Dermatological Image Classification

Shubham Saha
Chalmers University
of Technology
Gothenburg, Sweden
shubhams@chalmers.se

Sifat Nawrin Nova
Chalmers University
of Technology
Gothenburg, Sweden
sifatn@chalmers.se

Sudeshna Lama
Chalmers University
of Technology
Gothenburg, Sweden
sudeshna@chalmers.se

Abstract

This project aims to develop an image classification system to classify skin lesions as either melanoma or nevus using a subset of the 2018 ISIC challenge dataset. We explore the impact of various techniques, including normalization, residual connections, data augmentation, and transfer learning, on the performance of our models. The ultimate goal is to contribute to early skin cancer detection efforts, potentially improving patient outcomes through the application of advanced ML techniques. After rigorous training through the dataset, we achieved 0.94 accuracy with the VGG16 pre-trained model.

1 Introduction

Skin cancer, especially melanoma, is a critical public health concern if not treated before it metastasizes (spreads to other parts of the body). Its potential severity increases if it is not detected early. Our project focused on developing a machine learning system that distinguishes between melanoma and benign nevus from dermatological images. We utilized the VGG16 model, known for its effectiveness in image classification, and applied advanced techniques such as data augmentation and normalization to improve accuracy.

The dataset used in this project was a subset of the 2018 ISIC challenge(C1, 2018) dataset and only two types: melanoma and melanocytic nevus was classified. Our classification models include CNN along with normalization, and some pre-trained models like ResNet and VGG16. Common evaluation metrics like ROC AUC, accuracy, precision, and recall are used to analyze model performance.

2 Method

2.1 Data Collection

The dataset used in this project was derived from the 2018 ISIC challenge which is a processed subset which includes two primary classes: melanoma (MEL) and melanocytic nevus (NV).

Data augmentation play a pivotal role in the preparation of the dataset for training and validating our machine learning models to standardize the input data, introduce variability, enhancing the model's ability to generalize from the training data to unseen data. The augmentation is implemented using PyTorch's v2 transformation library.

2.2 Data Augmentation

For the training dataset, a series of transformations were applied to augment the data and prepare it for the training process. This includes:

- conversion to Python Imaging Library(PIL) image format for the compatibility of subsequent transformations.
- random resized crop on the image to a fixed dimension of 224x224 pixels to introduce variability in the scale and aspect ratio of the images, simulating different viewing angles and distances for improved quality of the resized images.
- random horizontal flip with a certain probability that mimics the natural variability in the orientation of skin lesions to improve the model's robustness to orientation changes.
- converting the image to a tensor of type torch.float32 and scaling the pixel values to a [0, 1] range to normalize pixel values which facilitates model training by providing a consistent scale for input features.
- normalizing pixel values using calculated mean and standard deviation values across

each channel (RGB) based on the dataset. This step ensures that the input data has a mean of 0 and a standard deviation of 1, which helps stabilize the learning process by standardizing the distribution of input features.

For the validation dataset, it follows a similar but slightly different procedure which is designed to evaluate the model's performance without the variability introduced by augmentation.

2.3 Data Representation

To represent the data effectively for machine learning purposes, images were converted into tensors using PyTorch's ImageFolder and DataLoader. This conversion simplifies the handling of image data and makes it more efficient and scalable. It facilitates the utilization of the dataset images in neural network models by ensuring they are in the correct format and batched appropriately for efficient training.

3 Model Training and Analysis

3.1 Selection of the Model

For selecting the best model we explored several architectures, including basic CNN and pre-trained models like ResNet and VGG16 along with some procedural modifications. This process began with the implementation of a basic CNN model before applying data augmentation which we considered as our baseline for further comparison. After that we applied data augmentation we have incorporated different normalization methods such as batch, layer and group into the basic CNN model. Eventually, we transitioned to pre-trained models, specifically ResNet and VGG16, to further improve our classification performance.

Basic CNN Model We designed the basic CNN model to capture the complex patterns and features in skin lesion images. This model comprised of several convolutional layers to extract features, followed by pooling layers to reduce dimensionality, and fully connected layers to perform the classification. The results of the basic CNN served as our baseline for comparison with more advanced techniques.

Incorporating Normalization Techniques After data augmentation, we experimented with three distinct normalization techniques applied to our basic CNN architecture: batch, layer, and group normalization. Each of these techniques was tested individually to assess its impact on model performance:

- Batch Normalization: By normalizing the input of each layer across each mini-batch it aimed to stabilize the learning process and improve convergence rates.
- Layer Normalization: By normalizing the inputs across all features within a layer, this technique makes it less dependent on the batch size and potentially more suitable for our dataset.
- Group Normalization: This technique divides the channels into groups and computes within each group the mean and variance for normalization. This approach is beneficial for tasks with smaller batch sizes or when the batch normalization's effectiveness is limited.

