, FedNova is also compatible with and complementary to gradient compression/quantization[31] and fair aggregation techniques[32].
Also FedNova has the luxury on choosing hyper-parameters and local solver, unlike FedAvg. FedNova allows clients to choose local solvers such as GD/SGD with decayed local learning rate, GD/SGD with proximal updates, GD/SGD with local momentum, etc. Using these we can also rest the central server from calculating , which can be updated through local solvers. Previously mentioned cross-client variance reduction techniques can further accelerate the training on client side[20].
This algorithm was evaluated using two setups; Logistic Regression on a Synthetic Federated Dataset and DNN trained on a non-IID partitioned CIFAR-10 dataset. Then during the test phrase many changes were made to ensure fair evaluation between all algorithms. And from the result we can say that FedNova achieved 6-9% improved test accuracy over FedAvg throughout various test cases. The authors has also done a test using FedNova-Prox, which has achieved 10% higher test accuracy over FedProx.
Overall, FedNova has achieved excellent results upon FedAvg, and its dynamic to be applied in addition to many things can further accelerate the results, proving FedNova a very flexible and efficient approach.

2.2.2 Dropout in Federated Learning

Dropout in Federated Learning (FL) serves two purposes, as a regularization approach and a method of uncertainty estimation. Dropout is also being used in decentralized environments such that the data is non-IID and the privacy constraints limit the ability to share data; in such cases dropout methods are being used more frequently to build robustness into a model and carry out personalization among the clients. Dropout has inspired a number of regularization methods for neural networks such as DropConnect, maxout, StochasticDepth, DropPath, ScheduledDropPath. With some probability , at every training step, some of the neurons are dropped out or in other words, they are removed temporarily together with their connections. This eliminates the probability of this model being extremely dependent on certain neurons and promotes more distributed and powerful features to be learnt. In the inference step, one exploits all neurons but the outputs are scaled to correspond to the corresponding values during the training session. In doing so, the dropout process has the effect of causing the entire network to act more like a collection of several smaller networks and therefore enhances generalization, i.e. prediction of data previously not seen.Therefore, dropout as a method to minimize overfitting in FL can also take center stage as a major component in measuring uncertainty, weighting client input, and permitting scalable individualization in heterogeneous learning settings. Our research also emphasises on dropout and tries to find a better solution that will also help with mitigating bias in our central server. Thus, it is necessary to understand the scopes and advancements of some of the dropout methods that relate to our approach. Hence, we will discuss some methods in these subsections.

2.2.2.1 FLuID

A subset of machine learning called federated learning uses each device to run and train models without transferring data to the server[12]. Stragglers are the

primary obstacle in this process[33]. Stragglers are relatively older devices with lower specifications that limit their speed capacity. These stragglers essentially lead to inefficiency by slowing down the entire training process. This study presents FLuID (Federated Learning using Invariant Dropout), a novel system that uses invariant dropout to address this issue. FluID adapts to each device by dynamically eliminating unnecessary or less significant neurons. This technique guarantees great precision and prevents straggler devices from affecting training time. There are a number of other techniques, such as random dropout, which creates a submodel for the straggler devices by randomly dropping neurons; nevertheless, the accuracy is impacted. Another technique, called ordered dropout, drops neurons according to a predetermined patterns[34]. However it is unable to handle all devices. The invariant dropout technique, on the other hand, tracks the training time after sending the global model to each device. It finds stragglers by comparing the training times of each gadget. Then, using this real-time data, FLuID determines which neurons are not making a significant contribution to the training process and dynamically eliminates them. This choice must be adjusted to fit the device. Only the neurons that provide the biggest contributions to the process are left, which helps the sub models attain high accuracy. Depending on the real-time data, it sends sub models to stragglers and the complete model to the higher-end devices. After that, the devices train the model and update the global server. Accuracy is guaranteed by this procedure even while training on several devices[35]. Three distinct datasets Shakespear, CIFAR-10, and FEMNIST were used to evaluate FLuID. In every dataset, FLuID performed better than both random and ordered random. For CIFAR-10 and SHAKESPEAR, it outperformed FEMNIST in accuracy by up to 1.6% and 0.6%, respectively, by 0.3%. In comparison to other models, the stragglers' training time increased by 26% and the overall training time decreased by 18%, demonstrating a notable improvement. FLuID occasionally causes barriers that result in overheads because it depends on real-time data to modify its sub models[36]. Additionally, future technologies will be significantly more expensive, making them more difficult to use. All things considered, FLuID provides a superior solution to the federated learning straggler problem.
In spite of these limitations, it has a great deal of promise for real-world applications in the future because it provides high precision in a short amount of time.

2.2.2.2 Meta-Variational Dropout

MetaVD is a method that is newly introduced and specially made for Personalization Federated Learning (PFL) where without sharing their private and confidential data users usually train a shared model.By doing this, two significant problems like (non-IID users having a very different types of data and very small types of data). In MetaVD hypernetwork(a unique helper model) is used so that the user can get a shared model that is slightly different through adjusting the dropout(the usage time). This increases the efficiency and helps to work a bit more finely for each individual. It smartly combines all the users model into one single model on the basis of the certainty or the surety of the model.In MetaVD ,training time gets faster and the amount of data that is sent between server and the user also gets reduced which allows it to work better than other older methods.
FL algorithms trains global model through altering among local training on global aggregation and client devices on a central server.In that, every single client can

minimize their own loss as they uses local data as the server they are using Combines through a weighted average and here all the weights are based on the local dataset sizes and as a result the process allows data distribution without sharing raw data and the client privacy is also being ensured with this process.
As FL allows all the devices train in a model that ensures data privacy,it also faces many challenges like non-IID because all the devices cannot join all the training rounds and their very small amount of data and excessive communication cost.SO for avoiding all of this Bayesian FL method is introduced as it uses probability distributions in place of a single value that handles uncertainty better and improves learning.Methods like FedBE, FedAG, FedPA, performs better in most cases but still have some issues like high computation cost.
The study introduces a new federated learning named MetaVD that has an accuracy that is much improved when the client data is very limited or different as it uses a smart helper model named hyperwork that helps to adjust the updated parts of the main model which makes the learning more reliable and personalized. MetaVD reduces the cost of communication as it makes the model smaller through personalization. Though there is more complexity,in most of the cases it works really well in comparison with the older model as some clients don't often join training most of the time.

2.2.2.3 DropBlock

Drop Block,a regularized technique that is designed for the betterment of the performance of convolutional neural networks in which an entire block of features drops from the feature map instead of randomly removes individual activations of traditional dropout[37].As the removal is more structured and effective than CNNs as dropping the whole regional forces the network for learning more generalized patterns,DropBlock hugely increases the model accuracy considering their benchmarks like COCO compared to other standard dropout.this includes all, the key parameters like drop rate and block size and it also performs at its best When through training the drop rate gradually gets increased as the method helps to prevent overfitting and improves generalization in deep networks.
DropBlock is a far more better regularization method than drop out or SpitalDropOut in ResNet and other deep network.When its applied in the later layers on both convolution using a large block size and also skipping all other connections.By dropping connected regions in place of using random units,it plays a huge part for the model to learn more generally and spatially distributing features. As a result, the performance gets boost like advanced models like AmeonaNets, increased accuracy like ImageNet.
Its structured regularization methods of CNNs drops blocks of features rather than individual units like Dropout[13].Through encouraging learning it improves the model performance .from a wide spatial area.On the other hand , experiments on COCO and ImageNet reflects that DropBlock easily out performs dropout. Specially, if we apply it on skip connections, convolution layers and when gradually the amount drop is increased in training.

2.2.3 Cancer Detection in Federated Learning

The most promising application of AI to date is in the field of cancer diagnosis through deep learning techniques based on medical imaging and other clinical information. Yet the ability to share such data across hospitals comes into contention with privacy law stipulations, which hinder access to patient information. Federated Learning (FL) is a feasible approach to collaboratively training models without revealing that private data to anyone involved. Therefore, this section presents the state of the art in FL related to cancer diagnosis and reveals shortcomings of the present studies, namely, data heterogeneity and generalization, which this study intends to address.

2.2.3.1 Skin Lesion Classification from Dermoscopic Images

Skin cancer, primarily melanoma, is considered extremely hazardous and through early detection lives can be saved. But, it is difficult, even for the doctors, to know whether a spot is bad (malignant) or is a harmless brown spot (benign). It is correct less than 65%-80% of the time[38]. Special close-up shots of skin can be useful to doctors, who can examine them with the help of dermoscopic images. However, it is very difficult to detect the slightest distinctions between cancerous and non-cancerous spots. So, the researchers chose to put deep learning into full science mode and divide the image of skin into two categories: malignant or benign. They used the popular VGG16 model, a deep neural network that had been trained on huge sets of images such as ImageNet[39], and they performed so-called transfer learning, that is, they took the experience VGG16 learned on all those images and adjusted it to detecting skin cancer. They have tested three methods of training the network over the publicly available dataset (the ISIC Archive dataset[40]) in their experiments. They discovered that the results were best with fine-tuning of the pre-trained VGG16 model compared to the alternative methods of initializing the model with blank models or freezing the majority of the layers.In a nutshell, the conclusion is that the most important thing is to have a trained model and tune it to the point of making more precise skin cancer detection.
The dataset that we have used was the ISBI 2016 Challenge which initially consisted of 900 training images and 379 test images[40]. Although it was actually disproportional. Accordingly we reduced it and modified it so that there were 346 training images and 150 test images and we also prepared the pictures to fit the model by normalizing the pixel values, reducing them to square pictures, and then scaling them to a 224 x 224 pixels before putting them into the model. Then we performed data augmentation to ensure the dataset was larger and reduce the risk of overfit, which included rotating the images up to 40 degrees, moving the images laterally, zooming, and inverting them. In the case of the model, we have used VGG16 but replaced the final layer with one that would support two only groups (malignant or benign), and the output would be a sigmoid since it is a two-way binary classification issue.We experimented with three training strategies, which included training fully and from scratch, transfer learning with frozen convolutional layers (VGG was a feature extractor), and transfer learning with fine-tuning (retraining only the upper layers and the classifier). The models have been constructed in Keras with Theano as the backend and trained in an NVidia TITAN x graphics card, the RMSProp optimizer, a batch size of 16, and a dropout of 0.5 in the fully connected layers[41].
The authors evaluated the appropriateness of the models by examining the aspect of

accuracy, sensitivity, specificity, and accuracy. Their results were not so good when they trained the model entirely at the beginning and the accuracy was only 71.87%. The one which applied transfer learning on frozen layers actually achieved good results in the training images, with an accuracy of 95.95% and a sensitivity of 96.21% but failed in the test images due to overfitting where the accuracy deteriorated to 68.67% and sensitivity dropped to 33.11%. The most successful version was the fine-tuned version with a result of 81.33%, 78.66% and 79.74% accuracy, sensitivity and precision respectively on the test data. These values improved on previous values on the ISBI 2016 data, which had a sensitivity of approximately 50.7% and a precision of 63.7%[41]. Based on this, the study demonstrates that the fine-tuning is more effective as it utilizes the characteristics discovered by ImageNet but modifies them to the skin lesion images to provide more balanced and accurate outcomes.
In this paper, the authors indicate that, in the fine-tuned VGG16 model, the model assists physicians to identify skin cancer more effectively by scoring higher in terms of accuracy, sensitivity, and precision compared with other researches. They also mention that it has certain limitations, such as a small dataset size and a possibility of overfitting. To improve their use in the future, they recommend larger and more diverse data to make the model more robust, more robust anti-overfitting measures, and attempt pre-training on skin-specific image datasets rather than general ones such as ImageNet. Altogether, the analysis demonstrates that fine-tuning of a transfer learning is effective with medical images in the case of limited data, and that the systems might aid in the detection of melanoma at its initial stages and provide physicians with useful information.

2.2.3.2 Classification and Detection of Skin Lesions Using Multi-Layer CNN

Skin cancer is hyper prevalent and melanoma is one of the most perilous. Although not the most prevalent skin cancer, it claims the lives of many individuals. The faster you have caught it, the better. Due to the presence of numerous spots that seem similar to each other, doctors have a hard time identifying the dangerous ones. Machine learning and deep learning are being applied by scientists to identify things in the pictures which we humans would have continuously overlooked.The researchers constructed a new deep learning model to be used in this research named SkinLesNet. It is a multi-layered convolutional neural network, which is able to identify three types of skin spots nevi, seborrheic keratosis, and melanoma[10].The most interesting thing is that we can apply it to images which were captured by using smartphones in the PAD-UFES-20[4] data. Not only to the images of high-quality dermoscopes.And that makes it a potentially applicable tool in real life.They trained it on other datasets such as HAM10000[5] and ISIC[3] and compared its results to other popular models such as ResNet50 and VGG16.
The researchers employed a dataset named PAD-UFES-20 Modified with 1,314 images of three skin spots in this research. All the images were made using smartphones to demonstrate that the model might be applied in real life. They preprocessed the images by way of resizing them to 224x224 pixels and data augmentation, which included flipping, rotating, and moving the photos, so that the model will have greater diversity and will not overfit. SkinLesNet is a basic 4 layer CNN which not only has convolution and pooling layers but also dropout to prevent overfitting and ReLU as its activation[10]. The last layer involves the selection of the picture by

SoftMax which category of the 3 it belongs to. The model was trained using Adam and a learning rate of 0.001, batch size of 32, and 100 epochs. The dropout rates were 0.5 and 0.3 on certain layers and others respectively. In the case of testing by us, SkinLesNet achieved an accuracy of 96% on the PAD-UFES-20[4] Modified set. Precision was approximately 97, recall approximately 92 and F1 -score approximately 92. To make comparison, ResNet50 was only accurate at 82 percent and VGG16 at 79 percent on the same set. In the HAM10000[5] set, SkinLesNet was found to achieve an accuracy of 90% which was considerably higher than the 80% and 75% of ResNet50 and VGG16 respectively. It reached 92% of the results on ISIC[3], surpassing the results of ResNet50 and VGG16. Overall, these findings demonstrate that SkinLesNet is better than such older unpopular models[10].
The authors indicate that although SkinLesNet is quite effective, it also has certain issues, such as overfitting and the lack of various types of data[10]. They propose that larger and more diverse datasets might be used by researchers in the future, experimenting with more effective methods to prevent overfitting, and with novel techniques such as active learning or self-supervised learning. They also explain that image segmentation could be used in conjunction with classification, which would assist in enhancing detection. In general, the experiment demonstrates that SkinLesNet is a useful application to diagnose skin spots, particularly since it can take images using a smartphone, which is more likely to detect melanoma at an early stage. This type of system could save lives and assist doctors with more improvements and larger data.

2.2.3.3 Skin Cancer Classifier for an Imbalanced Dataset

Skin cancer is dangerous as it leaves no chance of survival if detected late. Moreover, current diagnostic methods require a lot of time for precision and are limited by the numbers of skilled dermatologists[42]. To inspire further studies on this and to highlight the current advancements, this study proposes a novel framework for classifying skin cancer and show that it outperforms the state-of-the-art approaches, particularly in case of imbalance datasets.
This paper also mentions the significance of medical imaging-such as dermoscopy, CT, HRCT, MRI- in cancer diagnosis. Thanks to the rise of AI-based algorithms in computer-aided diagnostic (CAD) frameworks show better results than the clinical experts in detecting disease through imaging[43]. Also, this development is assisted by high-speed internet, computing resources and cloud storage, which enables widespread dataset sharing and model development.
One of the major concerns in skin cancer classification is data imbalance[44]. To add to this, this study uses the HAM10000[5] dataset, which has imbalance across 7 classes. Hence the proposed approach to employ data augmentation, which increases the sample size by transforming existing images. It increases the dataset of size 10000 to 32000 images with around 4000-5000 images for each class[42].
Here three CNN architectures are compared, which are AlexNet- the starting CNN architecture that popularized deep learning for image classification. Then, InceptionV3- a more parameter focused deep CNN derived from GoogleNet. Finally, RegNetY-320 - the best approach in this work[9]. They were trained with specific parameter settings (learning rate 0.01 and 0.001, 20 epochs, batch size 100, 70-30 train-test split). In both different learning rate settings RegNetY-320 performed best[42].

In both settings and for all of the approaches we generally had an imbalance dataset. From comparison, we see that lower learning rate helps with overall result. As it helps the model to train more efficiently with less generalizing. Next the proposed approach to data augmentation helps to balance the dataset[44]. As it uses various configurations of image augmentation such as- rotation, translation, rescaling, flipping, shearing, stretching on the images randomly cropped patches, to increase the dataset and balance the dataset. Previously, maybe one class had 5000 images while others with a few hundreds. Now they have around 4000-5000 images each. And by doing so, we see an overall increased performance across all settings and architecture. Not only does it improve results but also we see significant high value in ROC value.which indicates more True positive rate(TPR) and less False positive rate(FPR)[42].
Although this approach achieved such performance, the author also mentioned some limitations of it. As this approach performed well with this dataset, it has not been tested with other datasets, hence, the possibility of overfitting on these datasets can not be ignored[44]. Also, this approaches bias behavior toward this and a similar dataset indicates a limitation for this work.

2.3 Literature Review

In the previous section we have introduced the fundamental concepts by reviewing many papers related to the background of our work. Although there are other papers that we have studied that are closely aligned to our research.
These papers do also follow a similar approach to those papers that we have reviewed till now. While some papers may contain related concepts, the paper's work might not be related to our work. Yet the insight these papers will provide is essential for later research progress and analysis.

2.3.1 FL-FDMS (Friends to Help)

In a typical scenario, each client or device is supposed to participate equally in each training session. However, not every client can participate equally in every round due to a number of unforeseen reasons, such as poor battery life or network problems. Client dropout, sometimes known as passive partial participation, is the problem in question. Although a lot of effort has been done over the years on how to choose active clients, the issue of client dropouts has not received as much attention. The authors of this research identified this gap and attempted to close it by utilizing client resemblance, or what they referred to as "friendship," to lessen client dropout[45]. By substituting stalled or preset fixed values for idle clients, traditional federated learning models often overlook them, which can lead to biased results and affect the model's overall accuracy and performance. This paper's writers didn't simply disregard these dormant customers. Rather, they suggest a novel paradigm in which they replace the update for the inactive customers with that of other active clients who have comparable datasets and distributions. The server determined the clients' pairwise similarity scores for every round. The value of the normalized cosine similarity ranged from 0 to 1. Compared to stalling values or randomly changing values, this produced a more accurate output.
According to Lipschitz Smoothness, which states that the gradient moved gradually, the local gradient estimator was unbiased, and the local and global variance was

bounded, the authors provided several crucial presumptions in order to comprehend the theoretical ramifications. Based on these presumptions, the authors demonstrated that the suggested approach produced improved convergence in a client dropout scenario.
The experiment's datasets were CIFAR-10, which recognizes objects in pictures, and MNIST, which classifies handwritten digits. Two types of data settings were present. One client was randomly portioned, essentially non-IDI, while the other clients were grouped with the identical labels in the first setup. The comparison's baselines included the ideal scenario, in which all participants participated, the dropout or baseline scenario, in which a client dropped and the model was disregarded, and the stale substitute, in which stalled data were used in place of updates. There were two local epochs per round, a local learning rate of 0.1, and a global learning rate of 1. Accuracy vs training round count under two dropout ratios 0.5 or 50% dropout and 0.7 or 70% dropout ratio was used by the authors to assess the test. The accuracy of FL-FDMS was 96% in the MNIST clustered environment with α at 0.5, while the dropout and stalled rates were 88% and 91%, respectively. Even with α at 0.7, FL-FDMS produced an accuracy of 94%, which is noticeably better than dropout and stale. With dropout at 30% and stale at 36%, FL-FDMS maintained 44% accuracy in the CIFAR-10 clustered environment with α at 0.7. FL-FDMS maintained 40–43% accuracy, outperforming stale and dropout in non-IID settings as well.
FL-FDMS is entirely dependent on its ability to make friends with inactive clients. According to the authors' pairwise similarity scores, clients from diverse clusters displayed lower similarity scores, while grouped data settings provided higher similarity scores. Because CIFAR-10 uses CNN models and has more complicated data, the friend-finding procedure is more successful than MNIST[46].
Furthermore, this research made a significant contribution to the field of federated learning by first highlighting the need for improvement in the dropout section and then putting up a highly admirable FL-FDMS model to address the problem of passive participation. This FL-FDMS shown that it lowers the overall model error and dynamically discovers buddies for inactive clients.

2.3.2 FedAgg

A novel federated learning model was put forth by the authors of the study FedAgg: Adaptive Federated Learning with Aggregated Gradients. Even though there are several models for federated learning that have been around for a while, they still fall short in certain situations, such as device heterogeneity[12]. The authors of the suggested approach concentrated on developing an adaptive learning algorithm that was intended to improve accuracy in non-IDD data distribution systems. Conventional methods frequently fail in non-IDI data delivery systems[27, 18]. By placing a penalty term to quantify the model deviation, FedAgg addresses this problem by including an aggregated gradient term in the local model update, which aids in adjusting the learning rate.
Other developments in the subject of adaptive learning include adaptive hyperparameter updating, which includes FedAdp and Adaptive-B[47]. Adaptive-B determines the ideal batch size based on the deep reinforcement learning rate. The adaptive client selection mechanisms follow, such as FEDEMD, which identifies data similarity to

pick clients, and FedSAE, which computes training loss to select clients[48]. FedMDS, which works with clusters, and AsyncFedED, which computes the euclidean distance between stale and current models, are two components of adaptive asynchronous aggregation[49].
Clients in a typical FL model do not exchange raw data. Rather, they run a model on their device and send a local update, which eventually spreads to other devices[12]. Since numerous circumstances, such as non-IDI data distribution, can lead FL models to deviate significantly from the global model update, regulating the divergence between local and global updates appears to be the primary problem[16]. The authors developed a novel decentralized adaptive learning rate that takes into account both local parameters and the average parameter of all clients. This guarantees that the client may estimate the mean field terms and freely modify its learning rate.
It begins by estimating the mean field terms, which roughly correspond to the average and average gradient of the local parameters. The global parameters and mean field terms are initialized in this iterative process, which then computes the fixed points and updates the mean field terms until it converges within an error margin. Once this is finished, the global model parameters and mean field terms are initialized by the main FedAgg algorithm. Clients use the mean field terms to calculate their own adaptive learning rates. The client trains a local model using these rates, and then uses the weighted average of the local models to update the global model. To demonstrate FedAgg's convergence, the authors offered a theoretical convergence analysis. In addition to establishing upper bounds for optimality, the authors examined how convexity and nonconvexity affected the global loss function[50].
For the experiment, the authors used four datasets: CIFAR-10 and CIFAR-100, which have 50k training set sizes and 10 and 100 classes, MNIST, which has 60k training sets and 10 classes, and EMNIST-L, which has 124k training set sizes and 26 classes. FedAvg[12], FedAdam, FedProx[15], FedDyn, FedBABU, and FedGH were contrasted with FedAgg. Evaluations were conducted using neural network architectures such as LeNet-5, AlexNet, VGG-11, RestNet-18, and Google LeNet DenseNet121.
FedAgg outperformed other baseline models in terms of accuracy across a range of participating clients. Whether there were 100% or 20% of participants, FedAgg remained superior. FedAgg outperformed FedAvg on CIFAR-100 by 7% in extremely diverse data. In addition to increasing accuracy, it required fewer iterations to achieve minimal loss. FedAgg only required 100 iterations to obtain low loss on CIFAR-100, whereas FedAvg required 200. This adaptive learning rate technique is more beneficial for complex models such as RestNet-18.
Furthermore, the authors demonstrated how the adaptive learning rate enhances federated learning's overall performance. This aggregated gradient and adaptive learning rates can solve the heterogeneity problem[50].

2.3.3 FedRolex

FL is a machine learning paradigm that trains models from distributed clients with private data under the coordination of a central server. The paper "FedRolex: Model-Heterogeneous Federated Learning with Rolling Sub-Model Extraction" by Alam et al. [51] pointed out the limitations of model-homogenous federated learning methods which require identical models across all clients. In this work, we focus on

cross-device FL where clients are usually resource-constrained edge devices. The majority of existing crossdevice FL studies focus on the model-homogeneous setting [12],This makes it simply vertical and unaccommodating to lower-end devices.
One primary challenge in model-heterogeneous FL is the aggregation of heterogeneous client models. To address this challenge, knowledge distillation (KD)-based approaches have been proposed [52], in which the client models serve as teachers and the server ensembles the knowledge distilled from the individual client models. In particular, FedDF [52] distills knowledge from a set of classifiers trained with private data from a federation of client devices. But the problem with KD-based methods is that they are vulnerable to backdoor attacks and have some security concerns. In order to overcome the weakness of KD-based solutions, a complementary model-heterogeneous FL solution such as partial training (PT) has recently been proposed.
According to the ways which the sub-models are extracted out of the global server model, current approaches of PT-based techniques can be summarily divided into two categories, namely, random sub-model extraction and static sub-model extraction. Specifically, inspired by the dropout technique commonly used in centralized training [13], Federated Dropout [12, 15, 20]. Though, it resolves the problem of the KD based approaches, it also presents its limits which are the global model size is constrained to the most powerful client and the second is that training in the global model is uneven since clients are forced to utilize their resources in training distinctive sub-models.
Therefore, the authors has proposed FedRolex to overcome these challenges. FedRolex is a partial training(PT) based approach that supports model heterogeneity. With this approach, it is possible to scale a server model that is bigger than each client model. It operates under a rolling scheme of sub-model extraction in which sub-model gets trained uniformly over the layers.It also re-estimates all parameters of global model. It alleviates the client drift that is occasioned by the inconsistent model architectures across the globe.It selects the sub-models randomly and ensures even training convergence and reduces the performance degradation which is caused by uneven parameter updates.
The working process of this algorithm is that it gives each client a smaller part of the full model that fits its device's capabilities. Subsequently, it rotates which element of the model a client trains over time, so that with time all elements of a complete model undergo training equally. The sub-models are trained locally and their parameters are not always aggregated at the server, similar to secure aggregation protocols and do not require sharing of the data publicly, as it is in the case of knowledge distillation (KD)-based approaches like FedDF, DS-FL and Fed-ET. Conversely, it also outperforms both the PT - Based KD based baselines in both terms. FedRolex is a federated learning algorithm that is developed on heterogeneous and a bigger scale setting..It helps in both global and local model accuracy ,mostly on low end devices using a dynamic rolling extraction strategy which leads to faster and better parameter efficiency, convergence and makes it more suitable for real life application.

2.3.4 Adaptive Self-Distillation

From previous papers we already know that heterogeneity results in slower convergence, also these results may lead to nonoptimal results, due to local bias. Till now we had seen many approaches that contributed on the central server side to

improve aggregation and better client selection, whereas this paper by [53], proposes a regularization technique based on adaptive self-distillation (ASD) for training models on the client side. Because of that this method can be integrated to the state-of-the-art FL algorithms for further boost.
The idea for this approach was initiated from FedProx[15], SCAFFOLD[16] use of regularization at the client side, but ignoring the representation of the global model. Also this approach does not require any additional data. Another term Knowledge Distillation (KD) introduced by Hinton, Vinyals, and Dean [54], is being used here to get some info from a pretrained model.
For comparison and analysis, they used publicly available codebase to build experimental setup. And to simulate the real world scenario they formulated every variable, using Dirichlet distribution. Then they compared FedAvg[12], FedProx[15], FedDyn, FedSpeed and more with ASD and without ASD, in various amounts of heterogeneity. The datasets were, CIFAR-10 CIFAR-100 and Tiny-ImageNet. In every case with ASD their accuracy was improved.

2.3.5 Locally Adaptive FL

FedAvg uses the same step size that ensures the balance among the clients, but the problem with FedAvg is that it might slow down the learning process. One of the reasons behind it is that various clients have various data, and the same strategy might not work well for such variety. I have become acquainted with the paper in the title Locally Adaptive Federated Learning, the method presented in it is based on a strategy similar to FedAvg, but has new concepts beyond a mere averaging. It is called FedSPS, which is based on locally adaptive federated learning. In federated learning carried out in the proposed way, every client adjusts the algorithm separately to the local geometry of the problem, e.g., the peculiar shape of the manifold or the geometry of the loss surface or the data distribution of the client. This method is more effective, especially when the client data is diverse (non-IID).
The rationale attributed to the invention of a locally adaptive federated optimization algorithm is that in FedAvg, it applies vanilla SDG with a fixed stepsize, which not only is resistant to heavy-tailed gradient noise(prevalent in training language models such as ViT) and non-IID data among clients but also is not adaptable to training dynamics. Adaptive techniques are important since in centralized learning(Adam and AdaGrad), the optimization of controlling the learning rate so that the performance can be improved and SGD then can be surpassed. Then again, methods like FedAdam and FedAMS only bring adaptivity at the server side, but not to each client. The limitation of existing adaptive federated learning methods like Local-AMSGrad is that, although it allows partial local adaptivity, but still has to maintain and share the step sizes. This results in the failure of customization of the learning process based on each client's unique data. Thus, the authors developed an idea of a fully locally adaptive federated learning algorithm they named FedSPS. FedSPS algorithm is constructed based on SPS (Stochastic Polyak Stepsize). It incorporates SPS locally on each client and also guarantees strong convergence.A FedSPS extension is FedDecSPS. Further follow-up works have come up with various ways to overcome the limitations of vanilla SPS, such as when optimal stochastic loss values are not known [55], or when the interpolation condition does not hold , as well as a proximal variant for tackling regularization terms[56]. It is implemented by reducing step

sizes, where interpolation does not mean exact interpolation.. The founding of the experiments is FedSPS and FedDecSPS might match or exceed the performance of FedAvg and FedAMS.
In their problem formulation, they referred to each client as having its own loss function $f_{x_i}(x)$ wherein this gives an idea about how well the data is performing on the data of the client. All this is done with the sole aim of minimizing the average loss of all clients:

$$f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x)$$

As it is a non-IID, the main solution for every client might not be the same. That is why the authors have introduced a new learning method where the learning rate is adjusted using the gradient, current loss, a lower bound of the minimum loss(which is mostly 0 in practice), and a constant cap so that the step size does not get too large [55].
In the FedSPS approach the algorithm begins by training the adaptive step size per client by using SPS per client. Then they start to train with their own step sizes locally. After a few rounds, clients sent the updates to the server for averaging, which allows better learning for each client, and unlike other methods, it does not need any extensive tuning. They have also experimented with some other variants like FedSPS-Normalized and FedSPS-Global, but they did not outperform the main approach. The authors evaluated their proposed methods, FedSPS and FedDecSPS, both depending on the IID and non-IID data settings. In every experiment round, they took 500 communication rounds, 5 local steps from each client and batch size at 20. Other parameters are 10 clients took part in every round of 100 clients and the samples of every client included 2 different classes of non-IID data. For non-i.i.d. experiments, we assign every client samples from exactly two classes of the dataset, the splits being non-overlapping and balanced with each client having same number of samples[12, 15, 20].
FedSPS then performed better than best-tuned FedAvg in a convex setting, and it is much less sensitive to hyperparameter modification in IID setting on MNIST dataset. Although FedAMS slightly outperformed it but it still needs an extensive grid search over server learning rates. In case of strong data heterogeneity, FedSPS exceeded the performance of FedAvg, FedAMS, FedAdam and MIME. In non-convex environments, the authors benchmarked FedSPS and FedDecSPS over a variety of tasks MNIST, including LeNet, ResNet18 on CIFER-10, both in IID and non-IID data conditions. On non-IID cases in which classic approaches tend to struggle, FedSPS and FedDecSPS were compared to FedAvg and FedAMS and showed to be much better at training loss and test accuracy with good learning stability.
In short, this paper suggests FedSPS which employs a locally adaptive federated optimization algorithm with stochastic Polyak stepsizes in an attempt to achieve a faster convergence, capable of operating with overparameterized settings. As well, the authors propose yet another variant, in that it applies a decreasing stepsize, allowing the exact convergence to be appropriately guaranteed, is FedDecSPS. Both of the approaches are able to support robustness to hyperparameters and demonstrate good performance on both convex and non-convex tasks. It outperformed even FedAvg and FedAMS both in IID as well as non-IID setting in certain circumstances.

2.3.6 FL for Indoor Localization

The paper by Park et al. [57] mainly focuses on how to improve indoor localization with the use of a method called fingerprinting. The fingerprinting technique involves estimation of the location of a person according to the features of the wireless signals such as RSS (Received Signal Strength)[58]. Fingerprint data collection is expensive and also time-consuming. In parallel, as the size of the dataset increases, the localization system typically demands a more complex model with increased computational burden [59]. Therefore, the authors tried to solve these problems using federated learning. Although it protects the privacy and reduces the power and communication cost, it has its drawbacks as well, like each client device's data is mostly non-IID and unbalanced. Sometimes in federated learning, important data from unexplored areas can be missed with common solutions as well.
To solve these issues, the authors came up with a new method that considers model uncertainty to measure the reliability of every client in indoor localization, which is also a method in federated learning. Thanks to the aforementioned advantages, researchers considered the adoption of FL into advanced wireless communications and indoor navigation [60]. And while combining them, it uses this reliability score to weight the client models. To estimate model uncertainty efficiently, instead of using the Bayesian neural networks, they applied the Monte Carlo dropout, which is less expensive than the Bayesian Neural Network. The methodology was exerted to an actual wireless fidelity wireless RSS fingerprinting data set. This strategy will aid in making more decisions to trust the client in view of model updates.
Among the central questions, there is how federated learning is applied in indoor localization. To answer this, the clients collect their data on their own through the Wi-Fi signal strength from different locations [58]. Because of these different locations, the collected data from the clients is uneven. The clients in federated learning also train the models but they will do so on their data using which they will send updates to the central server. Then the central server combines the updates and sends them to the global model. In this combination, every client's model is assigned a certain weight, which determines the influence of each client's model on the actual global server. Bayesian deep learning makes it possible to measure the uncertainty of a NN with a probabilistic approach [61]. As mentioned earlier, researchers used Monte Carlo dropout to calculate model uncertainty; the reason behind this is that it is computationally less costly and also a simpler way to estimate it. In Bayesian deep learning, they have to consider how confident the prediction of the model is, whereas in Monte Carlo dropout, they just need the proximity of that process [62]. The work process of the Monte Carlo dropout is, during training, they randomly turns off some neurons multiple times for the same inputs, and then they get different outputs from the model that form a distribution which is used to estimate uncertainty. There are two types of uncertainty and they are Aleatoric Uncertainty and Epistemic Uncertainty [63].
Aleatoric uncertainty comes from inherent noise in the data, which can't be reduced with more data, and the other one comes from the uncertainty of the model, which can be reduced if more data is collected. The two of them describe the variation of the prediction error jointly. The client would be less reliable if there is more uncertainty in the model.
This system is similar to federated learning, in which it begins with the global model. Then, RSS fingerprints and their corresponding locations are stored as data on the

server. Then clients localize training their model and subsequently transfer their updated weights to the server. Monte Carlo dropout is used to assess the uncertainty by virtue of the server. Here, it performs multiple times the original inputs randomly dropping out the neurons. It estimates variance of forecasts. Then the uncertainty is calculated as the average of the variance. In this case, higher reliability is defined by lower uncertainty. After that, to weight each client's contribution to the global server, the server uses each client's reliability. The clients with lower uncertainty and higher reliability have more influence over the server.
The proposed federated learning algorithm was tested with the help of a real-world dataset named UJIIndoorLoc. The dataset was obtained using 20 mobile clients and 25 different devices in three large buildings. It was sampled out of 20 percent of randomly selected data which was used to establish a validation set. Compared to FedAvg, DNN (Deep Neural Network) and kNN, the proposed method was compared. The root mean squared error was adopted to examine the accuracy of localization. It can tell the distance from the true location to the prediction location. In the test results, it is visible that kNN showed the worst performance with an error of 7.18 meters in localization accuracy, where DNN outperformed with an error of 5.61 meters. The error did not decrease or change because no adjustments in the model were made during the training rounds. FedAvg takes a long time to converge having an error of 7.76 meters after 50 rounds. The suggested approach compared to FedAvg outperformed and the error was 6.06 meters at the end of 50 rounds.In case of training loss, to converge, it took about 37 rounds for FedAvg to have a higher loss of 3.4, where the proposed algorithm took 30 rounds to reach a loss of 2.5[57]. To sum things up, the paper proposed an improved federated learning method where Monte Carlo dropout is used to measure client reliability. Global model is accurate through the approval of more quality models. The proposed method has the accuracy of localization improvement compared to existing FL approaches of 1.7 meters.

2.4 Research Gap

The privacy constraints in medical imaging have led to the implementation of federated learning. Many studies show successful simulated training in a network of hospitals. Most of these implementations assess federated algorithms on artificially divided datasets with training and test distributions remaining the same, only split across clients. Federated learning setups in real clinical applications must deal with real domain shift because each site has substantially different data distributions due to different machines, imaging protocols and patient populations. This simulation approach disguises that. We notice a scarcity of systematic analyses that examine the performance of federated learning models when the advanced solutions exist for known challenges in federated optimization. FedProx's proximal terms are designed to handle systems heterogeneity and stragglers. SCAFFOLD adjusts for client drift by utilizing control variates to estimate gradient divergence. FedNova adjusts aggregation weights based on the number of local training steps. The primary obstacles which these methods assume are due to optimization dynamics, communication efficiency, and statistics of local data distributions. The literature did not take into account how regularization interacts with aggregation algorithms. Many researchers in federated learning work under the assumption that FedAvg works correctly no matter what regularization strategies clients use during their local training. In other words, they

treat aggregation as a simple weighted averaging problem, which is independent of the architecture.
Dropout-regularized networks do not satisfy this assumption but this failure mode has not been studied. The use of dropout is a widespread practice in deep learning that can be found in almost all state-of-the-art architectures of medical imaging. Recently, dropout is applied on attention-based models with rates 0.3-0.5 on critical layers. Independent clients applying stochastic dropout via different seeds train different networks structurally every round. Heterogeneous updates are received by standard federated averaging, which averages all the parameters uniformly, failing to differentiate between the actively trained neurons and those that were momentarily disabled for regularization. Previous papers do not analyze whether the aggregation strategy is mathematically reasonable for dropout-enhanced models, do not quantify performance effects, and do not propose logic modifications that account for regularization-induced parameter masking.
The gap is especially acute for attention mechanisms, where the values of parameters bear semantic significance, going beyond mere feature extraction. Squeeze-excitation blocks, channel attention modules, and self-attention layers learn explicit weighting factors to balance the impact of features. In order to operate properly, these weights must be correctly calibrated (i.e., collapsing a channel weight from 0.8 to 0.4 alters what features the network is attending to). Research on federated learning assumes attention weights are like convolutional filters, applying the same aggregation formula irrespective of the type of parameter: importance score or feature extractor. No prior work investigates if attention mechanisms stay selective and calibrated when cellularity trained, whether channel weights retain their target dynamic range after averaging many times, or if diagnostic feature prioritization continues across global rounds.
Also a similar study whose approach with federated convolutional neural networks which achieved far greater accuracy on HAM10000[5] has several methodological and experimental design choices that differ from our objectives and constraints of our work[64]. They employed aggressive dataset rebalancing using SMOTEENN prior to both centralized and federated training, giving their model enough to learn about all classes. As a result it simplifies the classification task and reduces minority class ambiguity, thereby boosting accuracy outcome. Furthermore, its federated learning setup relies on stratified and homogeneous client partitions, where each client receives nearly identical class distributions. It provides better convergence and stabilizes aggregation, but fully neglects real world deployments, where data is non-IID and institution specific. Hence, their high performance is already within our expectations from our research that performance under controlled and balanced conditions achieve great results but perform poorly when evaluated in cross dataset settings. Not only that, their evaluation is based on a single dataset (HAM10000), with limited emphasis on cross dataset or cross domain generalization. It can be understood that they focused on achieving greater accuracy on centralized federated settings. Meanwhile, our approach addresses real world scenarios and proposes a model that can perform better in such scenarios. Also our approach emphasizes on identifying issues that arise from FL implementation on a real world system, and proposes a solution to it.
The current benchmarks of federated medical imaging are convergence speed, communication rounds, final accuracy, and more. We are not sure what is actually

happening inside the model when we perform federated training, for example do internal representations remain coherent?, Do architectural components work like they are intended to? Are performance gaps due to aggregate corruption or problems inherent to federated learning? This study aims to contribute solutions to these challenges by isolating and quantifying the effect of aggregation-induced degradation on attention-based architectures, proposing aggregation strategies that recognize regularization structures, and validating the resulting solutions on heterogeneous and real-world medical datasets.

Chapter 3

Work Plan

Here, in figure 3.1 we present the structured workflow followed throughout this research. Firstly, we studied existing research on deep learning and federated learning. These approaches to skin lesion classification problems and their limitations were analyzed. Based on this, we proposed a research hypothesis, supported by a comprehensive literature review of state-of-the-art approaches and currently leading research.
Subsequently, we identified some research gaps and then began data collection and preprocessing it, realizing the heterogeneous state of the datasets from different medical sources. To compare and validate experimental consistency, we reproduced the baseline results.

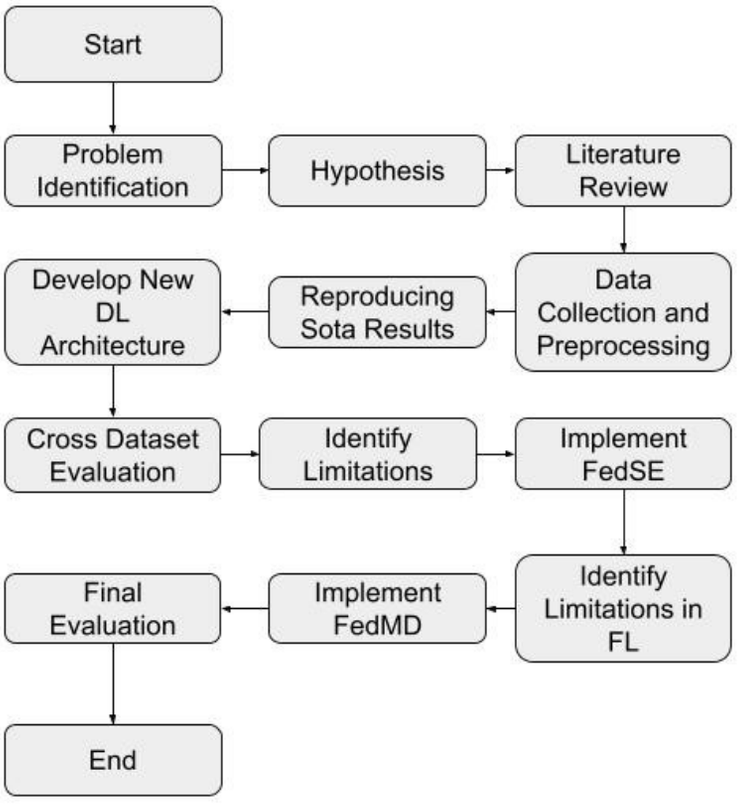


Figure 3.1: Work plan

Building upon this baseline, a deep learning architecture is developed and evaluated

across selected datasets and also using cross dataset experiments to assess it in real world scenarios. From this the initial limitations were confirmed and motivated the federated learning implementation.
As a baseline model, we implemented FedSE, and realized necessity and limitations of baseline federated learning. Finally, to address these issues, an improved model FedMD (Mask Aware Federated Learning) is implemented and thoroughly evaluated to compare the existing results with its outcomes.
Thus, following these steps we identified the limitations, confirmed them and then proposed our mask aware federated approach.

Chapter 4

Dataset Description

4.1 Dataset

For this study, three publicly available skin cancer image data sets are utilized: HAM10000[5], PAD-UFES-20[4] and the ISIC[3] 9 Class dataset. They are publicly available collections of images and clinical information rendered for automated skin lesion classification purposes. Table 4.1 outlines the appearance of different types of skin lesions across three prominent dermatological image collections. As shown, most of the major lesions (actinic keratosis, melanoma, nevus, basal cell carcinoma) are present across all three image sets, while additional lesion types (dermatofibroma, pigmented benign keratosis, vascular lesion) are not present in PAD-UFES-20. Likewise, some lesions (seborrheic keratosis, squamous cell carcinoma) are not present in HAM10000 but are present in the other two collections. This shows the variability and heterogeneity of lesion class presence across skin cancer classification data sets.

Table 4.1: Presence of skin lesion classes across datasets

Class Name	**HAM10000**	**ISIC**	**PAD-UFES-20**
Actinic keratosis	YES	YES	YES
Basal cell carcinoma	YES	YES	YES
Dermatofibroma	YES	YES	NO
Melanoma	YES	YES	YES
Nevus	YES	YES	YES
Pigmented benign keratosis	YES	YES	NO
Seborrheic keratosis	NO	YES	YES
Squamous cell carcinoma	NO	YES	YES
Vascular lesion	YES	YES	NO

4.1.1 HAM10000

The HAM10000[5] (Human Against Machine with 10,000 training images) database is composed of a grand total of 10,015 dermatoscopic images of various pigmented skin lesions collected from diverse populations and clinics. Such heterogeneity makes it an ideal source for training machine learning models designed for diagnostic deployment in the real world. In addition to image level labels, patient characteristics (age,

sex, and location of lesion) are also provided and documented. Lesion types were validated with histopathology for the majority of cases; for some, validation occurred via follow-up visits, expert agreement at clinical diagnosis, or confocal microscopy.

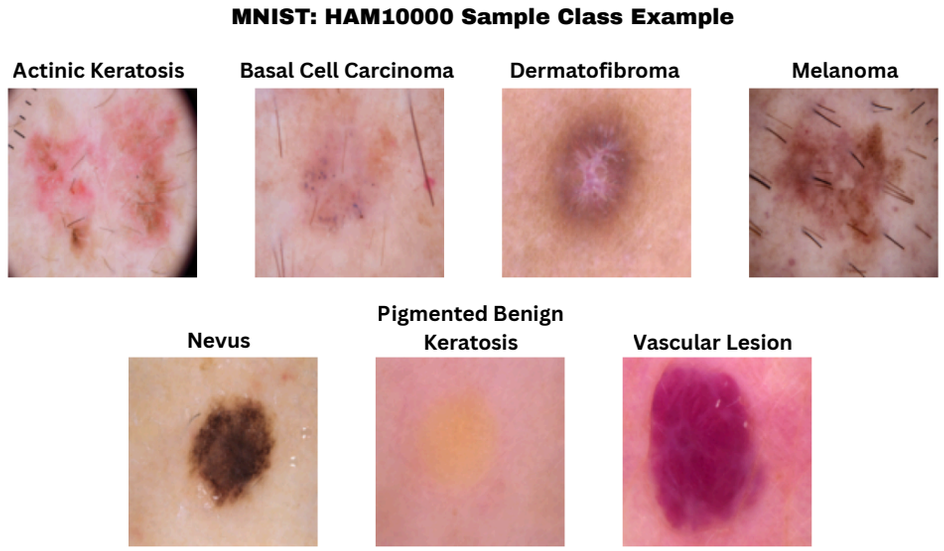


Figure 4.1: HAM10000 Sample class example

Regarding class frequency, nevus is the most prevalent class with 6,705 pictures, followed by melanoma with 1,113 pictures and pigmented benign keratosis with 1,099 pictures. The other classes are basal cell carcinoma (514 pictures), actinic keratosis (327 pictures), vascular lesions (142 pictures) and dermatofibroma (115 pictures). The imbalance in class frequencies corresponds to actual diagnosis frequency in the real world and will challenge model generalization.

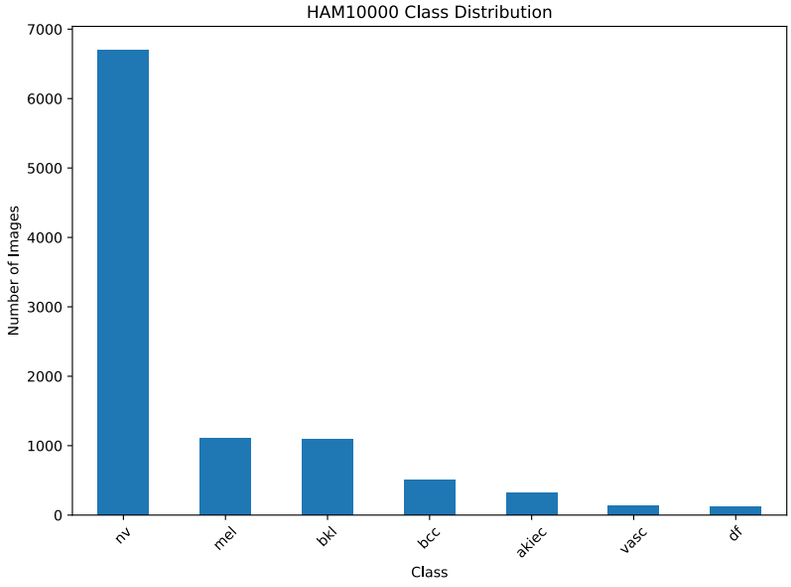


Figure 4.2: HAM10000 Class Distribution

4.1.2 PAD-UFES-20

The PAD-UFES-20[4] dataset encompasses 2,298 clinical images from 1,373 patients involving 1,641 different lesions. As a non-dermatoscopic dataset, it was created from images produced by standard smartphone cameras in the real world within a clinical setting, which makes this dataset even more attractive for model development for low-resource settings or general practice development. Approximately 58% of cases are biopsy-proven with the others diagnosed through clinical assessment. Furthermore, the dataset comes equipped with extensive metadata for each case, 26 clinical characteristics in total, going beyond image recognition to include patient age, sex, lesion size, and location, as well as smoking history and family history of skin cancer. Thus, PAD-UFES-20 is appropriate for multimodal applications that rely upon both visual and clinical data.