Leveraging Pre-trained Models Finally we explored the use of pre-trained models, specifically ResNet and VGG16. These models have been trained on large datasets which offers a profound base of learned features that can be fine-tuned for specific tasks. By adapting these pre-trained models to our dataset, we aimed to leverage their powerful feature extraction capabilities, expecting a significant boost in classification performance.

- ResNet: Known for its residual connections that facilitate the training of very deep networks by alleviating the vanishing gradient problem.
- VGG16: Characterized by its simplicity and depth, focusing on having a deep network of convolutional layers to capture detailed features.

The transition to pre-trained models enhances the model seclection process. The selection criteria for the final model were based on its performance on the validation set, with a particular focus on achieving a balance between accuracy and computational efficiency.

3.2 Evaluation

To thoroughly assess the quality of our classification system, we used several key metrics like AUC (area under the ROC curve), Accuracy test on unseen test data, Precision, and Recall. However, we have used AUC as our primary evaluation metric.

3.3 AUC - ROC Curve

The ROC curve is a graphical plot that illustrates the diagnostic ability of a binary classifier system. Its discrimination threshold is varied. The AUC - the area under the ROC curve - provides a single scalar value to evaluate the model's ability to discriminate between positive and negative classes across different thresholds. An AUC which is closer to 1 indicates a better performing model, whereas an AUC closer to 0.5 suggests no discriminative power.

3.4 Accuracy

We also computed the accuracy to quantify the overall rate of correct predictions made by our model. While accuracy is a straightforward metric. Validation accuracy measures the percentage of correct predictions out of all predictions made. It is dependent on the threshold chosen to classify probabilities into different classes.

3.5 Precision and Recall

Precision measures the accuracy of positive predictions made by the classifier. However, Recall, also known as sensitivity, measures the model's ability to identify all actual positives. These metrics provide a deeper understanding of the model's performance, where false negatives or false positives can have different implications.

4 Results

After training all the model we have come to a conclusion that the VGG16 model holds the maximum AUC ROC score of 0.94. The robustness of the model was quantified through several performance metrics. This model was tested with unseen dataset to test its capability in classifying dermatological images into melanoma and melanocytic nevus categories.

4.1 Validation Accuracy

VGG16 model's validation performance was impressive, with a final accuracy of 85.38%, as indicated in the comparison bar graph. This high level of accuracy suggests that VGG16's deep convolutional neural network is benefiting from pre-training on a vast array of validation images. All the models are evaluated based on a baseline which was a basic CNN model. VGG16 turns out to be well-suited to the task of skin lesion classification across all the other models.

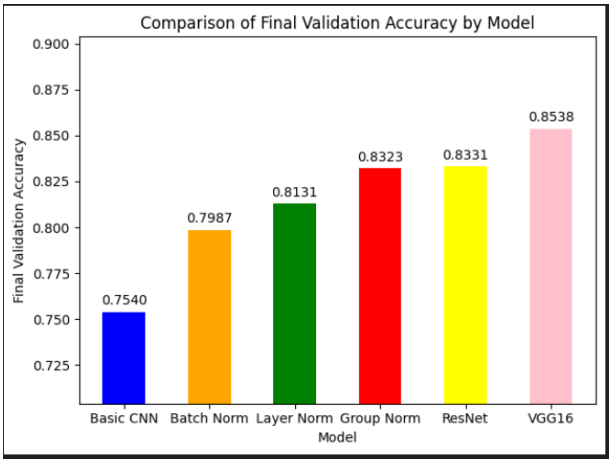

Figure 1: Comparison of Final Validation Accuracy by Model

4.2 Confusion Matrix

The analysis of the confusion matrix revealed that the model correctly identified 534 instances of melanocytic nevus (NV) and 635 instances of melanoma (MEL). There were 149 cases which where NV and was mistakenly classified as MEL and 48 cases where MEL was misclassified as NV. While the false negatives (MEL classified as NV) are of particular concern in medical diagnostics due to the potential for missed melanoma diagnoses. This relatively low number suggests that the model has learned to error on the side of caution, favoring false alarms over missed detections.