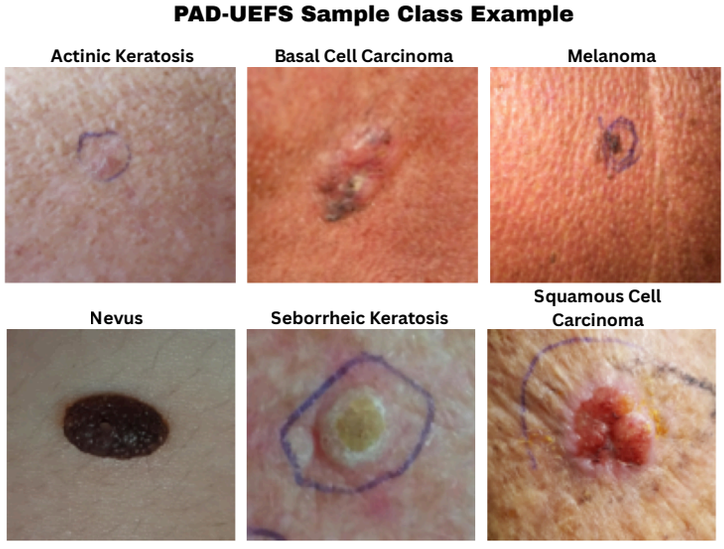


Figure 4.3: PAD-UFES-20 sample class example

In terms of class distribution, basal cell carcinoma and actinic keratosis are the most frequent with 845 images and 730 images, respectively. Nevus includes 244 images, while seborrheic keratosis has 235, squamous cell carcinoma has 192 images, and melanoma is the most infrequent with only 52 images. The significant class imbalance and lack of melanoma related images will complicate training effective classification models that are balanced and reliable.

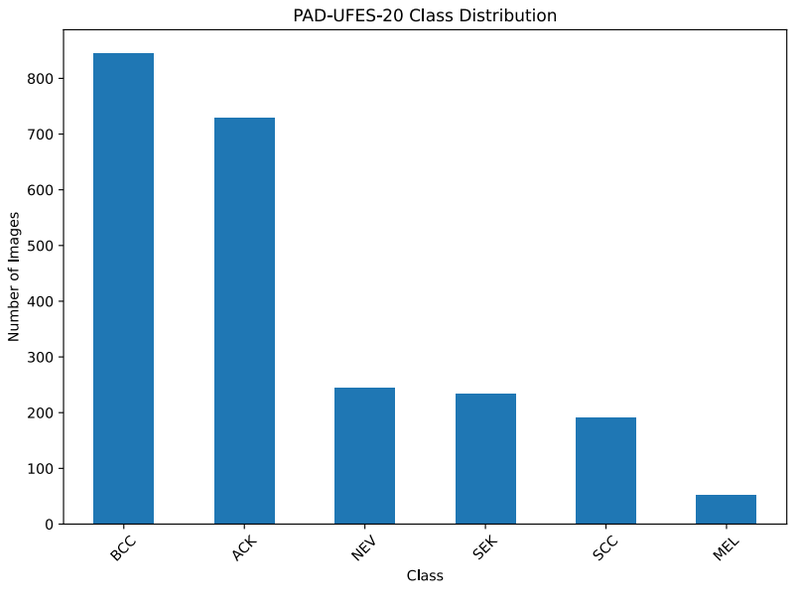


Figure 4.4: PAD-UFES-20 Class Distribution

4.1.3 ISIC 9-Class Dataset

The ISIC[3] dataset contains 2,357 dermatoscopic images arranged across nine classes of diagnosis, all curated and annotated according to ISIC classification. This dataset is commonly used as a benchmark for skin lesion classification tasks, particularly within the realm of automated melanoma diagnosis efforts. However, whereas HAM10000[5] and PAD-UFES-20[4] did provide some patient metadata, this dataset merely supplies the dermatoscopic images and their respective class labels.

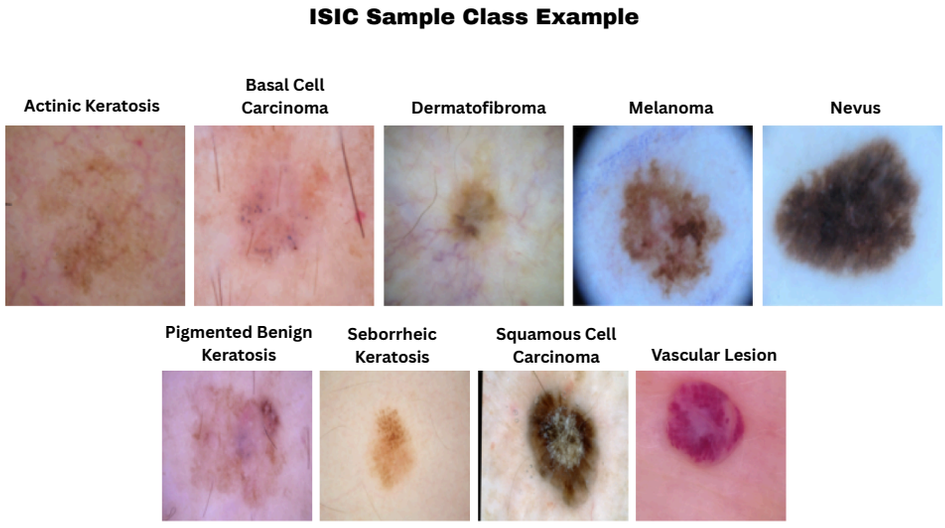


Figure 4.5: ISIC sample class example

Regarding distribution, the classes of pigmented benign keratosis and melanoma have the greatest number of images with 478 and 454 images as a second most common diagnosis, respectively. The other most common classes include basal cell

carcinoma (392) and nevus (373). The remaining classes include squamous cell carcinoma (197), actinic keratosis (130), vascular lesions (142), dermatofibroma (111) and the least common with 80 images, seborrheic keratosis. The dataset is fairly balanced, although there are a few more instances of melanoma and benign keratosis, respectively, which is important to note when training the model and interpreting the results.

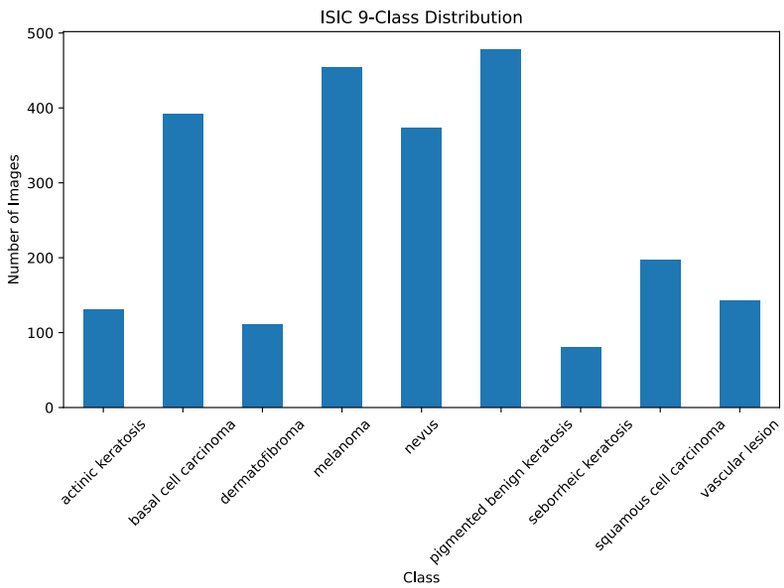


Figure 4.6: ISIC 9-Class Distribution

4.2 Data Preprocessing

Initially, all datasets were sorted into class folders to allow for auto labeling through PyTorch's ImageFolder. Subsequently, images were added to respective subfolders to manually create a training/testing image split. This image split was consistent across experiments for integrity and to avoid any risk of overfitting. Once the datasets were configured similarly, an expected preprocessing pipeline followed, where all images were resized, turned into tensors and normalized. Normalization was dataset specific, while resizing and turning into tensors were the same. Normalization parameters differed depending on the dataset due to model and data requirements.

Table 4.2: Class distribution across datasets (Train/Test splits)

Class Name	**HAM10000**		**PAD-UFES-20**		**ISIC**	
	Train	Test	Train	Test	Train	Test
Actinic keratosis	261	66	584	146	104	26
Basal cell carcinoma	411	103	676	169	314	78
Dermatofibroma	92	23	—	—	89	22
Melanoma	890	223	41	11	363	91
Nevus	5364	1341	195	49	298	75
Pigmented benign keratosis	879	220	—	—	382	96
Seborrheic keratosis	—	—	188	47	64	16
Squamous cell carcinoma	—	—	153	39	158	39
Vascular lesion	113	29	—	—	114	28

4.2.1 HAM10000

Images taken from the HAM10000[5] dataset were resized to 224 × 224 pixels to fit the anticipated input dimensions of the RegNetY architecture. Upon resizing, images were converted into tensors through the PyTorch transformation ToTensor(). Thereafter, data was normalized for all RGB channels by a (0.5), (0.5), (0.5) mean and standard deviation, effectively scaling image pixel values to a range of [-1, 1], where [-1] represents black and [1] represents white; this is the expected standards for inputs/outputs when training models from scratch. These transformations were accomplished on the fly during data loading through the torchvision.transforms library.

4.2.2 PAD-UFES-20

In the case of PAD-UFES-20[4], all other preprocessing steps matched those of HAM10000[5]; the images were resized to 224 × 224 pixels, transformed into tensors, and normalized with a mean and standard deviation of 0.5 for all three RGB channels. PAD-UFES-20 also provides more clinical metadata about patients, including their ages, locations of lesions, and risk factors, but these fields were not useful in model training, so only the image data was utilized. No preprocessing or evaluation fields were included; as they would have been from a separate field in HAM10000, to keep things standardized across the two datasets and ensure images alone could be classified.

4.2.3 ISIC

Preprocessing for the ISIC[3] dataset was adapted to align with conventions typically used for pretrained models. Like the other datasets, images were resized to 224 × 224 pixels and converted into tensors. However, normalization was performed using ImageNet statistics; specifically, a mean of [0.485, 0.456, 0.406] and a standard deviation of [0.229, 0.224, 0.225] across the RGB channels. This approach enabled compatibility with models like VGG16, which were originally trained on the ImageNet dataset. Unlike PAD-UFES-20[4], ISIC does not provide any clinical metadata; only the image data was processed and used during training.

4.3 Dataset for Cross Evaluation

To assess how well these models generalize to other datasets, a subset of common diagnostic classes was selected for cross-dataset evaluation. Out of all classes used in the three datasets, four classes were found to be common across the three datasets used: HAM10000[5], ISIC[3], PAD-UFES-20[4]: Actinic Keratosis, Basal Cell Carcinoma, Melanoma, and Nevus. For the cross-evaluation experiments, models were trained on images per dataset from only the common classes (i.e., class labels were preserved across datasets), and only trained on that class to ensure no performance was degraded due to class label misalignment and subsequent inclusion of unrelated images.
The number of training images per class per dataset is detailed below:

Table 4.3: Class distribution across datasets (Train only)

Class Name	**HAM10000**	**ISIC**	**PAD-UFES-20**
Actinic Keratosis	261	114	584
Basal Cell Carcinoma	411	376	676
Melanoma	890	438	41
Nevus	5364	357	195

This means that trained models could be fairly assessed across datasets since the same classes were selected and the only thing changing was the trained model output and the dataset itself, preventing excess variability from class imbalances on a given dataset or specific definitions of each label.

Chapter 5

Methodology

As previously discussed, this study follows a supervised learning approach for skin lesion classification using three publicly available datasets: HAM10000[5], PAD-UFES-20[4], and the ISIC[3] 9-Class dataset. Each dataset was paired with a specific convolutional neural network model selected based on the dataset size, structure, and image characteristics. The complete pipeline consists of image preprocessing, dataset splitting, model selection, training, and evaluation.
After preprocessing (see Section 3.2), each dataset was manually divided into training and testing folders, organized by class label. The images were then passed through a PyTorch ImageFolder loader, applying resizing and normalization during loading. A batch size of 32 was used across all experiments. The models were trained using the Adam optimizer with a learning rate of 0.0001, and the loss function was cross-entropy loss, suitable for multi-class classification. Each model was trained for 20 epochs on GPU hardware using PyTorch. Evaluation metrics included overall accuracy, per class precision, recall, F1 score, and confusion matrix analysis.

5.1 RegNetY-32GF

For the HAM10000[5] dataset, a high capacity convolutional neural network was required to meet the extensive dataset size and various lesions types. Therefore, the selected model was RegNetY-32GF, a scalable architecture derived from the RegNet family established by Facebook AI Research, renowned for its grouped bottlenecks and SE (squeeze-and-excitation) blocks.
The Implementation was employed using PyTorch's RegNetY-32GF model with ImageNet pretrained weights (weights='DEFAULT'), replacing the final fully connected layer with a linear layer compatible with the seven output classes from HAM10000[5]. Images were transformed to 224×224 pixels, resized to tensors, normalized with a mean and standard deviation of 0.5 for all three RGB channels.
The implementation loaded the dataset through ImageFolder, trained the model with the Adam optimizer at a learning rate of 0.0001, and applied cross-entropy loss. A batch size of 32 was utilized for both training and testing. The model was trained for 20 epochs, with results measured at each interval against the test data using accuracy results. Upon completion of the last epoch, a classification report and confusion matrix were generated for per-class performance measurement. The model weights were saved once trained, along with all reported results and measurements.

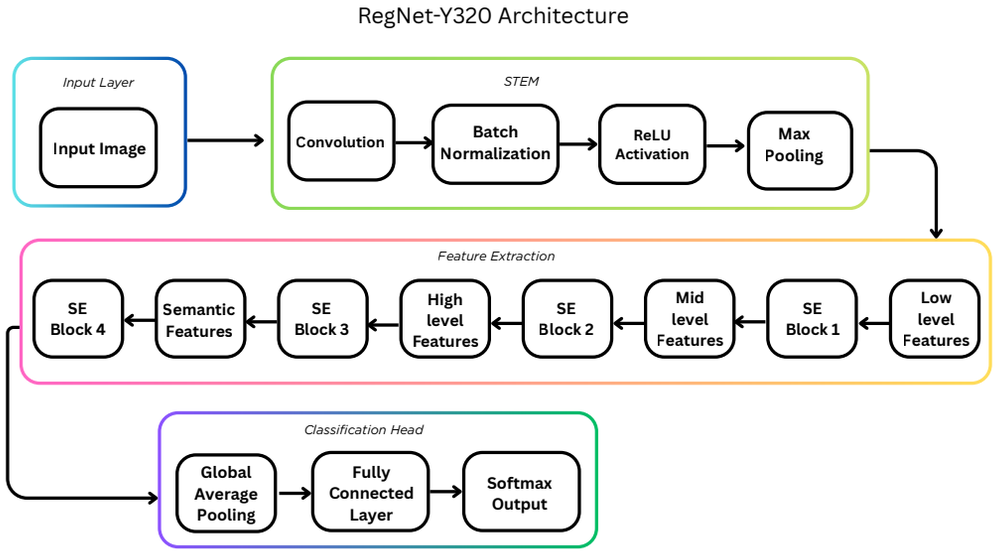


Figure 5.1: RegNetY-32GF

5.2 VGG16

The VGG16 model was applied to the ISIC[3] dataset containing 2,357 dermatoscopic images divided among nine classes. VGG16 was selected due to its efficient classification abilities relative to the complexity of the model required for such a mid-level classification project.

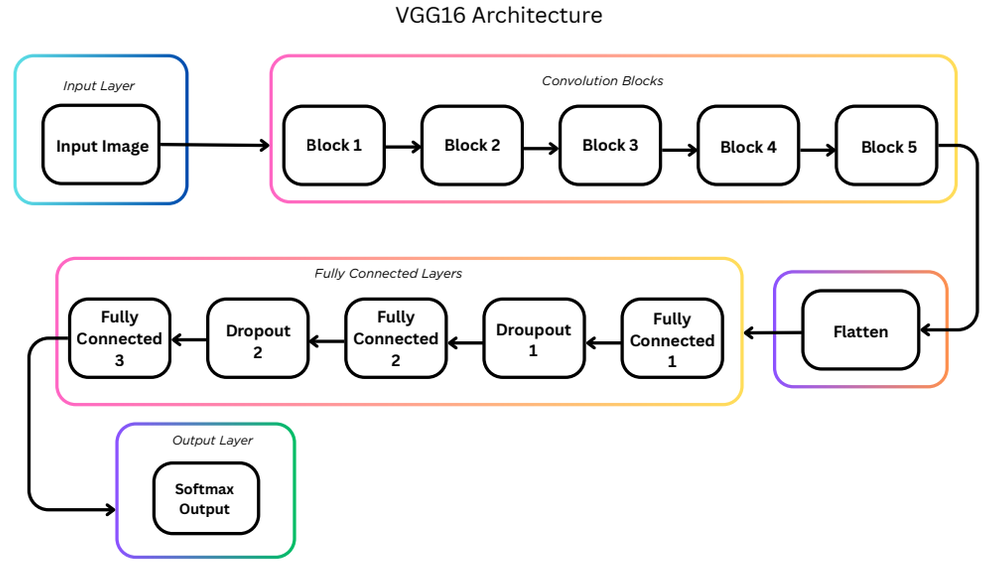


Figure 5.2: VGG16

In this application, VGG16 was trained from scratch (weights=None) without pre-trained weights, and the classifier was adjusted by changing the last dense layer to provide nine class logits. All images were resized to 224×224 for input, cast to tensors, and normalized via the ImageNet mean and standard deviation: [0.485, 0.456, 0.406] for mean and [0.229, 0.224, 0.225] for standard deviation.

The images were stored in PyTorch's ImageFolder, and a DataLoader was established with a batch size of 32 for transport. The Adam optimizer (0.0001) was used for training with cross-entropy loss as the loss function. Training occurred over 20 epochs, with the training/testing accuracy logged each epoch. After completion, the model was evaluated on the test set with a final evaluation, and confusion matrix/confusion report were generated and saved for future use.

5.3 SkinLesNet

Due to the smaller size and heterogeneous quality of the PAD-UFES-20[4] dataset, a custom lightweight CNN was created referred to as SkinLesNet[10]. This from scratch architecture was tailored to run efficiently on lower complexity medical images obtained through smartphone camera settings.
The basic architecture consisted of four convolutional layers, followed by a ReLU activation function and max pooling. There existed a dropout layer before the classifier block to avoid overfitting from the last convolutional block, which was followed by a flattening layer, a dense layer with 64 units and ReLU activation, a final dropout layer (p=0.3) and an output layer producing logits for the six diagnostic classes[13].

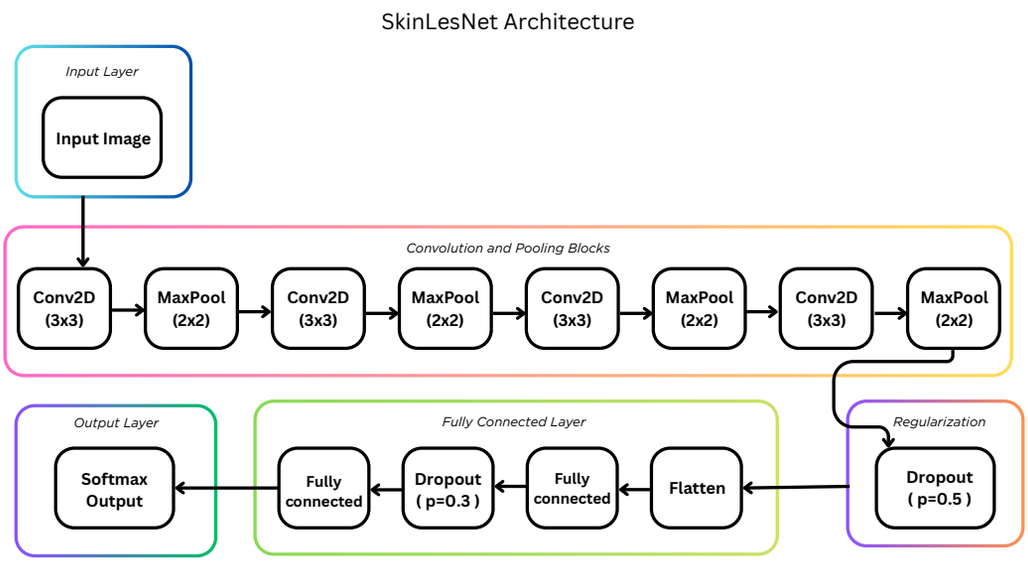


Figure 5.3: SkinLesNet

All images were adjusted to 224×224 dimensions and normalized by the mean and standard deviation of 0.5 for all channels, in line with the experimental non-ImageNet ones. The model was fitted for 20 epochs using the Adam optimizer (learning rate = 0.0001) and cross-entropy loss. A batch size of 32 was also implemented. During training, training and testing accuracy were recorded per epoch. At the end of the last epoch, a model evaluation was performed via the classification report, confusion matrix, and final test accuracy.

5.4 DermaNet Description

Clinical Context and Design Philosophy

Standard CNNs have difficulty dealing with dermoscopic images classification and require enhanced modelling. The skin cancer datasets which are popular in the skin lesion classification exhibit severe class imbalance where melanoma examples are heavily dominated by benign lesions like melanocytic nevi. In addition, the diagnostic characteristics in dermoscopy rely largely on subtle texture patterns, variations in color and detailed structures that necessitate deep feature hierarchies and attention mechanisms.

A traditional approach would either adopt generic architectures with no domain specific changes or apply aggressive data augmentation that creates unrealistic variations that could train the model to identify artifacts instead of pathology. Through a three stage hybrid architecture of transfer learning, SE block at channel level, and structured regularisation, DermaNet aims to enhance these models. The design philosophy emphasizes clinical realism. Minimal augmentation mimics photographic conditions. Focal loss manages class imbalance with no synthetic oversampling[17]. Meanwhile, the attention mechanism enables the model to prioritize diagnostically important features without treating all channels equally.

Overall Architecture Design

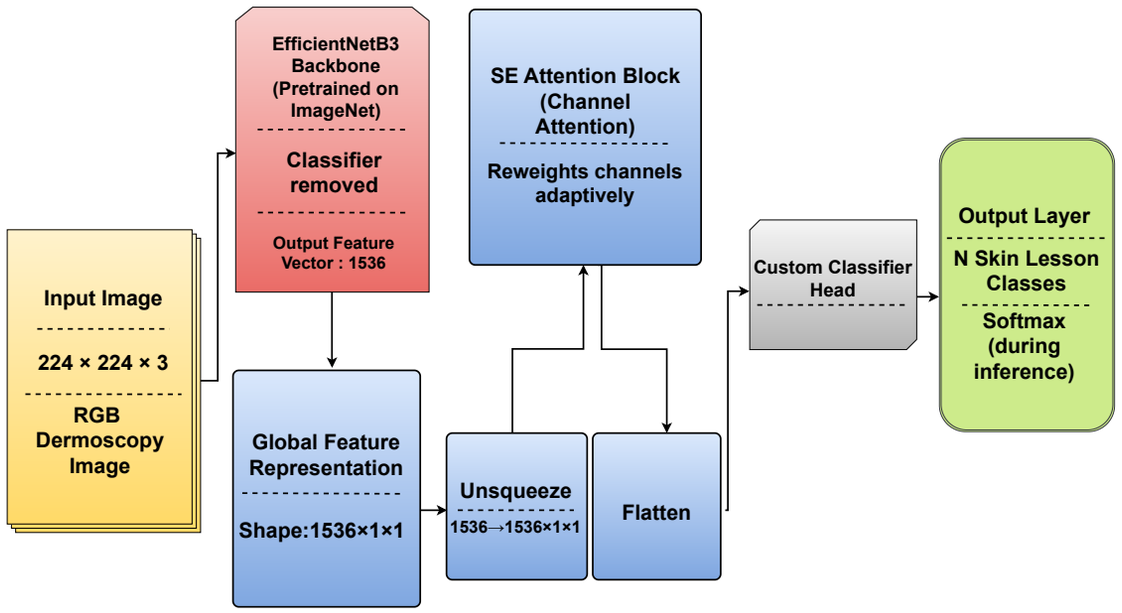


Figure 5.4: DermaNet Pipeline

Figure 5.4 illustrates the complete DermaNet pipeline, showing how a $224 \times 224 \times 3$ RGB dermoscopic image flows through the network. The architecture consists of three distinct but synergistic stages. Feature extraction via EfficientNet-B3[65], channel wise feature recalibration through Squeeze-Excitation[7], and hierarchical classification through a custom head. This separation of concerns allows each component to specialize, the backbone learns rich visual representations, the attention

module identifies which features matter most for dermoscopy, and the classifier makes the final diagnostic decision with appropriate regularization to prevent overfitting on the relatively small medical dataset.
Unlike end to end architectures that treat feature extraction and classification as a single rigid block, DermaNet's staged design enables targeted optimization. The backbone leverages ImageNet pretraining, the SE block adds domain specific adaptability, and the classifier is trained from scratch to learn task specific decision boundaries[7]. This hybrid approach consistently outperforms purely pretrained models which lack dermoscopy specific recalibration and purely custom architectures which struggle with limited medical data.

5.4.1 Feature Extraction: EfficientNet-B3 Backbone

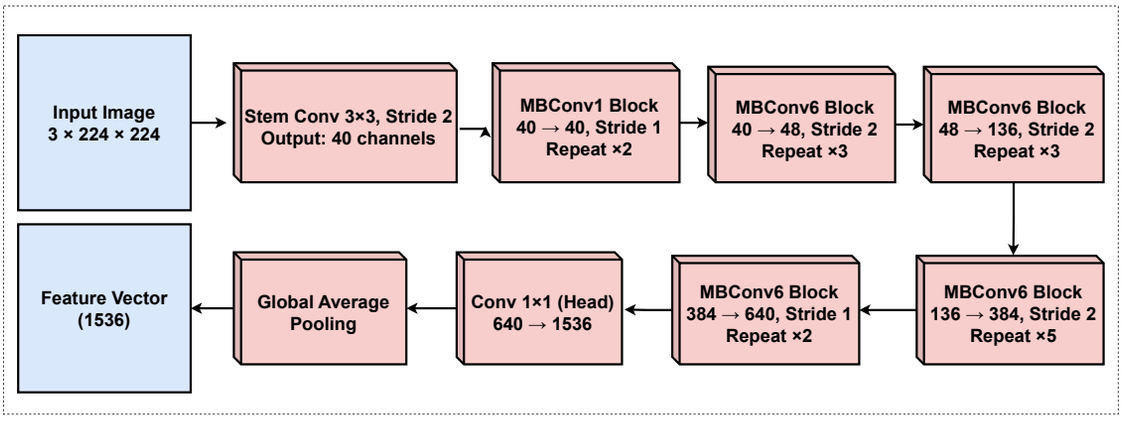


Figure 5.5: EfficientNet-B3

Due to its powerful performance and computational efficiency, EfficientNet-B3 is used as the backbone network in this work for feature extraction[8]. The architecture's performance relies on the compound scaling technique that enhances the depth, width, and input resolution of each network in a balanced way leading to improved learning of features without drastically raising model complexity.
The first layer of a network is a stem convolution layer of size 3×3 with a stride of 2. It reduces the spatial size of the input image while increasing the number of feature channels. This phase captures low level visual features like edges, color contrast, and texture patterns found in dermoscopic images. After the stem, EfficientNet-B3 consists of lots of Mobile Inverted Bottleneck Convolution (MBConv) blocks.
A number of MBConv blocks form the building blocks of the EfficientNet family of architectures. These MBConv blocks make use of depthwise separable convolutions and inverted residual connections. Thus, the network is parameter efficient but with high representational power. Both the MBConv1 and MBConv6 building blocks are being used and the expansion factor is used to control the channel growth in every building block. With the help of spatial down sampling in stride-2 MBConv blocks, hierarchical features can be extracted from shallow layers to deep layers. In the end, a 1×1 convolving head collects the features together and forms a compact high dimensional feature vector that summarizes the lesion's characteristics.

5.4.2 Squeeze-Excitation Block

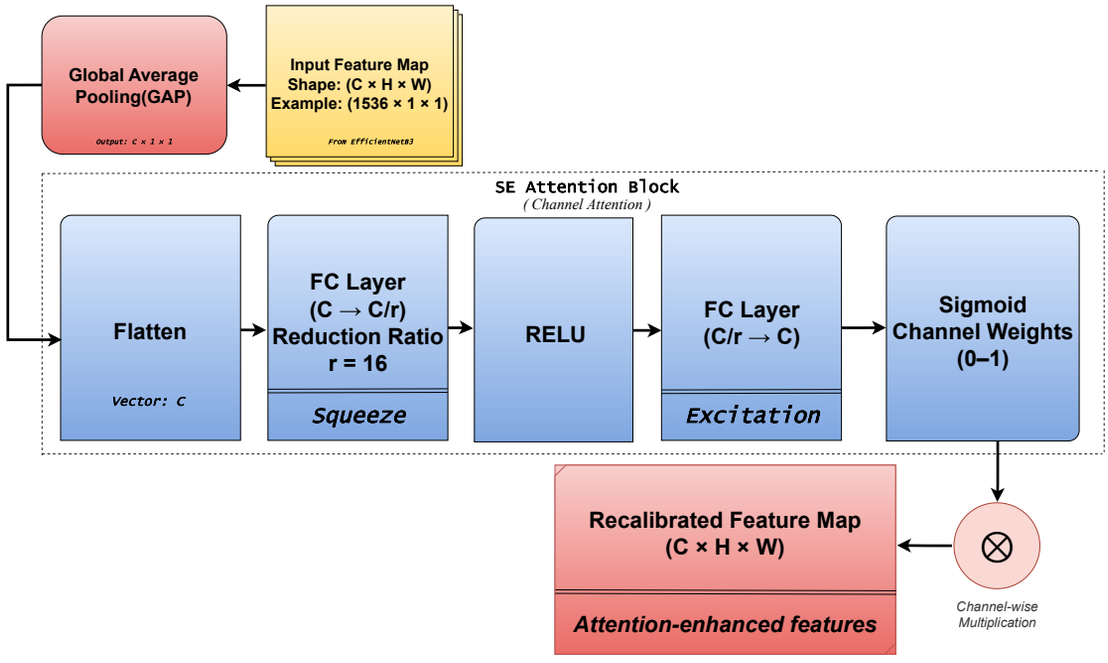


Figure 5.6: SE Block

The raw features incorporate all 1536 channels equally but dermoscopic diagnosis does not work in that way. For example, vascular patterns are much more diagnostic than other patterns which do not provide any major diagnostic value for cancer detection. SE block (Figure 5.6) addresses this by means of explicit channel recalibration[7]. Basically, the SE mechanism is based on two phases, named the squeeze phase and the excitation phase. In the squeeze phase, the spatial information of each channel is reduced to a single scalar via global average pooling, yielding a channel descriptor $\mathbf{c} \in R^{1536}$[7]. The global statistics of each channel are captured in this descriptor. High value means that the channel has strong activations all the time. Low value means that the channel has weak or very little activation.
The excitation phase then learns channel interdependencies through a bottleneck architecture:

$$s = \sigma(W_2 \cdot \mathrm{ReLU}(W_1 \cdot c))$$

$W_1 \in R^{(1536/16)\times 1536}$ reduces the dimensionality by a factor of $r = 16$, which is the reduction ratio, and $W_2 \in R^{1536\times(1536/16)}$ brings the dimensionality back to the original channel size. A smaller ratio causes the network to learn a compact and informative representation of channel relationships rather than memorizing them. The final sigmoid activation σ yields weights $\mathbf{s} \in [0,1]^{1536}$, which amplify channels perceived as useful (values close to 1) while suppressing those deemed irrelevant (values close to 0).
Essentially, the recalibrated features are computed by channel wise multiplication,

$$\tilde{\mathbf{F}} = \mathbf{s} \otimes \mathbf{F},$$

where $\mathbf{F}$ is the original 1536-dimensional feature vector. Empirically, SE blocks consistently upweight texture descriptor channels while down weighting background

noise channels, closely mirroring the process a dermatologist follows when examining a lesion[7].
The reduction ratio was set to $r = 16$ because smaller ratios (e.g., $r = 8$) introduce a significant increase in parameters without corresponding performance gains, as the model already captures the required channel relationships. Conversely, larger ratios such as $r = 32$ create a severe bottleneck that discards useful information. Thus, choosing $r = 16$ strikes a favorable balance between expressiveness and efficiency.

5.4.3 Classification Head: Hierarchical Decision-Making

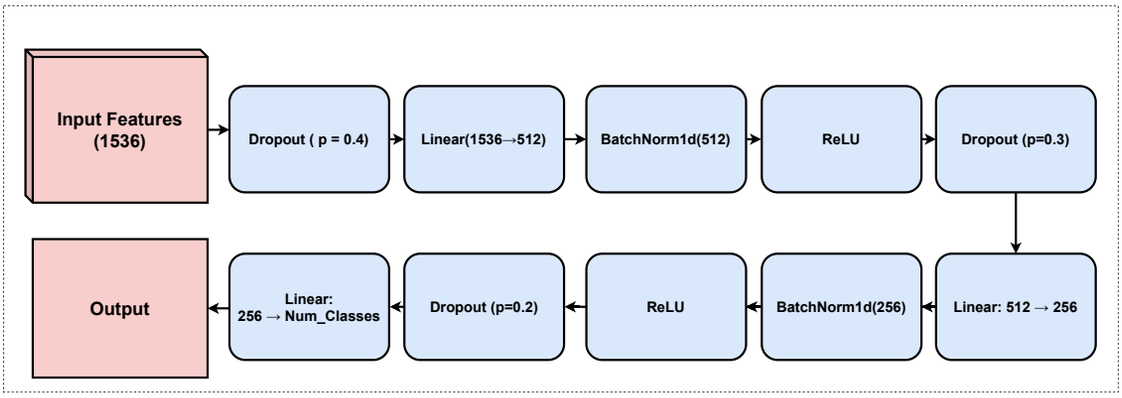


Figure 5.7: Classifier Head

The attention modified 1536-d features are fed to a custom classifier that prevents overfitting on the relatively small datasets. We begin with an overview of the architecture of classifier shown in Figure 5.7 which uses progressive dimensionality reduction and aggressive regularization.
The first layer applies dropout with $p = 0.4$, resulting in 40% of its features being zeroed randomly during training. We risk overfitting medical datasets due to their intrinsic sample diversity. Furthermore, we apply strong regularization at the input layer so that the classifier does not memorize correlations in feature space. After dropout, the projection from 1536-dimensions to 512-dimensions takes place using a linear layer. Following which, activation normalization is achieved using BatchNorm1d which facilitates gradient flow. ReLU uses non linear activation that helps the neural networks learn complex functions.
The second stage repeats Dropout $p = 0.3 \rightarrow$ Linear $512 \rightarrow 256 \rightarrow$ BatchNorm $\rightarrow$ ReLU, with a somewhat less aggressive regularization scheme. We use a decreasing dropout schedule ($0.4 \rightarrow 0.3 \rightarrow 0.2$) to reflect the intuition that later layers learn a lot of task specific patterns that should be preserved and not aggressively randomized. In the end, the output layer produces logits for each skin lesion category (Dropout $p = 0.2 \rightarrow$ Linear $256 \rightarrow$ `num_classes`).
The information is put into the system, which leads to the generation of a deep neural problem in dermatology. This multi layered incorporates notations. 512-d performs the general evaluation like melanocytic and non-melanocytic. 256-d classification from melanoma or nevus is performed. And in the end through the last layer classification occurs. Using a single layer classifier ($1536 \rightarrow$ `num_classes`) as a baseline is weak as single layer classifiers cannot create complex decision boundaries.

Using too deep of a classifier such as five or more layers incurs overfitting despite dropout. Thus, through empirical results, we finally take a three layer classifier, which works best for dermoscopy classification tasks.

Training Strategy and Clinical Realism

Apart from architectural decisions, DermaNet's formation protocol mirrors clinical considerations. Rather than using cross entropy we utilize focal loss. The formula is:

$$-\alpha(1-p_t)^\gamma \log(p_t).$$

The predicted probability for the true class of a model can be denoted by p_t whereas $\alpha = 1$ adjusts positive/negative instances and $\gamma = 2$ down weights easier instances. The loss $(1-p_t)^\gamma$ of this term is important. Hereby, when the model correctly predicts a lesion confidently ($p_t \to 1$), the loss effectively goes to zero. As a result, training is allowed to focus on hard, ambiguous cases. This tackles class imbalance naturally, without synthetic data and class weights needing to be specified.
Augmenting the data was kept fairly simple. We applied a horizontal flip, as dermoscopy images do not have any canonical orientation, $\pm 10°$ rotation, to simulate slight tilts in the camera. Finally, we did a minor colour jitter where brightness/contrast was $\pm 10\%$ to simulate a variation in lighting. We avoid strongly applied augmentations such as vertical flips (uncommon in clinical), large rotations (unrealistic angles), or elastic deformations (introduce artifacts). This approach stops the model from learning to classify any augmentation artifacts instead of the real pathology as it should.
We built PyTorch model with weighted random sampling for generating mini batches even when we have a class imbalance in the training set. Then we have also used AdamW optimizer with weight decay (1e−5), which implies a soft L_2 regularization on the weights apart from dropout. Initially not too impactful, the learning rate (1e−4) halves whenever the validation accuracy plateaus, allowing for fine tuning without losing pretrained features.

5.5 Skin Cancer Detection in Federated Learning

5.5.1 FedSE

DermaNet works very well in centralized settings, but deploying medical AI systems in the real world comes with a fundamental limitation. Hospitals and medical institutions cannot share raw patient data without consent due to privacy regulations such as HIPAA and GDPR[11]. Federated Learning (FL) is a promising alternative, allowing several institutions to collaboratively train a shared model without sharing their local datasets. But, translating architectures from the centralized setting to the federated one is no easy accomplishment for deep learning, especially if the architecture uses any stochastic regularization like dropout[13]. FedSE is our baseline federated implementation that operates in the standard Federated Averaging (FedAvg). It sets up the basic problem that motivates the design of mask aware aggregation algorithms introduced in this work.

Federated Architecture Overview

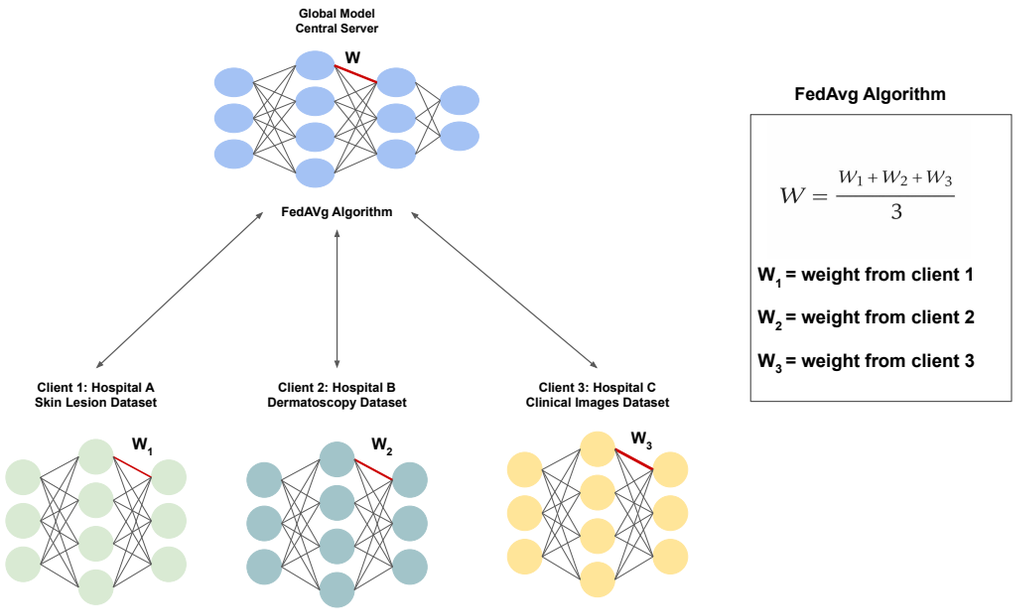


Figure 5.8: Federated Learning

As shown in Figure 5.8, three independent clients representing hospitals with heterogeneous dermoscopy datasets collaboratively train a model with FedSE topology. Each client corresponds to an actual entity. Client 1 uses the HAM10000[5] dataset, which are dermoscopy images from a well standardized dermatology clinic and acquired with up to ten different dermoscopy devices. Client 2 uses images from the ISIC[3] dataset, which are from dermatology images acquired with a variety of devices and conditions from different institutions. Client 3 uses the PAD-UFES-20[4] dataset, which are clinical photographs acquired from different hospitals with different devices and protocols and with varying levels of quality. The diversity mirrors real world federated implementations in which hospitals deploy different imaging tools and patient groups and diagnosis protocols. The central server initializes the DermaNet feature extractor trained globally, $\mathbf{w}_{\text{global}}$, and sends it to all clients. Then the federated training cycle takes place for each client. Every client takes these shared weights, incorporates them into their local model architecture, and trains for numerous epochs on their personal dataset. After the completion of local training, the clients send only the updated weights of feature extractor $(\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3)$ to the server. Raw patient images do not leave the hospital. The server aggregates these weight updates using the FedAvg algorithm, producing a new global model $\mathbf{w}_{\text{global}}$, which is then sent out to the devices for the next round of training. Observe that, although the backbone which extracts features is shared and jointly trained, each of the clients maintains its classification head. Likewise, this classification head is tailored to the specific classes and labels a client has. Thus, the output spaces of the various clients can be different despite the models being jointly trained.

Client Side Training Protocol

Every client employs a duplicate of the DermaNet architecture previously described, including the complete pipeline from EfficientNet-B3 backbone through SE to the classification head. The key difference in FedSE is how the SE block behaves in federated training. The stochastic dropout layer ($p = 0.3$) within the SE bottleneck acts independently at each client with randomly generated dropout that change at each forward pass. What this means is that, for example, at Client 1, certain neurons of the fully connected layers in the SE block may be active while these same neurons are dropped at Client 2. Thus, not all clients will have the same network structure in the federation.

Local optimization uses the same principled protocol as centralized DermaNet. Each client is assigned the AdamW optimizer with a conservative learning rate of 1×10^{-4} and a weight decay of 1×10^{-5}. This is designed to avoid overfitting on relatively small medical datasets[66]. The Focal Loss function solves problems of class imbalance in the clients local data distribution[17]. We set $\alpha = 1$ and $\gamma = 2$ so that our model will focus it's learning effort on difficult, minority class examples and not continuously optimize lesions that have already been well classified. For each round of global, training is done through four local epochs with a batch size of 32 along with a weighted random sampling. This, as a result, guarantees that mini batches of unequal class frequencies remain balanced. The amount of data augmentation that was performed was limited and realistic in a clinical way, such as horizontal flips, small rotations ($\pm 10°$), and small color jittering, to not teach the model to recognize unrealistic artifacts rather than real pathological features.

Server Side Aggregation

Once all clients finish their local training, the server collects all the updated feature extractor weights and performs the standard Federated Averaging. The aggregation equation is simple.

$$w_{\text{global}} = \sum_{k=1}^{K} \frac{n_k}{N} w_k$$

In the expression, $\mathbf{w}_k$ refers to the weight tensor at client k, n_k refers to the number of training samples at client k, and N is the overall number of samples across clients. In this scheme, larger clients (having more training data) play a proportionately greater role in the global model which makes intuitive sense. For instance, a hospital with 5000 dermoscopy images ought to contribute more to the shared representation than one with 500 images.

Only feature extractor parameters will be shared in the aggregations i.e. EfficientNet-B3 backbone and SE block. Classification heads vary from client to client and remain local as different clients may have a different number of output classes (HAM10000 has 7 lesion types but ISIC may only focus on binary melanoma/nonmelonoma classification). This is a fundamental characteristic of the federated framework. The backbone learns universal dermoscopic feature representations while heads maintain flexibility for institution specific diagnostic tasks. After the aggregation process has

finished, the server propagates $\mathbf{w}_{\text{global}}$ to all clients and begins the subsequent global round.

The Zero Dilution Problem

The stochastic dropout feature in SE block during the local training would lead to standard FedAvg aggregation to a slight but damaging problem[13]. We can visualize this problem by using the excitation layer from the SE block in Figure 5.8. Imagine one training round where in the SE fully connected layer, the dropout mask of Client 1 randomly shows mask = 1 for this weight, meaning Client 1 uses it. Client 1 successfully trains this weight and produces a weight update of 0.82. Thus, it is a strong and learned representation of this channel. Client 3 keeps this neuron on. It learns a different but meaningful weight of 0.31, a reflection of its data. Nonetheless, stochastic dropout of Client 2 turns off the neuron (mask = 0) and it gets the weight exactly 0.00 in this round.
The standard FedAvg does not work here. When the server aggregates these three updates, it computes

$$(0.82 + 0.00 + 0.31)/3 = 0.376$$

and takes the zero from Client 2 to mean like a learned weight for this channel "this is not important". But this zero is not structural but semantic. That is, Client 2 does not mean this channel is not important. Client 2 just turned off this neuron for regularization. FedAvg scales down the global representation by averaging this structural zero with learned weights. The aggregated weight of 0.376 is 54% smaller than Client 1's learned value and represents a corrupted compromise between genuine learning and arbitrary dropout noise.
This dilution worsens dramatically throughout the network. The SE mechanism depends on optimally assigned channel weights to adjust features once weights are assigned wrongly and diluted from round to round, the SE module loses the ability to enhance diagnostic relevance and reduces noise. The problem is not limited to the SE block. Dropout layers in the classification head behave the same way, and as different clients drop different neurons at different times, the inconsistency builds up. The early layers tend to suffer the least while the SE block and classifier head suffer the most. As the global rounds progress, the feature representations become weaker and less effective, convergence slows down drastically, and final accuracy drops compared to using a centralized version of the same architecture.

Training Configuration

The FedSE training runs for 5 global rounds, with each round having 4 local training epochs at each client. Communications needs gets balanced with speed of convergence to global model by way of using a 5×4 schedule (20 total local epochs per client distributed across global rounds). More local epochs per round means less expensive communications however greater risk of client drift. A situation where the local models drift a little too far from the global model before being synchronized.
The three clients show significant heterogeneity that mirrors real federated deployment. Client HAM (Hospital A) is trained on approximately 10,000 dermoscopy images from the HAM10000[5] dataset, with a balanced distribution of seven lesion categories including common nevi and rare melanomas. The client ISIC (Hospital B)

uses approaximately 2500 images sourced from the ISIC[3] dataset. The classes have different distributions from ISIC dataset with a bigger proportion of melanoma lesions and imaging variations from multiple institutions contributing to the ISIC dataset. The client PAD (Hospital C) is trained on a total of 2,300 images from PAD-UFES-20[4] which consists of clinical images instead of standardized dermoscopy resulting in domain shift in image quality, resolution and lighting conditions. The purpose of this heterogeneity is to assess whether the proposed federated model is able to learn robust features.
The data preprocessing is similar to the centralized DermaNet, which finds the 224×224 RGB images normalized with the mean: $[0.485, 0.456, 0.406]$ and standard deviation: $[0.229, 0.224, 0.225]$ the ImageNet statistics. Avoid artifacts through muted and realistic augmentation. Aggressive transformation is avoided. After each global round, we perform evaluation using each client's held out test set. The metrics include per client accuracy, weighted F1score, and overall performance on all clients. Additionally, we keep track of training loss and accuracy per round to detect convergence behavior and signs of zero dilution degradation.