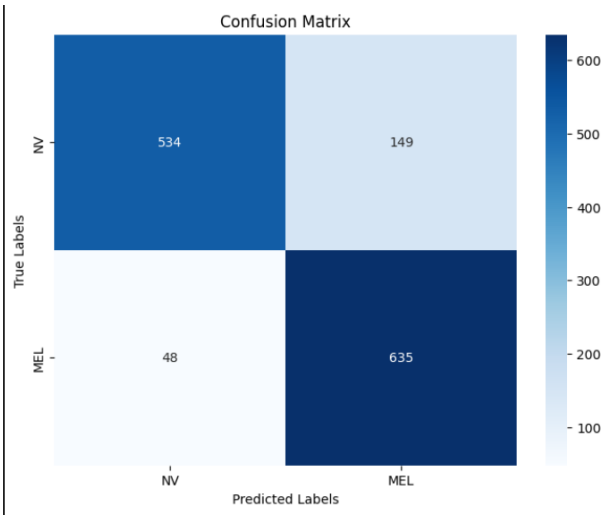

Figure 2: Confusion Matrix by VGG16 Model

4.3 Test Accuracy

Calculating the prediction results, the model achieved a test accuracy of 85.58%, demonstrating excellent consistency. This close alignment between validation and test accuracies indicates that the model performs reliably on unseen data and validates the model's potential for real-world application.

4.4 ROC Curve

The Receiver Operating Characteristic (ROC) curve further ensured the model's diagnostic ability, with an Area Under Curve (AUC) score of 0.94. This was our primary evaluation matrix. This outstanding score implies that the model has a high true positive rate across various threshold levels, which is vital for medical diagnosis applications where the cost of false negatives is high.

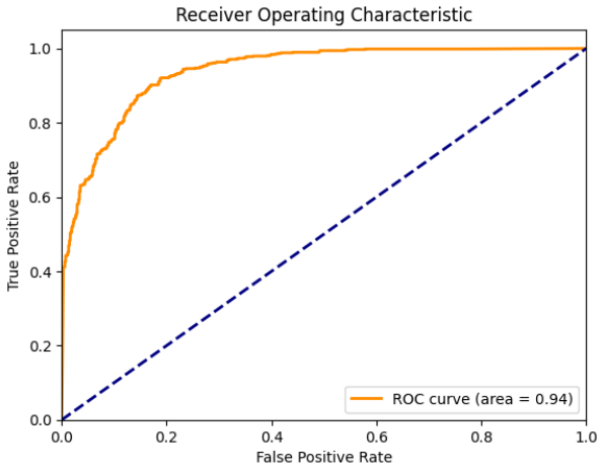

Figure 3: ROC AUC comparison

5 Limitation

One of the primary limitations we encountered during the development process was the lack of high-performance computing resources. This constraint significantly affected the model training which results in prolonged durations to achieve expected results and iterate over model improvements. Due to the limited computational power of our personal computers, each training epoch extended over considerable periods. It made the debugging and testing process time-consuming and cumbersome. In several instances, issues that surfaced deep into the training sessions necessitated restarts, leading to further delays. Which led to minimal experiments in hyperparameter tuning.

In terms of real-world application, the usability of the system we developed in a clinical setting would require rigorous validation against a broader array of dermatological conditions and would need to be contextualized within the diverse spectrum of skin lesions encountered in clinical practice.

6 Ethical Considerations

Deploying our skin cancer detection system in real life must be done carefully by keeping adherence to data privacy laws, such as GDPR, and the AI Act. We need to ensure people's health data is safe and they know how it is used. The system should also be fair to everyone, regardless of their skin type, and we must avoid biases that could harm some individuals while helping others. It is also important to make clear that our tool is for extra help, not to replace doctors. We don't want people to wait too long for a real checkup if they think something is wrong. Lastly, we must be sensitive to how people might feel using our system and we don't want to scare them with false alarms or give them false hope. The system should guide users gently, reducing worry and encouraging them to see a doctor when needed.

7 Conclusion

Our project of image classification between melanoma and nevus concluded with a high AUC score of 0.94 and an impressive accuracy using VGG16 model, showcasing our model's ability to effectively differentiate images. However, with high-end computing resources with better infrastructure, the system could be significantly improved. Also addressing the ethical and practical implications of this technology, particularly in ensuring adherence to privacy laws the system is envisioned to support, not substitute, professional healthcare advice. With enhanced computational power and a focus on ethical integration, our goal will be to make this tool a valuable asset in healthcare, aiding early detection and promoting positive patient outcomes.

References

2018. https://challenge.isic-archive.com/landing/2019/ Isic 2018 challenge.