Baseline Characteristics and Limitations

FedSE deliberately uses naive aggregation, exposing the zero dilution problem faced by standard federated learning when combined with stochastic regularization. This baseline does not represent optimal federated performance and it sets a lower bound that our mask aware aggregation needs to exceed. The system is expected to be slower in convergence as compared to the centralized DermaNet . The final performance is degraded especially on harder minority classes due to the diluted SE weights failing to emphasize rare diagnostic features. There is a high variance across clients since some clients may suffer more than others depending on how the stochastic dropout masks align. The SE block shows particularly severe weight corruption. Specifically, channel recalibration weights that should range from 0.1 to 0.9 are forced to compress to 0.4–0.6, after aggregation. Thus, the SE mechanism's selectivity is effectively neutralized.
The baseline represents the main comparison for all the federated methods presented in this thesis. By measuring the amount of standard FedAvg performance lost to zero dilution, we can closely evaluate whether these proposals via fixed dropout masks, mask aware aggregation, and structured dropouts are correcting for the problem and not just altering a hyperparameter. The FedSE findings provide the answer to the question of what happens if you naively federate a dropout heavy architecture, which motivates advanced aggregation strategies that respect the structural nature of the zeros induced by regularization.

5.5.2 FedMD: Mask Aware Federated Learning for Dropout Enhanced Models

Motivation: From Zero-Dilution to Mask-Awareness

FedSE has exposed a fundamental incompatibility between stochastic dropout and federated averaging, the randomly generated dropout masks create structural zeros that standard FedAvg, gives similar importance as the highly representative learning

client. systemically corrupting the global model. When the server receives weight tensors containing zeros, it needs to differentiate between "learned zero' and "dropped zero." FedMD resolves this through deterministic mask generation, ensuring each client's dropout mask is fixed randomly and reproducible and mask aware aggregation enabling the server to exclude structural zeros from averaging. The key benefit of this is that, as the dropout masks are known and deterministic, the server can use this info to better aggregate and update the model rather than adding noise to the system.

Deterministic Mask Generation

FedMD generates a fixed dropout mask for each client's SE attention layers at initialization, before the training begins. it is seed based. A hash function gets a unique integer seed from the client identifier (e.g., HAM[5], ISIC[3], PAD[4]), which starts a random number generator. Then, the seed generator produces a binary mask for each SE FC layer parameter, each element set to 1 (active) with probability 0.7 or 0 (dropped) with probability of 0.3. These masks are stored permanently, without changing throughout training, as a result, the same neurons remain dropped at every iteration, epoch, and global round.
This approach provides two critical properties, first, uniqueness across clients as different clients generate different mask pattern, ensuring diverse parameter subsets are collectively trained rather than all clients repeatedly training identical neurons. Second, fixedness within clients, creating consistent network topology. And the 30% dropout, keeps the neuron active approximately 70% per client. As a result, the model can perform mask aware aggregation.

Mask Aware Aggregation Algorithm

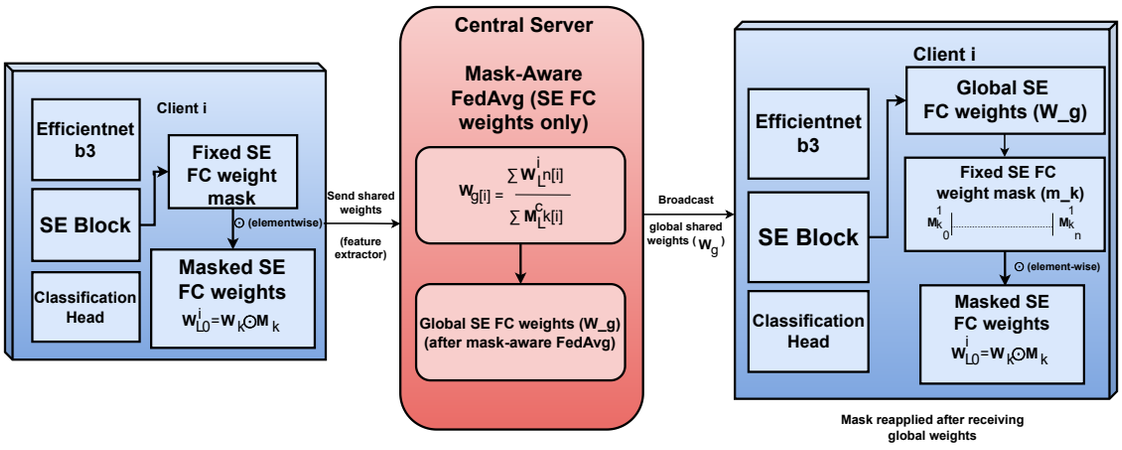


Figure 5.9: FedMD Core Mechanism

Figure 5.9 illustrates the FedMD workflow, with mask aware aggregation formula mentioned in the central server panel. unlike standard FedAvg's simple weighted average

$$w_{\text{global}} = \sum_k \frac{n_k}{N} w_k$$

, FedMD employs element wise mask conditioned normalization. Here, for each parameter position [i] in SE FC weight matrices, the server computes

$$w_g[i] = \frac{\sum_{k=1}^{K} M_k[i] \cdot w_k[i]}{\sum_{k=1}^{K} M_k[i]}$$

, where $M_k[i] \in \{0, 1\}$ is client K's binary mask value at position [i]. The denominator counts how many clients had this parameter active, while the numerator accumulates contributions only from those clients.

To visualize this mechanism, consider a specific channel attention weight in the SE block's excitation layer. Three clients send updates from this parameter. Client HAM has trained it actively and transmits $w_{\text{HAM}} = 0.73$ (with mask $M_{\text{HAM}} = 1$ indicating this neuron active), Client ISIC's deterministic mask permanently dropped this neuron so it ends up sending $w_{\text{ISIC}} = 0.00$ ($M_{\text{ISIC}} = 0$), while client PAD also trained it and contributes with $w_{\text{PAD}} = 0.91$ ($M_{\text{PAD}} = 1$). Under standard FedAvg, the server computes

$$\frac{0.73 + 0.00 + 0.91}{3} = 0.547$$

, clearly diluting two genuine two genuine learned values by giving similar importance to all weights. The result is 25% lower than it should be. Meanwhile, FedMD mask aware formula is

$$\frac{0.73 \times 1 + 0.00 \times 0 + 0.91 \times 1}{1 + 0 + 1} = \frac{0.73 + 0.91}{2} = 0.82$$

The critical difference is the denominator here, rather then dividing by 3 (total clients, giving similar importance), FedMD divides by $\sum_k M_k = 2$ (active clients only), effectively removing dropped parameters' aggregation. The output successfully represents the average learned from HAM and PAD, the two clients with genuine learned and optimized parameter. This recovery of diluted value demonstrates how mask awareness prevents aggregation noise, ensuring the global model getting authentic learned representations rather than corrupted average contaminated by regularization artifacts.

Selective Application Strategy

Our mask aware aggregation applies selectively across the architecture. As show in Figure 5.9, only SE attention FC layer weights undergo mask aware aggregation, here we restrict structural zeros' influence. The efficientNetB3 backbone uses standard FedAvg. Also classification heads remain local, but never aggregate, since our model has dynamic option for clients to have different output class clients. The parameters where zero dilution occurs, and to avoid unnecessary overhead the selective strategy is implemented there. It's implementation is straightforward. The server checks each parameter name, if it belongs to SE FC layer apply mask aware formula, else do standard FedAvg. This way, computational cost remains minimal, and adding negligible overhead compared to communication and training costs.

Parameter Coverage Analysis

As our mask is deterministic and reproducible, it creates heterogeneous but robust parameter coverage. Then, with 30% dropout, each client trains approximately 70% of SE FC parameters while dropping 30%. The clients with their independent masks, the probability that any parameter is dropped by all three simultaneously is $(0.3)^3 = 0.027$, meaning roughly 97% of parameters receive training from at least one client. Also, the expected distribution is, approximately 34% of parameters trained by all three clients , 44% by exactly two clients , 19% by single clients , and less than 3% by zero clients.
This distribution provides both robustness and diversity. Parameters that are trained by multiple clients receive averaged knowledge from different clients, providing generalization. Single client parameters learn very institution specific pattern, for example, if only HAM trains on a particular channel weight, it might emphasize standardized dermoscopy features versus clinical photography. The small fraction of untrained parameters retains pretrained values, which provides reasonable initializations for low level visual features. Thus, with three clients 30% dropout, we achieve near complete coverage and sufficient diversity to prevent mode collapse.

Client side Training with Mask Enforcement

In figure 5.9 the left panel shows the client side workflow where the "fixed SE FC weight mask" remains constant throughout training. The masking occurs at some critical points. First, during backward pass after gradient computation, when FedMD performs the element wise multiplication between gradients and the fixed mask, zeroing gradients from dropped neurons while the optimizer receives no update signal for these parameters. Second, immediately after the optimizer updates weights, FedMD again multiplies the weight tensor by the mask, correcting for any numerical drift from momentum or floating point errors, ensuring dropped weights remain exactly zero.
Here, when the server broadcasts the aggregated global model, this model contains contributions from other clients' neurons that might occupy positions where the receiving client may dropped neurons. And without remasking it would violate the fixed mask assumption. Therefore, upon receiving global weights, each client performs $w_{\text{local}} = w_{\text{global}} \odot M_k$, where $\odot$ denotes element wise multiplication. This ensures neurons dropped in the client's mask remain zero, preserving topology consistency across local iterations, global rounds, and model synchronizations.

Server side Aggregation Process

The server panel in figure 5.9 shows the aggregation workflow. After the clients send their feature weights to the server, the server does the element wise aggregation of SE FC parameters. Then for each element $[i]$, the server takes the active client count $\sum M_k[i]$, ranging from 0 (no training) to 3 (all clients trained it). And for the numerator $\sum(w_k[i] \times n[i])$ generates the weight contribution where $n[i]$ represents each client's proportional weight. The division

$$w_g[i] = \frac{\sum(w_k[i] \times n[i])}{\sum(M_k[i] \times n[i])}$$

produces the aggregated weight. The dynamics of this formula is that it adjusts based on active clients. After aggregating SE FC parameters through mask aware processing and other layers through standard FedAvg, the server broadcasts the complete global model back to clients. Clients fix their weight by reapplying the fixed masks, and commence to the next training round.

Architectural Integration add Implementation

While FedMD adds minimal complexity for the numerator and denominator of aggregation formula, it maintains complete architectural compatibility with DermaNet attention. Also, there are no modification needed for the neural network structure. The EfficientNetB3 backbone, SE block and classification head remains identical to both centralized DermaNet and FedSE. With the addition of the mask management module that generates the masks at initialization and applies them during training, and also while without altering forward or backward pass. It ensures that the performance comparison is fairly based on aggregation strategy rather than architectural variations.

Our training hyperparameters match the baseline. 5 global rounds with 4 local epochs per round, batch size 32, AdamW optimizer with learning rate 1×10^{-4} and weight decay 1×10^{-4}, and focal loss with $\alpha = 1$, $\gamma = 2$. We implement data augmentation following the same minimal protocol, and evaluating using identical per client test sets with weighted F1 scores and accuracy. Also for the masks generated locally using deterministic seeds adds no additional communication overhead as they are never transmitted. Furthermore, the masks are reproducible from the clients' ids. Hence, the minimal implementation complexity and no communication overhead makes FedMD compatible with any standard federated learning framework, giving extensive support in custom aggregation functions, also not requiring protocol modifications beyond what FedAvg provides.

Chapter 6

Results

Having outlined previously, after 20 training epochs, model performance was measured on the corresponding test set of each dataset. Classification quality was measured in all experiments by accuracy, precision, recall, and F1-score. Confusion matrices and classification reports were generated for per-class performance. Results for each experiment are detailed below.

6.1 HAM10000 (RegNetY-32GF)

On the HAM10000[5] test set, the final accuracy reached by the RegNetY-32GF[9] model is 86.48%. The model performed very well on classes with high sample percentages like nevus (F1-score: 0.93) and basal cell carcinoma (F1-score: 0.82), but less on low sample classes like actinic keratosis (F1-score: 0.66) and melanoma (F1-score: 0.68).

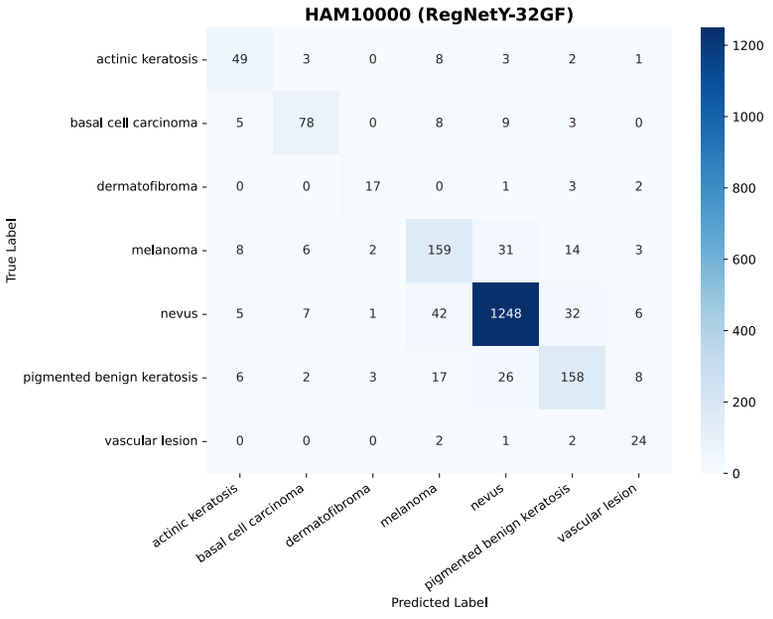


Figure 6.1: HAM10000 (RegNetY-32GF)

The training accuracy exceeded 98% after a few epochs and never dropped after that suggesting convergence was achieved. The test accuracy was stabilized in the range

of 86–88% over the course of the epochs with limited overfitting. The model was stable across classes (weighted F1 score = 0.87; macro F1 score = 0.79) suggesting good performance relative to more frequently and infrequently appearing classes.

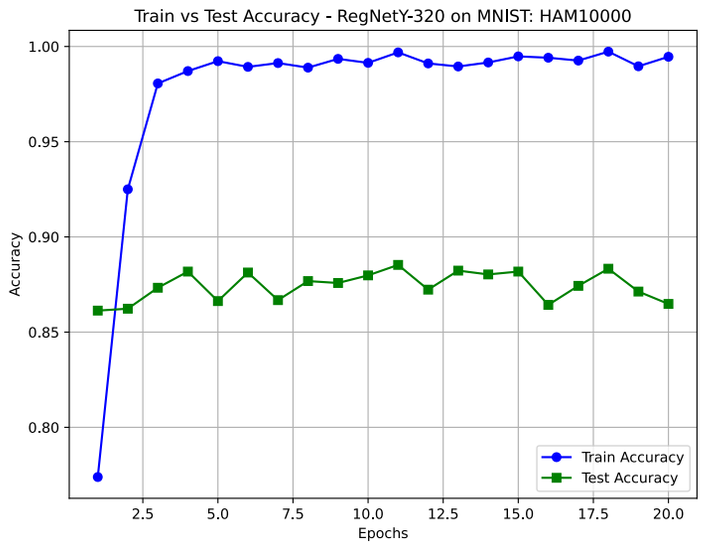


Figure 6.2: Train vs Test Accuracy - RegNetY-32GF on MNIST: HAM10000

6.2 ISIC 9-Class (VGG16)

When assessing the ISIC 9-Class[3] dataset, the VGG16 model final test accuracy came in at 40.68%. Although, the training accuracy for the VGG16 model compiled

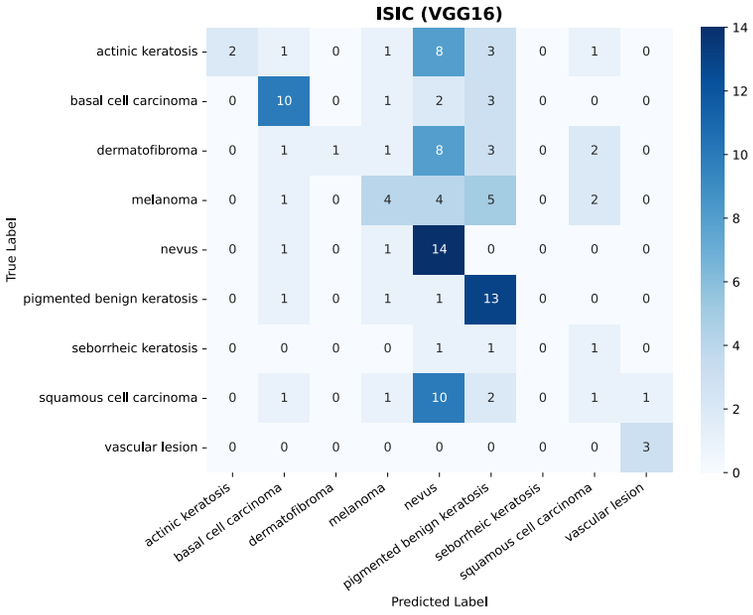


Figure 6.3: ISIC (VGG16)

from the training datasets shows that training accuracy improved with each epoch and reached 86.82% at the 20th epoch. However, the model suffers from overfitting since the test accuracy has not really improved since the first few epochs (as shown in Figure 6.4) and lingers between 33%-42% in the subsequent epochs.
Per-class performance was inconsistent and ranged highly. For some classes, such as vascular lesions, precision and recall calculated to 1.0 as there were very few test samples. For other classes, such as seborrheic keratosis, squamous cell carcinoma, and dermatofibroma, they performed poorly, with F1-scores around or equal to zero. The calculated macro F1-score was 0.36 and the calculated weighted F1-score was 0.33, meaning these predictions were imbalanced and inconsistent.

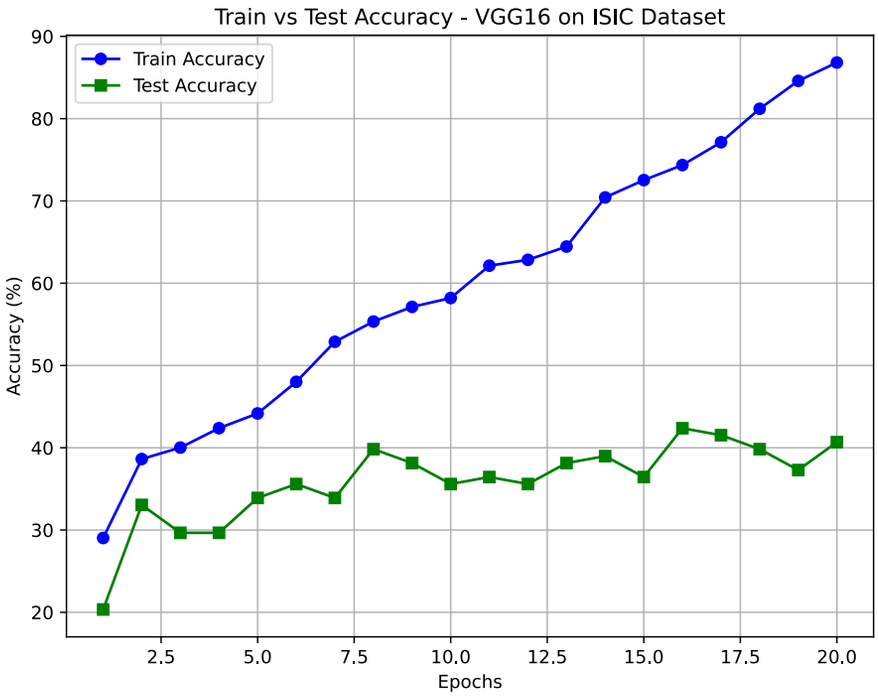


Figure 6.4: Train vs Test Accuracy - VGG16 on ISIC Dataset

6.3 PAD-UFES-20 (SkinLesNet)

The final test accuracy of the custom SkinLesNet model trained on the PAD-UFES-20[4, 10] dataset was 57.48%. The performance continued to improve over time with epoch, training accuracy also reaching 62.22% toward the end. The model adequately generalized despite its simple architecture, though class performance per level significantly varied.

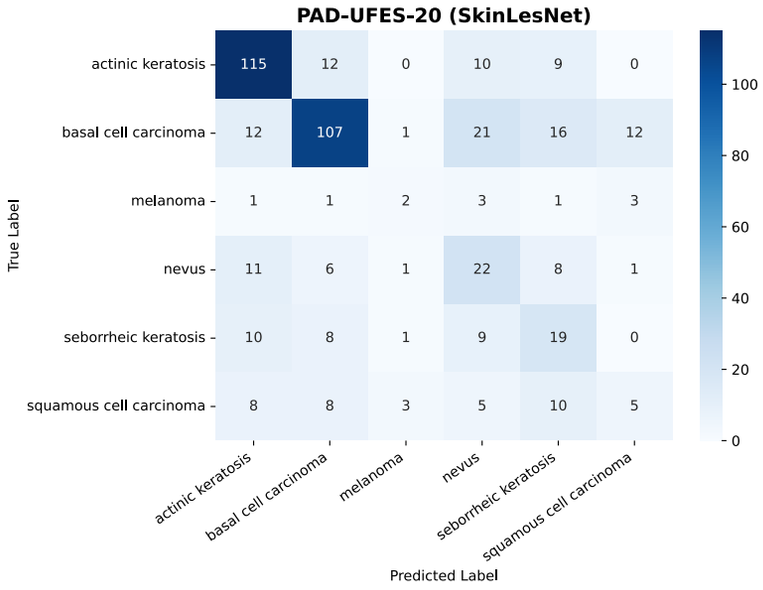


Figure 6.5: PAD-UFES-20 (SkinLesNet)

The best performances were with basal cell carcinoma (F1-score: 0.63), actinic keratosis (F1-score: 0.65), while the weak performances were associated with melanoma and squamous cell carcinoma. Interestingly, squamous cell carcinoma had an F1-score of 0.00, indicating that the model was unable to learn any distinguishing features for this class. The macro F1-score was 0.41 and the weighted F1-score was 0.54; indicating some classification of non-negligible measures within a low-data context.

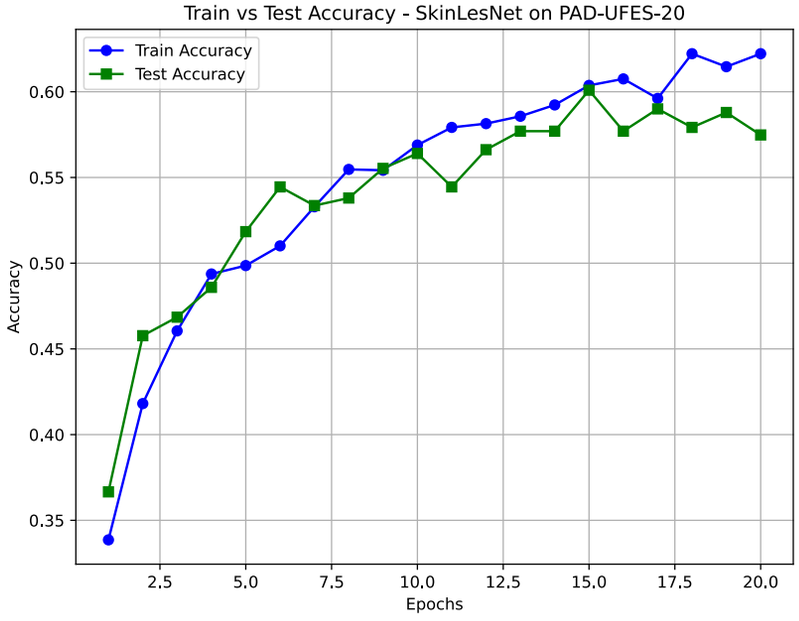


Figure 6.6: Train vs Test Accuracy - SkinLesNet on PAD-UFES-20

6.4 DermaNet Result

To overcome limitations of baseline architectures, we propose DermaNet, a hybrid model which utilizes transfer learning, channel-wise attention, and structured regularization. Table 6.1 gives an overview of DermaNet's performance on all three datasets compared to their baselines.

Dataset	Accuracy	F1-Score	Baseline Model	Baseline Acc	Improvement
HAM10000	87.83%	0.8799	RegNetY-32GF	86.48%	+1.35%
ISIC 9-Class	67.39%	0.6863	VGG16	40.68%	+26.71%
PAD-UFES-20	70.50%	0.7039	SkinLesNet	57.48%	+13.02%

Table 6.1: DermaNet performance compared to baseline models across datasets

The findings show a fascinating pattern. HAM10000[5] noticed a small improvement, while ISIC[3] and PAD-UFES-20[4] showed amazing improvements of 26.71% and 13.02% respectively. Such an observation implies that the architectural choices of DermaNet are particularly advantageous when confronted with challenging scenarios characterized by severe overfitting or domain shift.

6.4.1 HAM10000: Establishing Baseline Competence

Training on HAM10000 shows smooth, steady convergence (Fig. 6.7). After four training epochs only, the accuracy of the model on the test set went up from 72.12% to 82.14%. This indicates that EfficientNet-B3 backbone that is pre-trained provides a good feature representation for dermoscopic images[8]. Learning then entered a refinement phase before reaching 87.83% at epoch 15 and stabilizing.

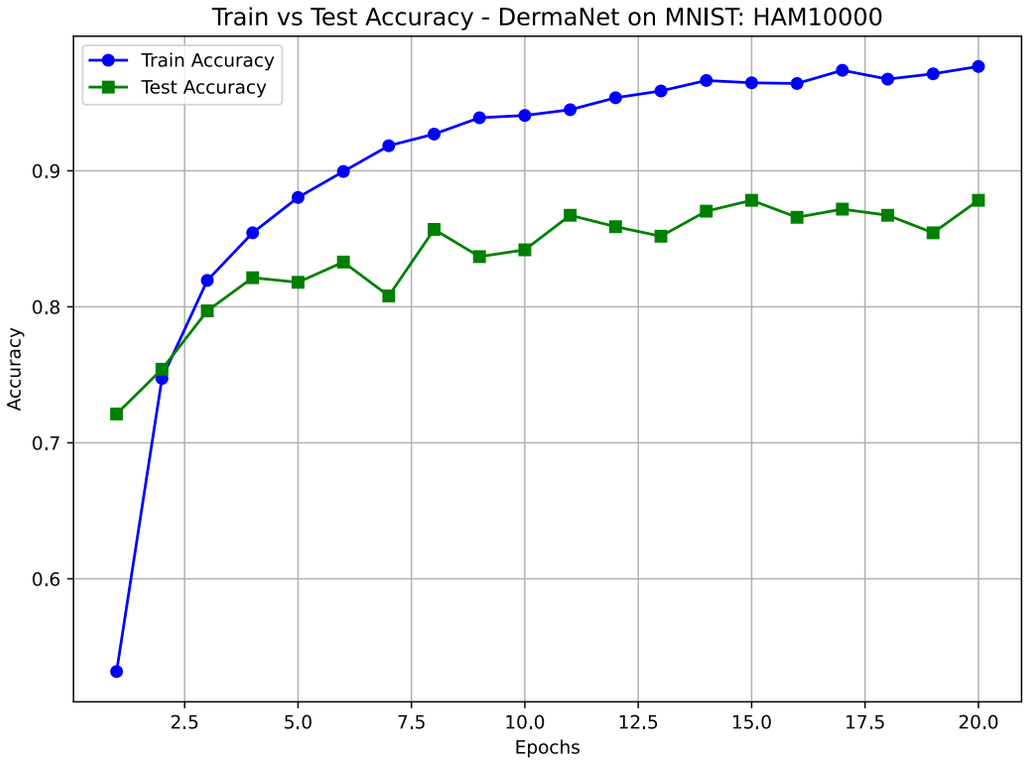


Figure 6.7: Train vs Test Accuracy – DermaNet on HAM10000

What stands out is the controlled train-test gap of $\sim 10\%$. Training accuracy reached 97.67%. So, model has acceptable generalization. The gap was under control which was about 9 percentage points. Different types of lesions exhibited varying levels of effectiveness (Figure 6.8). Vascular lesions were the easiest class to classify, achieving F1: 0.95 and misclassifying only 5 samples out of 29. "Despite being the most frequent class, Nevus had good performance at F1: 0.93." Nonetheless, the clinically important melanoma category was tough. F1: 0.70 means the model misclassified 42 malignant melanoma as benign nevus cases. This pattern is similar to real world diagnostics where the early forms of melanoma resemble benign nevi which are basically benign moles.

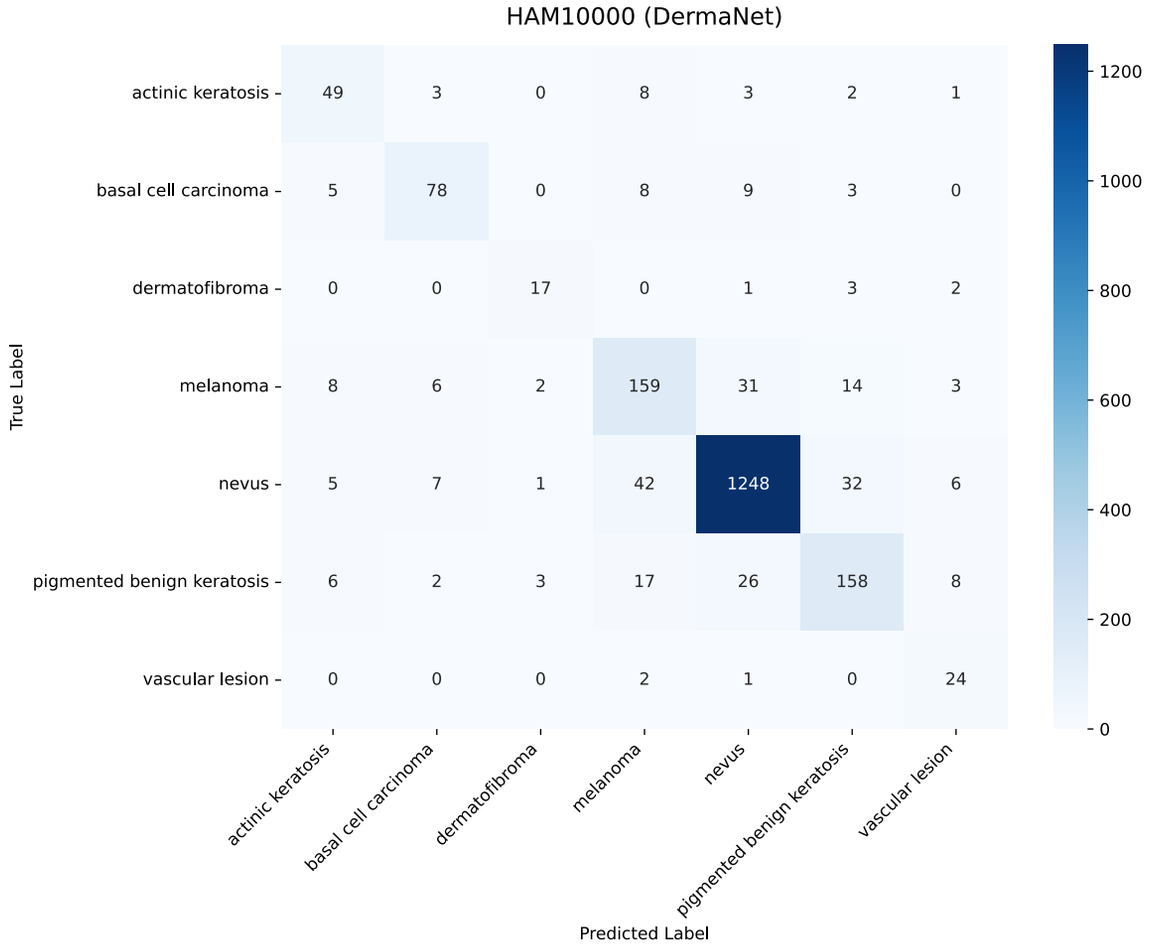


Figure 6.8: Confusion matrix of DermaNet on HAM10000

Weighted F1-score of 0.88 was close to the overall accuracy, while macro F1 was 0.83. The narrow gap between the weighted score and macro score indicates the success of focal loss in preventing the model from ignoring the minority classes in favor of the dominant nevus class[17].

6.4.2 ISIC 9-Class: Recovering from VGG16's Collapse

The ISIC findings indicate an architectural rescue operation. While VGG16 developed severe overfitting, DermaNet managed to recover and obtain an accuracy of 67.39%, a gain of 26.71 percentage points. The emerging training dynamics (Figure 6.9)

witness a rapid increase leading to 65% learning by epoch 3. Unlike HAM10000[5] that the model improved gradually, the ISIC training where the model was already able to pick up the important features but could not manage to go beyond a certain threshold.

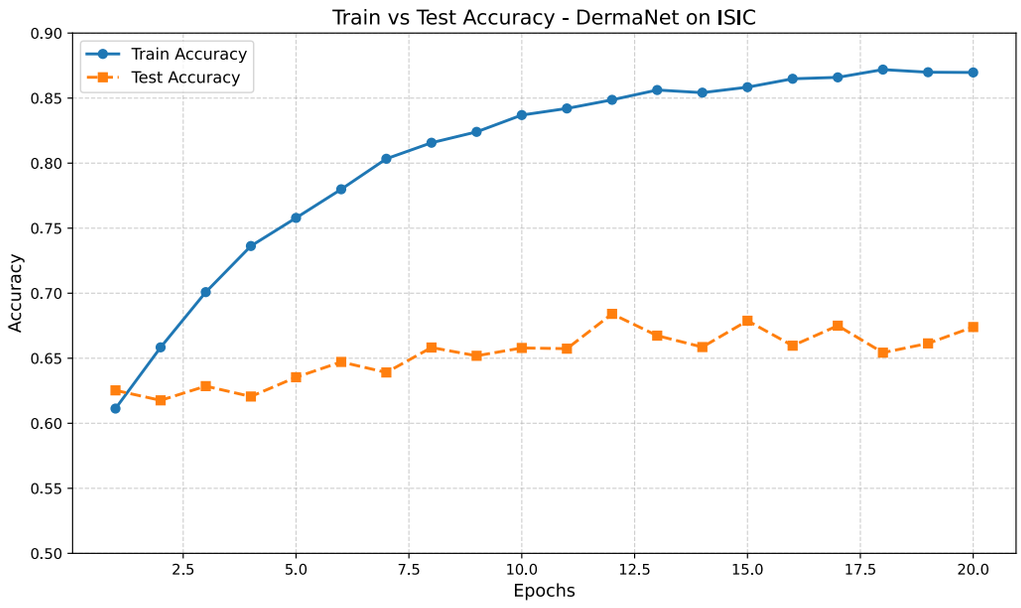


Figure 6.9: Train vs Test Accuracy – DermaNet on ISIC

DermaNet succeed but VGG16 fail because of the regularization. VGG16 with uniform dropout was unable to stop the model from memorizing the training images and achieved 86.82% training accuracy with 40.68 test accuracy. The structured dropout $(0.4, 0.3, 0.2)$ used in DermaNet with focal loss maintained a 23% points gap. The SE blocks add another layer of defence. They are able to learn which features actually matter rather than treating all channels the same.

Class-wise analysis (Figure 6.10) shows many successes and some shortcomings. The model showed an F1 score of 0.90 with 69 correct predictions out of 78 for basal cell carcinoma. Vascular lesions sustained their strong performance at F1: 0.93. Still, the performance of melanoma is dramatically unsatisfactory with F1: 0.30, as we confuse 38 cases with the nevus. Seborrheic keratosis may present a challenge as it has the lowest F1 score of 0.08. This is mainly due to the test set containing just 16 samples, rendering effective statistical learning virtually impossible.

DermaNet did not suffer complete classification failures as VGG16. VGG16 scored multiple F1 scores of 0.00 in the testing phase, whereas DermaNet at least scored some class scores, showing useful learning.

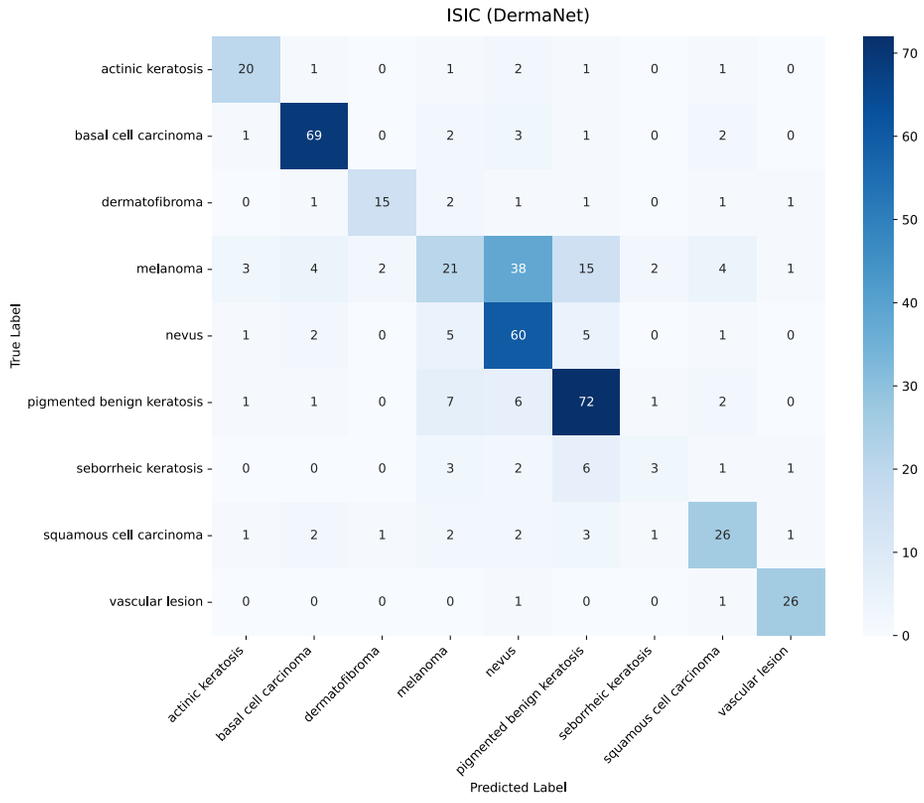


Figure 6.10: Confusion matrix of DermaNet on ISIC

6.4.3 PAD-UFES-20: Crossing the Domain Gap

PAD-UFES-20[4] presented a totally different challenge. The smartphone images lack dermoscopy's controlled lighting and magnification, resulting in a domain shift

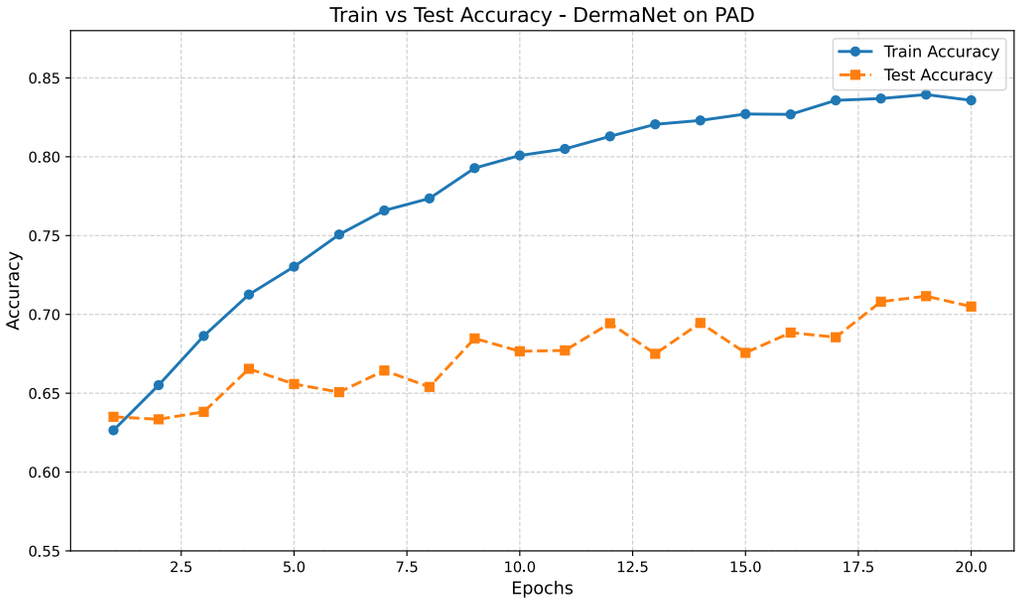


Figure 6.11: Train vs Test Accuracy – DermaNet on PAD

from the pre trained ImageNet. Nonetheless, DermaNet managed to achieve 70.50%

accuracy and SkinLesNet at 57.48% which is an improvement of 13.02%.
The training curve (Figure 6.11) behaves differently than dermoscopic datasets. Test accuracy varies significantly during training, ranging between 60% and 70%. The fact that smartphone images show mild fluctuations rather than instability is due to a smaller test set size and heterogeneous image quality. The training accuracy increased progressively to 85%, having a reasonable gap thus indicating that the model did not memorize artifacts.
The confusion matrix provides some clues why DermaNet's smartphone image handling is better than SkinLesNet's. The F1 scores of actinic keratosis and basal cell carcinoma, which both refer to non-melanoma cancers, are respectively 0.76 and 0.74. The transfer learning from ImageNet, which consists of natural photographs similar to those taken with smartphones, likely provided a better initialization than training from scratch. The SE blocks then adjusted those features for medical pattern diagnostic[7].

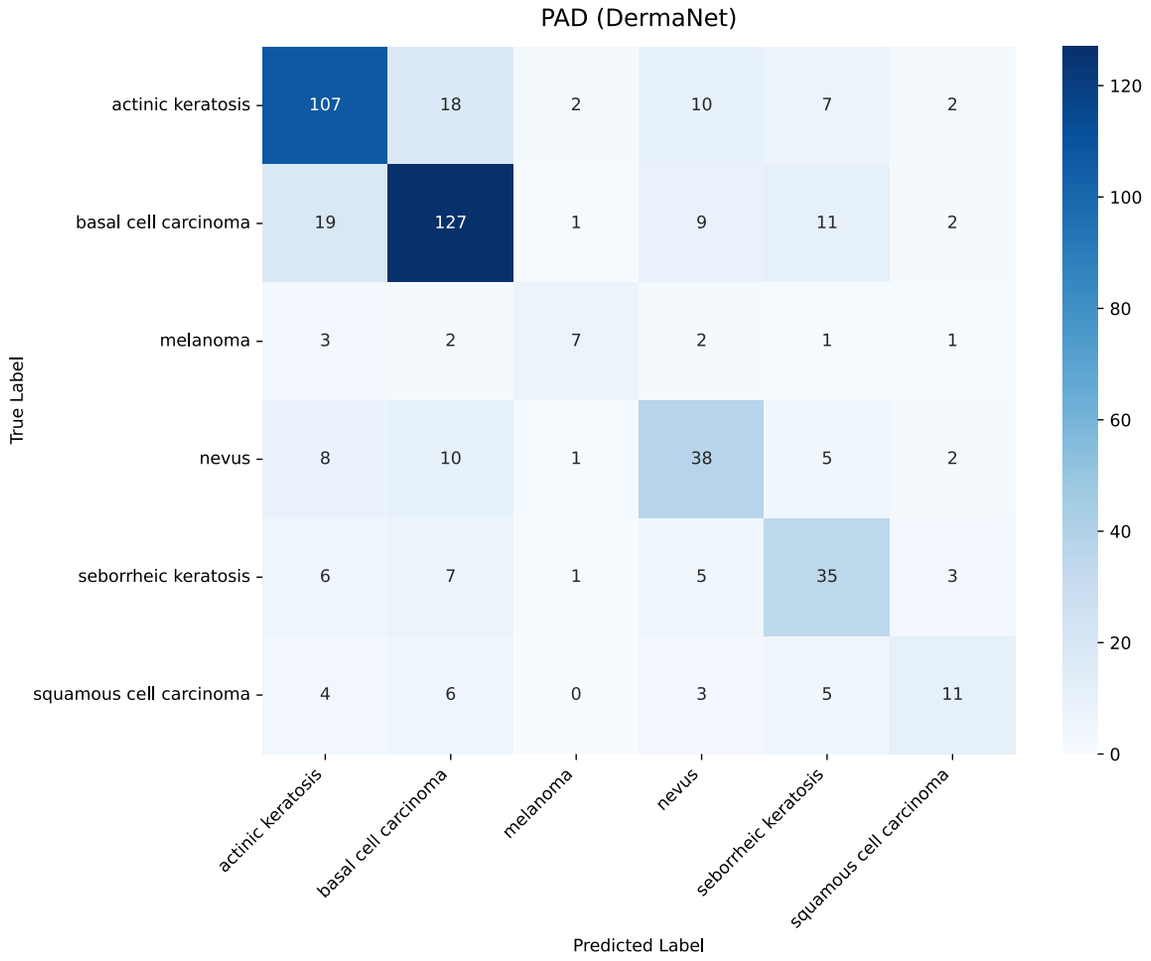


Figure 6.12: Confusion matrix of DermaNet on PAD

Squamous cell carcinoma is still an issue as denoted by F1: 0.29, although at least this is an improvement from SkinLesNet achieving F1: 0.00 on this class. The clinical significance should not be underestimated. Screening for skin cancer using smartphones can be done in addition to available dermoscopic equipment.

6.4.4 Architectural Insights and Training Dynamics

When we compare training curves on different datasets, we can see how DermaNet adapts to different challenges. Smooth convergence with minimal wave as expected since it is bigger and has a consistent dermoscopic quality ISIC learn rapidly at first before being hindered by a plateau. This may reflect the model's learning of key features and inability to differentiate complex models. PAD-UFES-20[4] was the most volatile of the evaluated signals since the accuracy fluctuated between 55% and 70%. The reason for this was that this dataset is small in size and the smartphone images show a heterogeneous quality. Even with these gaps, train and test differences were reasonable for DermaNet on all three datasets.
These improvements came from three architectural decisions. Channel wise feature recalibration is enabled by SE blocks, allowing the model to highlight diagnostically relevant patterns and downplay noise[7]. This helped especially with the distinction of melanoma from nevus, which relies on texture rather than overall appearance. Focal loss automatically down-weights easy examples and focuses training on hard, misclassified examples with no manual class weight tuning needed[17]. Thanks to this mechanism, the model did not ignore the dermatofibroma and vascular lesion. Through a structured dropout schedule of $(0.4, 0.3, 0.2)$ on top layers, it helps the convolutional layers to learn diverse feature in the early layers while allowing the lower layers to keep task specific decision patterns. Therefore there was not such severe overfitting that we see in VGG16.

6.5 Cross-Dataset Evaluation of Model Performance

We tested cross dataset more thoroughly to evaluate the model's capacity to generalize beyond the source dataset. Every model was run on the test set from all three datasets without any fine tuning. This was done to mimic real life where a model trained at one institution encounters data from other hospitals with different acquisition protocols, patient demographics and imaging devices. This evaluation addresses a fundamental obstacle behind federated learning: centralized models trained on a relatively large dataset with a complex architecture do not generalize well to the heterogeneous data distribution present in medical datasets.
Table 6.2 shows the cross dataset accuracy results for all models. The baseline architectures faced considerable performance issues when subjected to out of domain testing. The RegNetY-32GF model trained on HAM10000[5] was able to perform well with in domain results of 87%. However, it dropped significantly to 59% on ISIC[3] dermoscopy images and collapsed to only 11% on PAD[4] smartphone images, corresponds to drop of 28 and 76 percentage points respectively. The VGG16 model trained on HAM achieved 76% on HAM, 53% on ISIC, and 20% on PAD. When trained on ISIC it achieved 28% on HAM, 41% on ISIC, and 25% on PAD. The native performance on ISIC is 41%. SkinlesNEt trained on HAM scored 75% on HAM, 36% on ISIC, and 19% on PAD and when trained on PAD it scored 19% on HAM, 22% on ISIC and 57% on PAD.

Trained Dataset	Model	HAM	ISIC	PAD
HAM	RegNetY	87%	59%	11%
HAM	VGG16	76%	53%	20%
HAM	SkinLesNet	75%	36%	19%
HAM	**DermaNet**	88%	40%	20%
ISIC	RegNetY	53%	52%	18%
ISIC	VGG16	28%	41%	25%
ISIC	SkinLesNet	22%	35%	18%
ISIC	**DermaNet**	14%	67%	17%
PAD	RegNetY	16%	15%	70%
PAD	VGG16	19%	13%	52%
PAD	SkinLesNet	19%	22%	57%
PAD	**DermaNet**	10%	16%	70%

Table 6.2: Cross Dataset Accuracies

The results reveal a systematic phenomenon where all the baseline models experience a catastrophic drop in performance when deployed across institutions resulting accuracy drop of 10–70 percentage points. When transferring from HAM to ISIC, the drop varies by model such as, RegNetY: 87%–59%. Meanwhile, the imaging modality gap between dermoscopy and smartphone photography is almost impossible to bridge.

Although the design utilizes advanced architectural components, DermaNet shows similar catastrophic cross dataset failure due to overfitting. DermaNet, when trained on HAM10000[5], has an in domain accuracy of 88%, which drops to 40% on ISIC[3] and 20% on PAD[4]. This reflects a loss of 48 and 68 percentage points, respectively. The ISIC trained variant performed even worse by scoring 67% on its native dataset but collapsing to 14% on HAM and 17% on PAD accuracy levels almost consistent with random guessing on these multi class problems.

The PAD trained DermaNet was able to achieve 70% on smartphone images. However, it failed to generalize to dermoscopic datasets. It only managed 10% accuracy on the HAM dataset and 16% on the ISIC dataset.

A key insight from the results is that architectural sophistication cannot compensate for distributional differences. The SE blocks seem to overfit in a dataset specific manner for texture patterns, rather than learning the channel wise dermoscopic features that are useful for diagnosis. For example, the high resolution dermoscopy (HAM, ISIC) emphasis of fine grained vascular and pigmentation patterns is not transferable to smartphone images with inconsistent light, resolution and field of view (PAD). On the other hand, the smartphone images do not contain sufficient information about the structures and hence explain an accuracy of 10–16% when the PAD trained model is applied to a dermoscopic dataset.

Federated Learning Implications

These broad cross dataset results confirm the failure of centralized training approaches as models cannot generalize across heterogeneous data sources, regardless of architecture, dataset size, or training sophistication. DermaNet is a dermoscopic image classifier with domain specific attention mechanisms and clinical realistic training

protocols. It achieves at best 40% cross domain accuracy. These conduct are clinically unacceptable and potentially dangerous for automated diagnostic deployment.
This systematic failure indicates that federated learning is not just a privacy preserving alternative to centralized training, but an essential approach for creating robust and generalizable skin treatment detection systems. Federated learning is the only approach that is able to do this by allowing collaborative training across heterogeneous institutional data sources without requiring data centralization, it can lead to models that can reliably work across the different imaging devices, patient populations, and clinical protocols seen in day to day dermatological practice. Our federated learning implementation and the collaborative training of various institutions resolve the generalization failures we found in Table 6.2.

6.6 Federated Learning Baseline: FedSE results

The cross dataset evaluation revealed that models, be it complex or not, do not generalize of heterogeneous medical datasets. Federated Learning (FL) provides a solution to this core limitation by facilitating joint training across institutions without sharing data. In order to build a federated baseline, we implemented FedSE, a standard FedAvg based system that deploys DermaNet across three heterogeneous clients which represent real world hospital scenarios. HAM10000[5] (10,000 dermoscopic images, 7 classes), ISIC[3] (2,300 dermoscopic images, 9 classes), PAD-UFES-20[4] (2,200 smartphone images, 6 classes). This baseline shows the potential but also the significant limits of naive federated deployment that uses stochastic regularization. The federated training had an overall configuration of 5×4 global communication rounds with 4 local epochs per communication round at each client to strike a balance between communication efficiency and client drift. Each client had its own classification head for institution specific label spaces. They controlled the classifier heads while only sharing the feature extractor EfficientNet-B3 backbone + SE block via FedAvg. The FedAvg was weighted according to dataset size. The system maintained the original DermaNet training protocol, including AdamW ($\text{lr} = 1\text{e}{-4}, \text{weight decay} = 1\text{e}{-5}$), focal loss ($\alpha = 1, \gamma = 2$), structured dropout ($0.4 \rightarrow 0.3 \rightarrow 0.2$), and clinical realistic augmentation. The three clients varied a lot. HAM was a fully fledged dermatology clinic with dermoscopy standardized protocol. ISIC pooled dermoscopic data across institutions, with varying protocols. PAD had lower quality smartphone photographs, from resource constrained settings. These variations caused a significant domain shift.
Figure 6.13 demonstrates the dynamics of federated training over 5 global rounds. All three clients consistently converged. For training accuracy, the HAM, ISIC and PAD clients achieved 67.66%, 37.65% and 47.69% in Round 1. This increased to 93.13%, 81.47% and 79.46% in Round 5. Notably, training loss decreased steadily from an average of 0.8712 in Round 1 to 0.1870 in Round 5. Hence, there was stable optimization without diverging.

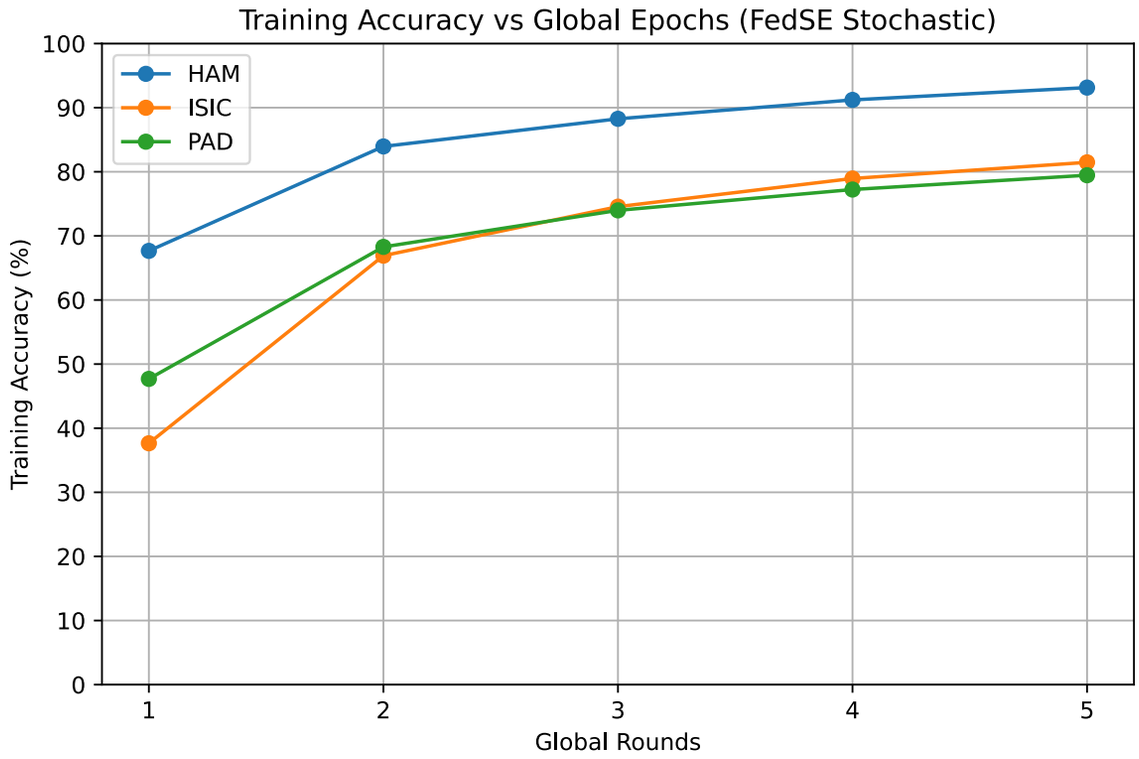


Figure 6.13: Training accuracy across global epochs

The evaluation of the final test set has produced a promising result as compared to the performance using centralized cross-dataset. The score of HAM is 84.54% (F1: 0.8532), ISIC is 69.55% (F1: 0.7153) and PAD is 65.29% (F1: 0.6354). When applied to test distributions and trained on distinct training sets, our model delivers significant enhancement compared to cross domain centralized models, which collapsed to 10–20% performance.

Table 6.3 compares centralized and federated performances directly. In HAM's case, the federated accuracy (84.54%) was slightly lower than the corresponding accuracy of the centralized RegNetY-32GF (86.48%). The difference is marginal (1.94 percentage points). Above all, ISIC's federated performance of 69.55% was much higher than VGG16's centralized baseline of 40.68%, a delta of 28.87 percentage points. Under federation, PAD reached 65.29%, considerably better than SkinLesNet's centralized performance of 57.48% a gain of 7.81 points. The results indicate that federated learning can successfully leverage the data diversity across institutions, especially for clients with small or complex datasets that would otherwise be affected by a lack of local training data.

Client	Centralized Baseline	Baseline Acc	FedSE	Improvement
HAM	RegNetY-32GF	86.48%	84.54%	-1.94%
ISIC	VGG16	40.68%	69.55%	+28.87%
PAD	SkinLesNet	57.48%	65.29%	+7.81%

Table 6.3: Centralized vs. Federated Performance Comparison

These promising results notwithstanding, FedSE has a fundamental limitation in comparison with the centralized DermaNet performance as we see. The federated

accuracy of HAM[5], which is 84.54%, is 3.29 percentage points lower than that of centralized DermaNet (87.83%). To be more specific, ISIC[3] achieved only 69.55% under federation and 67.39% centralized for DermaNet, which is a mere 2.16 point gain that falls short of expectations. The most severe form of degradation is shown by PAD[4], which suffers a loss of 5.21 points from 70.50% centralized to 65.29% federated, even with diverse access to multi institutional data.
The reason behind this systematic underperformance is the zero dilution problem in standard FedAvg, caused by stochastic dropout of clients. During local training, the dropout masks at each client deactivate different neurons in the SE block and classification head at random. When the server does standard averaging on the weights, it treats the structural zeros (the neurons dropped for regularization) as learned values that represent "unimportant" features. The global model's channel recalibration in the semantically misaligned branches is misled into forcing the SE weights, originally meant to span 0.1–0.9, now collapse to 0.4–0.6. This equips the attention mechanism with a non-selective body.
Analysis of per round training dynamics reveals further signs of zero dilution. The convergence is stable, but the rate is not optimal. Across clients there is a large and notable variance the largest dataset, HAM, does not degrade very much while PAD the smallest dataset with the highest domain shift degrades the most. The detection of the minority class is particularly harmed, as diluted SE weights fail to give sufficient weight to rarer diagnostic features. The F1 score differences observed in centralized and federated settings (HAM: 0.8799 $\rightarrow$ 0.8532, PAD: 0.7039 $\rightarrow$ 0.6354) show that naive aggregation fails to adequately deal with class imbalance issues.
The experiments of FedSE also serve to establish another important baseline. Federated learning can effectively outperform cross domain centralized deployment. This finding validates that FL is indeed a suitable approach for multi institutional skin cancer detection. Nonetheless, the 3–5% accuracy drop from centralized DermaNet performance shows that standard FedAvg is not compatible with architectures employing heavy stochastic regularization. The zero dilution problem demonstrates a fundamental algorithmic limitation rather than simply a hyperparameter tuning problem. It therefore motivates the development of mask-aware aggregation strategies which respect the structural nature of the zeros of dropout. We will explore this in more detail in this section.

6.7 FedMD Result

6.7.1 Experimental Results and Performance Analysis

Training Configuration and Setup

We used three heterogeneous clients representing distinct medical institutions to evaluate FedMD: HAM (Hospital A with 10000 dermoscopy images from HAM10000[5] dataset), ISIC[3] (Hospital B with 2200 images from ISIC dataset), and PAD[4] (hospital C with 2300 clinical photographs from PAD-UFES-20 dataset). Our batch size is 32, the training is done for 5 global rounds with each client performing 4 local epochs per round. We also used AdamW optimizer with learning rate 1×10^{-4} and weight decay 1×10^{-4}, combined with focal loss ($\alpha = 1$, $\gamma = 2$) to handle class imbalance. The SE dropout rate was 30%, ensuring approximately 70% active

parameters per client in the attention layers.

Parameter Coverage Analysis

Our mask aware aggregation strategy provides diversity and robustness as discussed earlier. Analysis of the SE attention FC layers revealed that on average each weight in the attention mechanism was actively trained by two out of three clients. In the first SE FC layer (1536→96), the clients dropped only 2.76% of the parameters, while the second FC later (1536→96) showed a similar rate of 2.67%. resulting in over 97% of attention parameters in both layers to receive updates from at least one client, ensuring near complete coverage through deterministic masking. Also, through the small fraction of untrained parameters we retained their pretrained ImageNet initialization, which provides low level feature representation without hindering model performance.

Convergence Behavior and Training Dynamics

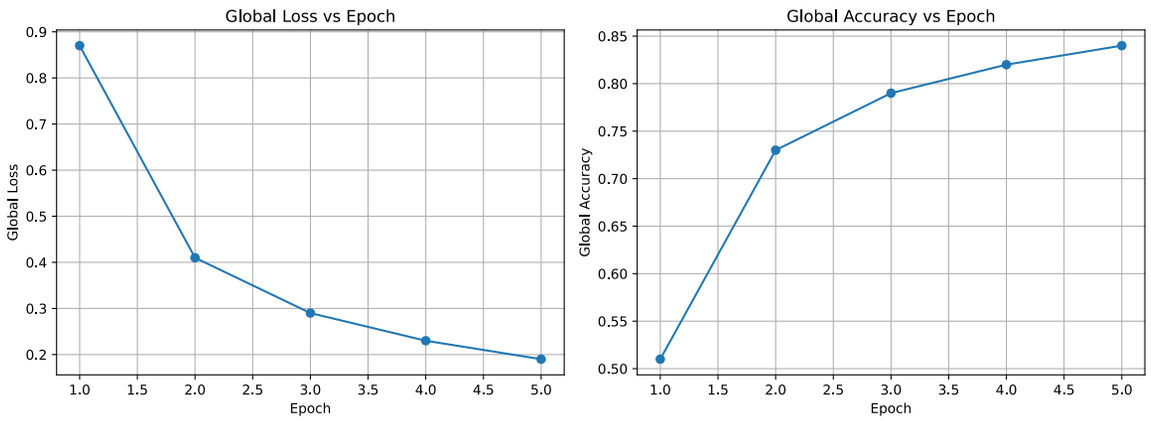


Figure 6.14: Global Loss and Accuracy per Epoch

Figure 6.14 visualizes the convergence characteristics of FedMD over 5 global rounds. Here we see a constant decrease in global loss from 0.8719 at epoch 1 to 0.1921 at epoch 5, showing stability and monotonic convergence without oscillations or divergence. Similarly, the global accuracy has improved from 51.04% to 84.27% representing a 33.23% gain over 5 rounds. We see a steep improvement happen during the first two epochs (51.04%→72.62%), where fundamental dermoscopic features were learned from collaborative training. Later epochs show diminishing but steady gains, as the accuracy curve plateauing toward 84% by epoch 5, suggesting a convergence within the training time.

The figure 6.15 shows that per client training accuracy trajectories converging towards a fixed range while being heterogeneous datasets themselves. Client HAM, with the largest and most balanced dataset, achieved the highest training accuracy throughout all epochs, starting from 67.08% and reaching 92.46% by epoch 5. Then client ISIC starting from 37.51% (lowest initial accuracy due to diverse imaging conditions) to 81.68% gaining over 44% accuracy. And client PAD also gained similar accuracy gain from 48.53% to 78.67%, benefiting most from the shared SE attention representations

despite having clinical diverse photographs. Therefore, they converged towards 78 to 92 percentage range by epoch 5 demonstrating that mask aware aggregation successfully transferring knowledge across heterogeneous data distributions.

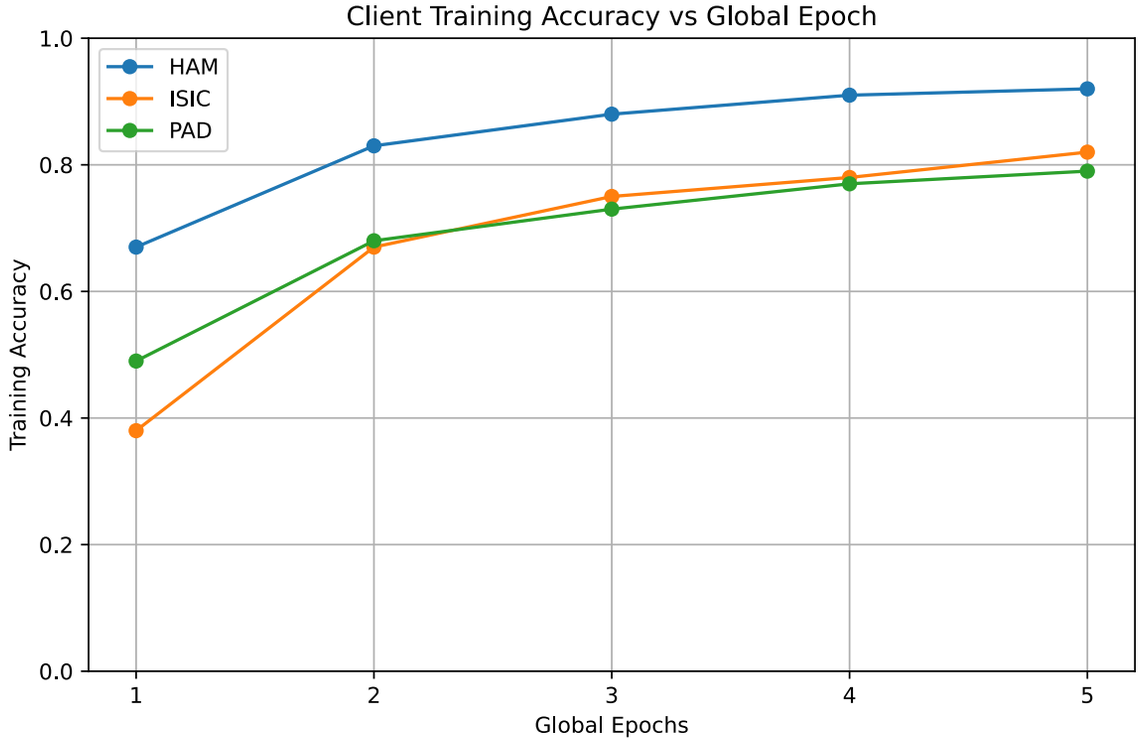


Figure 6.15: Client wise training accuracy vs epochs

Final Test Performance and Clinical Metrics

Table 6.4: FINAL TEST PERFORMANCE SUMMARY

Client	**Accuracy (%)**	**F1-Score**	**Test Loss**
HAM	85.59	0.8626	0.4286
ISIC	71.92	0.7301	0.7119
PAD	75.29	0.6852	0.9716
Average	**77.60**	**0.7593**	**0.7040**

Table 6.4 summarizes the final test set performance after 5 global rounds. Client HAM achieving the highest test accuracy of 85.59% with weighted F1 score of 0.8626, representing its relatively large training set and balanced class distribution. Also, the lesion types achieved strong performance, basal cell carcinoma (F1=0.90), dermatofibroma (F1=0.96), nevus (F1=0.92), and vascular lesion (F1=0.88) all exceeded 0.85 F1-score. Then, the test loss of 0.4286 showing well calibrated predictions without severe overconfidence.
While having less training samples and greater imaging heterogeneity from HAM, client ISIC showed reasonable performance having 71.92% accuracy with a F1 score of 0.7301. similarly, it excelled at basal cell carcinoma (F1 = 0.98) and vascular lesion (F1 = 0.97) classification, but struggled with melanoma (F1 = 0.45, recall = 0.39) and seborrheic keratosis (F1= 0.04, recall = 0.06).

Whereas, client PAD's performance shows its characteristics clearly. With an accuracy of 75.29% and F1 score of only 0.6852, it shows its significant class imbalance in both prediction and performance.

6.7.2 Comparative Analysis: FedMD vs FedSE

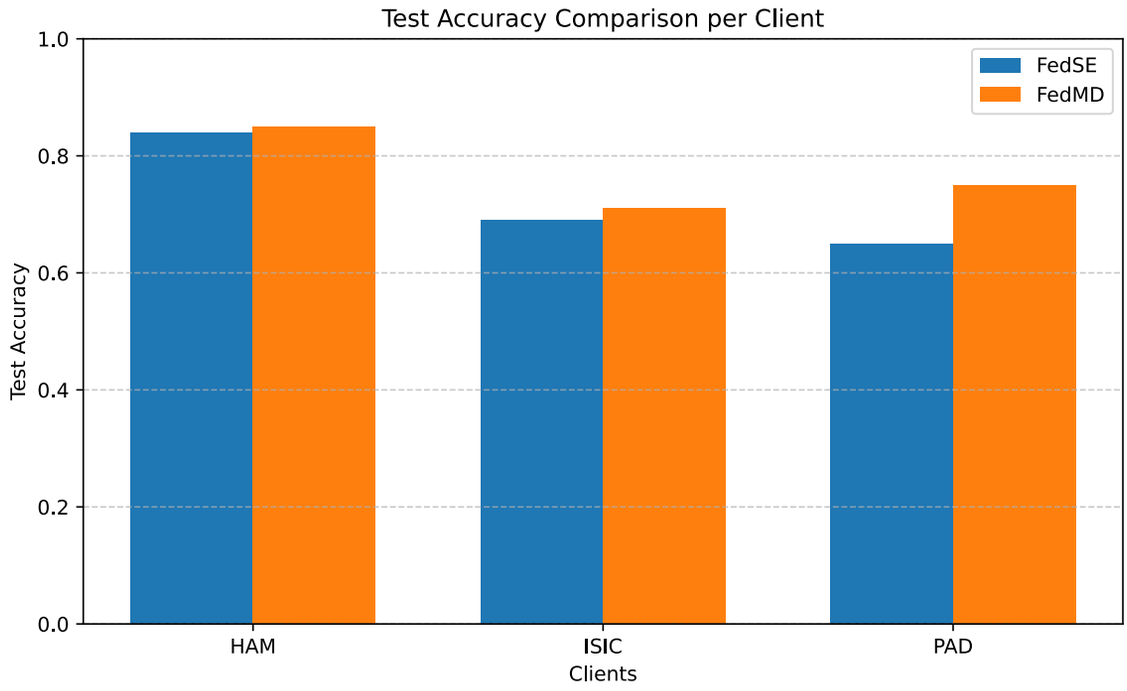


Figure 6.16: Test Accuracy Comparison per Client

Here we can clearly see the difference between the test accuracy of FedSE and our proposed FedMD model, in figure 6.16. Our FedMD model constantly outperformed the baseline across all three clients, with improvements of +1.05% for HAM (84.54%→85.59%), +2.37% for ISIC (69.55%→71.92%), and +9.90% for PAD (65.39%→75.29%). The drastic improvement of PAD is visible. It has significantly boosted with the help of shared SE representations. Also our selection of focal loss and AdamW optimizer helped the imbalanced class representation[17, 66].
And as for the modest improvements at HAM (+1.05%) and ISIC (+2.37%) shows their relative robustness to zero dilution, as they share similar dermoscopy imaging modalities and theirs weights updates are less conflicting naïve aggregation. However, the consistent gain across all clients validates that mask awareness provides systematic benefits rather then trading off performances. The elimination of zero dilution yields practical gain, as the aggregation accuracy improved approximately +5.7%, demonstrating it in federated medical imaging scenarios.

Lesion Specific Performance Analysis

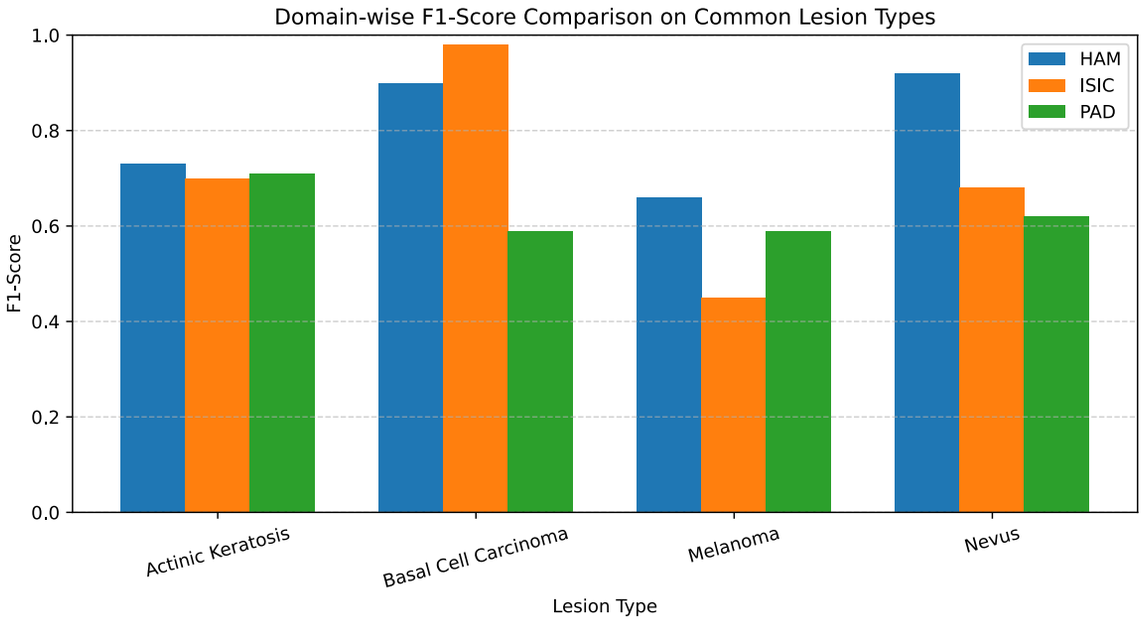


Figure 6.17: Domain Wise F1 Score Comparison on Common Lesion Types

In figure 6.17 we analyze between the four common lesion types that appear across the clients: actinic keratosis, basal cell carcinoma, melanoma, and nevus. As for actinic keratosis, all clients performed comparable similar performance (HAM: 0.73, ISIC: 0.70, PAD: 0.71), suggesting a consistent visual characteristic across all imaging modalities. As for Basal cell carcinoma performed great in HAM (0.90) and ISIC (0.98), but had performed poorly in PAD (0.59), suggesting its characteristics are also very visible in dermoscopic images but PAD's clinical photographs failed to highlight most of it, resulting in such contrast performance.
Melanoma classification showed that, HAM and PAD with their respective F1 score 0.66 and 0.59, but ISIC collapsed to 0.45. From ISIC's classification matrix we see that 36/90 melanomas were classified as seborrheic keratosis, indicating the model learned correlations between features (dark pigmentation patterns) common to both lesion types in ISIC's particular dataset. It shows a significant obstacle in federated learning for medical imaging, where even with mask aware aggregation, clients with small sample sizes(ISIC had only 90 melanoma test cases) can not receive sufficient training to learn robust classifiers.
Similar analysis can be seen in Nevus classification, where HAM (0.92) performed well for having large training sets and PAD(0.62), ISIC (0.68) performed comparatively low. One of the reason for low performance can be nevus resemblance to early stage melanoma, and distinguishing needs subtle pattern recognition that can be achieved from large training sets such as HAM. Though, PAD's 0.62 F1 score despite using clinical images validates the mask aware aggregation successfully transferring dermoscopic nevus features to clinical imaging domain, enabling reasonable cross generalization.

Practical Implications and Clinical Relevance

The above results and analysis shows that FedMD effectively solves zero dilution while enabling shared learning across heterogeneous medical institutions. The mask awareness has also improved performance where domain shift and class imbalance would cause catastrophic performance under naive federated averaging. Meanwhile, we maintained strict data privacy, the aggregation accuracy of 84.27% and mean of F1 score 0.726% has approached the performance of centralized models trained on data.

from the clients performance variations (HAM: 85.59%, ISIC: 71.92%, PAD: 75.29%) we can suggest that federated models should be evaluated and calibrated on each clients local test set before clinical use. also from the lesion based performance analysis we realize that while large datasets help improve performance and capture generalized dermoscopic features, certain lesion types like melanoma and squamos cell carcinoma require more fine tuning based on institution or larder federated training sets to achieve acceptable results.

Also our decision of 30% dropout masking strategy resulting in approximately 97% SE parameters at least trained by one client and the rest helped retain pretrained data. and the result analysis confirms the robustness and creating real world medical federated deployment, where institutional participation may fluctuate due to technical issues, policy change or patient privacy concerns.

Chapter 7

Future Work

Our research has demonstrated the effectiveness of mask aware federated learning for skin cancer detection, but its results are still far from expected. Several promising research directions are still unexplored that could strengthen our theoretical foundations and practical deployment of FedMD. In our research we involved three heterogeneous clients representing distinct hospital environments. In real world federated networks, it may include tens or more participating institutions with vast heterogeneous data in quality and quantity, along with institution specific protocols and patient demographics. Further research needs to ensure mask aware aggregation maintains its zero dilution mitigation benefits as federated size increases. It also introduces additional challenges such as statistical heterogeneity across clients, increased probability of client dropout during training rounds, and whether it adds any structural or computational complexity or not. Thus, it is highly important to investigate whether our deterministic masking strategy scales effectively or requires additional modifications based on increased federations of big and small institutions with highly non-IID data distribution.

Our model employs a fixed dropout of 30% and a deterministic seed based mask generator across all clients(based on client id). If clients have large, high quality datasets they benefit from lower dropout rates to preserve more learned features, while clients with small or noisier data require more dropout to regularize their data. Thus, it is necessary to do further research on the impacts of different dropouts on FedMD. Also, an adaptive strategy that dynamically adjusts dropout rates based on clients' different qualities could better optimize FedMD results.

Additionally, current experiments mostly focus on dermoscopic and clinical photograph image data. Incorporating additional modalities such as patient metadata (age, sex, lesion location, family history) would improve diagnostic accuracy but introduces significant challenges. Patient metadata is highly sensitive information and requires following various privacy protocols. In spite of all these, aggregating heterogeneous data differently across modalities while ensuring mask awareness requires very complex and careful architectural designing.

As our study focuses on convolutional networks using squeeze excitation attention. However, there are various studies that use different attention and dropout mechanisms as explored in literature review. And in recent times, transformer based models are widely used in medical imaging. Models such as Vision Transformer (ViT), Swin Transformers and hybrid CNN Transformers architectures that rely on multi head self attention have achieved state-of-the-art performance on various

diagnostic tasks. Therefore, FedMD's applicability would extend if further investigation could ensure whether zero dilution similarly affects transformer attention heads, key value projections and feed forward layers. Primary experiments suggest similar structural zero patterns in transformers models from attention dropout, although higher dimensionality and multi head structures may require modified mask aware aggregation formula.

Moreover, we have seen that zero dilution degrades federated performance by 3-5% and corrupts attention weight distribution, but rigorous theoretical analysis remains incomplete. A deeper analysis of convergence could clarify how zero dilution influences training dynamics, generalization and final model quality. Also understanding how factors such as dropout rate , number of clients, and parameter coverage interact could help estimate the expected amount of dilution. It would provide guidance for choosing better dropout and mask settings. Thus, comparing the convergence behaviour of standard FedAvg and mask aware aggregation in different heterogeneous settings would support our empirical findings and extend a better guide for future research.

Research such as ours may be essential for algorithm development, but it can not fully capture real world deployment challenges. To provide clinical validation, employment at partner hospitals with dermatologists actively using FedMD predictions would assess practical utility. Such would evaluate not only diagnostic accuracy but also interface usability and prediction explanations quality and workflow integration. Collecting live feedback would help in model refinement and help evaluate the effectiveness of mask aware aggregation over extended deployment periods with dynamic data arrival times and frequencies.

Besides, in our experiment we assume all clients complete local training before aggregation occurs. However, real hospital networks face bandwidth constraints, unreliable connectivity, and varying computational resources which affect model performance, as discussed in literature reviews. Hence, asynchronous federated learning, where the servers aggregate updates as they arrive rather than waiting for stragglers, could improve training efficiency but introduces staleness, where clients train outdated global models. Therefore, further investigation is needed to check whether mask aware aggregation remains effective in such scenarios and can the model perform or further modifications are essential for a decent performance.

Data augmentation is an approach which can result in higher accuracy, but it can result in several key issues. As we know that picture quality and alignment is institution specific. Thus, to filter one client's image for better representation, means it needs to be done across all clients. But simply following a simple augmentation architecture will not improve global performance, it may lead to global model overfitting that client or completely diverge from it. Hence, for future work it can be studied for dynamic client specific augmentation to achieve higher accuracy.

Addressing these research directions would strengthen FedMD's theoretical foundations, extend its application to state-of-the-art settings, and facilitate real world clinical deployment in various institutions with heterogeneous network, clients and diverse constraints and requirements.

Chapter 8

Conclusion

This thesis tackles the dual problems of domain shift and data privacy in skin cancer detection with an extensive study of centralized deep learning, federated optimization, and aggregation schemes for attention-enhanced medical imaging models.

We present the first results of our baseline reproduction across three state of the art architectures, RegNetY-32GF, VGG16, and SkinLesNet, on HAM10000, ISIC and PAD-UFES-20 datasets, which deliver a critical revelation. Models achieving 85 to 90% accuracy on their training distribution grievously fall down to 10 to 40% when faced with heterogeneous institution data. This systematic cross dataset assessment quantifies the generalization crisis which makes centralized training fundamentally inappropriate for clinical deployment in which hospitals vary in imaging equipment, acquisition protocol, and patient demographics. Despite any architecture sophistication, such failure continues to exist. It shows that attention mechanisms and advanced regularization are not able to resolve distributional mismatch. This is when training data comes from one institution only.

To overcome these limitations, we construct DermaNet that is a hybrid architecture formed using EfficientNet-B3 backbone with Squeeze-Excitation attention, structured dropout (0.4→0.3→0.2) and focal loss to address class imbalance. DermaNet achieves 87.83% accuracy on the HAM10000 dataset but its most important achievement is that it recovers from severe baseline failures on challenging datasets. Specifically, it improves ISIC performance over VGG16 by 26.71 percentage points (40.68%→67.39%) and PAD-UFES-20 performance over SkinLesNet by 13.02 points (57.48%→70.50%). Based on these results, domain optimized architectures with clinically realistic training protocols provide several benefits. However, DermaNet too suffers from devastating cross-dataset collapse (88% on HAM→40% on ISIC, 67% on ISIC→14% on HAM), confirming that architectural innovation cannot substitute for the diversity of multi institutional data.

The need for powerful and privacy preserving skin cancer classifiers has made federated learning a must. Our FedSE baseline, which utilizes DermaNet over diverse 3 clients through usual FedAvg, achieves 84.54%, 69.55%, and 65.29% accuracy on HAM, ISIC, and PAD respectively, and are remarkably higher than 10–40% accuracies in cross domain centralized deployment. This demonstrates that collaborative training allows models to access institutional diversity without the need to centralize sensitive patient information, resulting in a recovery of 40–60 percentage points in generalization performance. Although federated training has its benefits, its emergence is not without failure modes. We observe zero-dilution, where standard FedAvg blindly

averages structural zeros caused by stochastic dropout days with actual learned weights. This corrupts attention mechanisms and results in a centralized performance drop.

Our major contribution, FedMD, generates deterministic masks and mask-aware aggregation to eliminate zero-dilution. FedMD achieves the full representational power lost to dilution by assigning each client a unique but fixed dropout mask, and by using just the actively training parameters to carry out an element wise normalized average. Results show +1.05% (+1.05% on HAM (from 84.54% to 85.59%), +2.37% on ISIC (from 69.55% to 71.92%), and +9.90% on PAD (from 65.39% to 75.29%)) improvements in consistency. The system has F1-scores of 0.8626, 0.7301, and 0.6852 respectively. The striking recovery of PAD demonstrates that mask-awareness can be particularly valuable for clients facing severe domain shift, as naive aggregation would misappropriate their specialized features. Parameter coverage analysis suggests that this SE is almost fully trained already. Attention weight distributions recover their full dynamic range (0.1–0.9), restoring the type of diagnostic selectivity that zero-dilution had compressed to 0.4–0.6.

Essentially, FedMD introduces no additional communicational overhead. In addition, it incurs little computational cost, as only element wise operations are employed. Also, it does not require any architectural change and thus can be employed in existing federated frameworks immediately. An overall accuracy of 84.27% and mean F1 score of 0.726 closely approach centralized performance while ensuring strict data privacy as no raw patient image leaves its institution. Our results show that proper treatment of structural zeros induced by regularization allows attention-based architectures to work correctly in the distributed setting, which can unlock the full power of privacy-preserving collaboration between institutions.

This has great clinical implications. Modern day diagnostic AI systems that are trained at well resourced institutions cannot be generalized to a diverse segment. FedMD allows three diverse parties to collaboratively train a model that generalizes across their heterogeneous data distributions. Through the project, a dermatology clinic in Europe, a rural hospital which uses smartphone photography, and a multi institutional research consortium do this while preserving privacy and regulatory constraints. The mask-aware aggregation improved minority class detection melanoma recall of 81% on HAM, actinic keratosis recall of 92% on ISIC, shows that we do not overly dilute rare diagnostic features as would happen by naïve averaging.

The mask-aware aggregation technique of federated learning is not only an option over centralised training. In fact, it is the required approach for building clinically deployable medical imaging AI systems. FedMD tackles how stochastic regularization is incompatible with standard federated aggregation so that attention enhanced architectures found throughout medical imaging can benefit from multi institutional diversity without losing privacy, performance, or efficiency. The benefits go beyond skin cancer detection to any federated application using dropout-regularized attention mechanisms such as transformer-based medical imaging models, multi-modal diagnostic systems, and other privacy-sensitive healthcare analytics at distributed hospital networks.

Bibliography

[1] Andre Esteva et al. "Dermatologist-level classification of skin cancer with deep neural networks". In: *Nature* 542.7639 (2017), pp. 115–118.

[2] Joaquin Quiñonero-Candela et al., eds. *Dataset Shift in Machine Learning*. MIT Press, 2009.

[3] A. Developer. *Skin Cancer ISIC – The Skin Cancer Data*. https://www.kaggle.com/datasets/nodoubttome/skin-cancer9-classesisic. Accessed: 2025-10-04. 2020.

[4] Andre GC Pacheco et al. "PAD-UFES-20: A skin lesion dataset composed of patient data and clinical images collected from smartphones". In: *Data in brief* 32 (2020), p. 106221.

[5] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions". In: *Scientific Data* 5.1 (Aug. 2018). ISSN: 2052-4463. DOI: 10.1038/sdata.2018.161. URL: http://dx.doi.org/10.1038/sdata.2018.161.

[6] Ashish Vaswani et al. "Attention is all you need". In: *Advances in Neural Information Processing Systems*. Vol. 30. 2017.

[7] Jie Hu, Li Shen, and Gang Sun. "Squeeze-and-excitation networks". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 7132–7141.

[8] Mingxing Tan and Quoc V. Le. "EfficientNet: Rethinking model scaling for convolutional neural networks". In: *Proceedings of the 36th International Conference on Machine Learning*. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 6105–6114.

[9] Tauseef M. Alam et al. "An Efficient Deep Learning-Based Skin Cancer Classifier for an Imbalanced Dataset". In: *Diagnostics* 12.9 (2022), p. 2115. DOI: 10.3390/diagnostics12092115.

[10] Muhammad Azeem et al. "SkinLesNet: Classification of Skin Lesions and Detection of Melanoma Cancer Using a Novel Multi-Layer Deep Convolutional Neural Network". In: *Cancers* 16.1 (2023), p. 108. DOI: 10.3390/cancers16010108.

[11] Soufiane Barbaria et al. "Advancing compliance with HIPAA and GDPR in healthcare: a blockchain-based strategy for secure data exchange in clinical research involving private health information". In: *Healthcare* 13.20 (2025), p. 2594.

[12] H Brendan McMahan et al. "Communication-efficient learning of deep networks from decentralized data". In: *AISTATS*. 2017.

[13] Nitish Srivastava et al. “Dropout: a simple way to prevent neural networks from overfitting”. In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[14] Hyuna Sung et al. “Global cancer statistics 2020: GLOBOCAN estimates of incidence and mortality worldwide for 36 cancers in 185 countries”. In: *CA: a cancer journal for clinicians* 71.3 (2021), pp. 209–249.

[15] Tian Li et al. “Federated optimization in heterogeneous networks”. In: *MLSys*. 2020.

[16] Sai Praneeth Karimireddy and et al. “SCAFFOLD: Stochastic Controlled Averaging for Federated Learning”. In: *ICML*. 2020.

[17] Margaret Yeung et al. “Unified focal loss: Generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation”. In: *Computerized Medical Imaging and Graphics* 95 (2022), p. 102026.

[18] Peter Kairouz et al. “Advances and open problems in federated learning”. In: *Foundations and trends® in machine learning* 14.1–2 (2021), pp. 1–210.

[19] Kevin Hsieh et al. “The non-iid data quagmire of decentralized machine learning”. In: *International Conference on Machine Learning*. PMLR. 2020, pp. 4387–4398.

[20] Jianyu Wang and et al. “Tackling the objective inconsistency problem in heterogeneous federated optimization”. In: *arXiv preprint arXiv:2007.07481* (2020).

[21] Jonas Geiping et al. “Inverting gradients-how easy is it to break privacy in federated learning?” In: *Advances in neural information processing systems* 33 (2020), pp. 16937–16947.

[22] Milad Nasr, Reza Shokri, and Amir Houmansadr. “Comprehensive privacy analysis of deep learning”. In: *Proceedings of the 2019 IEEE Symposium on Security and Privacy (SP)*. Vol. 2018. 2018, pp. 1–15.

[23] Wenhao Yuan and Xuehe Wang. “FedAgg: Adaptive Federated Learning with Aggregated Gradients”. In: *arXiv preprint arXiv:2303.15799* (2023).

[24] Irene Wang, Prashant Nair, and Divya Mahajan. “Fluid: Mitigating stragglers in federated learning using invariant dropout”. In: *Advances in Neural Information Processing Systems* 36 (2023), pp. 73258–73273.

[25] Jiawei Shao et al. “Dres-fl: Dropout-resilient secure federated learning for non-iid clients via secret data sharing”. In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 10533–10545.

[26] Sixin Zhang, Anna E Choromanska, and Yann LeCun. “Deep learning with elastic averaging SGD”. In: *Advances in neural information processing systems* 28 (2015).

[27] Felix Sattler et al. “Robust and communication-efficient federated learning from non-iid data”. In: *IEEE transactions on neural networks and learning systems* 31.9 (2019), pp. 3400–3413.

[28] Filippos Efthymiadis et al. “Advanced Optimization Techniques for Federated Learning on Non-IID Data”. In: *Future Internet* 16.10 (2024), p. 370.

[29] Tian Li et al. “Feddane: A federated newton-type method”. In: *2019 53rd Asilomar Conference on Signals, Systems, and Computers*. IEEE. 2019, pp. 1227–1231.

[30] Jianyu Wang et al. “Slowmo: Improving communication-efficient distributed sgd with slow momentum”. In: *arXiv preprint arXiv:1910.00643* (2019).

[31] Feijie Wu et al. “On the convergence of quantized parallel restarted SGD for central server free distributed training”. In: *arXiv preprint arXiv:2004.09125* (2020).

[32] Tian Li et al. “Fair resource allocation in federated learning”. In: *arXiv preprint arXiv:1905.10497* (2019).

[33] Reent Schlegel et al. “CodedPaddedFL and CodedSecAgg: Straggler mitigation and secure aggregation in federated learning”. In: *IEEE Transactions on Communications* 71.4 (2023), pp. 2013–2027.

[34] Samuel Horvath et al. “Fjord: Fair and accurate federated learning under heterogeneous targets with ordered dropout”. In: *Advances in Neural Information Processing Systems* 34 (2021), pp. 12876–12889.

[35] Daniel J Beutel et al. “Flower: A friendly federated learning research framework”. In: *arXiv preprint arXiv:2007.14390* (2020).

[36] Zheng Chai et al. “FedAT: A high-performance and communication-efficient federated learning system with asynchronous tiers”. In: *Proceedings of the international conference for high performance computing, networking, storage and analysis*. 2021, pp. 1–16.

[37] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V. Le. “DropBlock: A regularization method for convolutional networks”. In: *Advances in Neural Information Processing Systems*. Vol. 31. 2018.

[38] Giuseppe Argenziano and H Peter Soyer. “Dermoscopy of pigmented skin lesions–a valuable tool for early”. In: *The lancet oncology* 2.7 (2001), pp. 443–449.

[39] Olga Russakovsky et al. “Imagenet large scale visual recognition challenge”. In: *International journal of computer vision* 115.3 (2015), pp. 211–252.

[40] Anastasia V Demidova, Dmitry S Kulyabov, and Eugeny Yu Shchetinin. “Skin cancer classification computer system development with deep learning.” In: *ITTMM*. 2020, pp. 57–69.

[41] Adria Romero Lopez et al. “Skin lesion classification from dermoscopic images using deep learning techniques”. In: *2017 13th IASTED international conference on biomedical engineering (BioMed)*. IEEE. 2017, pp. 49–54.

[42] Talha Mahboob Alam et al. “An efficient deep learning-based skin cancer classifier for an imbalanced dataset”. In: *Diagnostics* 12.9 (2022), p. 2115.

[43] Hatice Catal Reis et al. “InSiNet: a deep convolutional approach to skin cancer detection and segmentation”. In: *Medical & Biological Engineering & Computing* 60.3 (2022), pp. 643–662.

[44] Joachim Krois et al. “Generalizability of deep learning models for dental image analysis”. In: *Scientific reports* 11.1 (2021), p. 6102.

[45] Hao Wang and Jian Xu. "Combating client dropout in federated learning via friend model substitution". In: *arXiv preprint arXiv:2205.13222* (2022).

[46] Avishek Ghosh et al. "An efficient framework for clustered federated learning". In: *Advances in neural information processing systems* 33 (2020), pp. 19586–19597.

[47] Liangchen Luo et al. "Adaptive gradient methods with dynamic bound of learning rate". In: *arXiv preprint arXiv:1902.09843* (2019).

[48] Li Li et al. "FedSAE: A novel self-adaptive federated learning framework in heterogeneous systems". In: *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2021, pp. 1–10.

[49] Yu Zhang et al. "FedMDS: An efficient model discrepancy-aware semi-asynchronous clustered federated learning framework". In: *IEEE Transactions on Parallel and Distributed Systems* 34.3 (2023), pp. 1007–1019.

[50] Sashank Reddi et al. "Adaptive federated optimization". In: *arXiv preprint arXiv:2003.00295* (2020).

[51] Samiul Alam et al. "Fedrolex: Model-heterogeneous federated learning with rolling sub-model extraction". In: *Advances in neural information processing systems* 35 (2022), pp. 29677–29690.

[52] Tao Lin et al. "Ensemble distillation for robust model fusion in federated learning". In: *Advances in neural information processing systems* 33 (2020), pp. 2351–2363.

[53] M Yashwanth et al. "Adaptive Self-Distillation for Minimizing Client Drift in Heterogeneous Federated Learning". In: *arXiv preprint arXiv:2305.19600* (2023).

[54] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. "Distilling the knowledge in a neural network". In: *arXiv preprint arXiv:1503.02531* (2015).

[55] Antonio Orvieto, Simon Lacoste-Julien, and Nicolas Loizou. "Dynamics of sgd with stochastic polyak stepsizes: Truly adaptive variants and convergence to exact solution". In: *Advances in Neural Information Processing Systems* 35 (2022), pp. 26943–26954.

[56] Fabian Schaipp, Robert M Gower, and Michael Ulbrich. "A stochastic proximal polyak step size". In: *arXiv preprint arXiv:2301.04935* (2023).

[57] Junha Park et al. "Federated learning for indoor localization via model reliability with dropout". In: *IEEE Communications Letters* 26.7 (2022), pp. 1553–1557.

[58] Simon Yiu et al. "Wireless RSSI fingerprinting localization". In: *Signal Processing* 131 (2017), pp. 235–244.

[59] Gibrán Félix, Mario Siller, and Ernesto Navarro Alvarez. "A fingerprinting indoor localization algorithm based deep learning". In: *2016 eighth international conference on ubiquitous and future networks (ICUFN)*. IEEE. 2016, pp. 1006–1011.

[60] Peng Wu et al. "Personalized federated learning over non-iid data for indoor localization". In: *2021 IEEE 22nd International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. IEEE. 2021, pp. 421–425.

[61] Charles Blundell et al. “Weight uncertainty in neural network”. In: *International conference on machine learning*. PMLR. 2015, pp. 1613–1622.

[62] Yarin Gal and Zoubin Ghahramani. “Dropout as a bayesian approximation: Representing model uncertainty in deep learning”. In: *international conference on machine learning*. PMLR. 2016, pp. 1050–1059.

[63] Eyke Hüllermeier and Willem Waegeman. “Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods”. In: *Machine learning* 110.3 (2021), pp. 457–506.

[64] Khadija Shahzad et al. “Federated Convolutional Neural Networks (F-CNNs) for privacy-preserving multi-class skin lesion classification”. In: *Array* (2025), p. 100667.

[65] Faiqa Adnan et al. “EfficientNetB3-Adaptive Augmented Deep Learning (AADL) for Multi-Class Plant Disease Classification”. In: *IEEE Access* 11 (2023), pp. 85426–85440. DOI: 10.1109/ACCESS.2023.3303131.

[66] Ilya Loshchilov and Frank Hutter. “Decoupled weight decay regularization”. In: *arXiv preprint arXiv:1711.05101* (2017).