Performance evaluation of various CNN-based architectures on automated skin-lesion diagnosis

Balavignesh Vemparala Narayana Murthy
Department of Mechanical and Aerospace Engineering, The Ohio State University, USA

Abstract

Deep Learning (DL) models have been used in the recent past in automating skin lesion diagnosis, which has the potential to assist clinicians before diagnosis, to identify the severity of the disease. However, for the use of DL models to become mainstream in the medical community, high levels of accuracy and fairness is necessary. But, the datasets used for these purposes are generally biased, underrepresenting certain class labels. Thus, the DL models trained on these unbalanced datasets also learn these inherent biases already present in the data. In the current study, we compare several DL architectures, including plain Convolutional Neural Network (CNN), CNN with residual blocks, and CNN with inception blocks, on their performance for skin-lesion diagnosis. Moreover, we also use two different datasets – HAM10000, Diverse Dermatology Images (DDI), train the models on the former and test on both the test set from HAM10000, as well as on DDI, which shows how the models generalize well. Moreover, in order to address the imbalance in datasets, we also use several oversampling methods, and study the effect on performance. Following this, we identify the best DL architecture, and perform an ablation study to understand which model components contribute the most to performance.

1 Introduction

These days, Artificial Intelligence (AI) techniques, including Deep Learning, are increasingly used in medical image analysis to assist clinicians in diagnosis. Despite their success, DL models are susceptible to biases present in the training data, which make them unreliable. For instance, it is a well-known fact that predictions on skin lesion images with darker skin tones has a significant drop in accuracy compared to light tones. This has often been attributed to the inherent bias present in the datasets used to train these models (Yuan et al 2022, Wu et al 2022, Du et al 2022, Rezk et al 2022). Similarly, it is also important for these datasets to have equal representation of different classes (for instance, benign and malignant lesions), as the ultimate goal is to classify a given skin lesion into either of these labels. However, it is often the case that the datasets over-represent one of the labels, which introduces another type of bias. Moreover, there is a dearth of detailed studies on model generalization in this field. That is, training and testing a model on a single dataset may not be sufficient, since we want to make sure that the models are generalizable with similar accuracy on a wide range of population. Hence, it is also important to use multiple datasets to ensure model generalizability.

In this study, we try to address some of these above-mentioned issues by first performing a thorough comparison between several DL architectures – Plain CNN, CNN with residual blocks (res-CNN), CNN with inception blocks (incept-CNN). We use Bayesian Optimization to tune the hyperparameters for each of these architectures, and then identify the best performing architecture. Moreover, we also use multiple datasets – HAM10000 (Tschandl el al 2018), DDI (Daneshjou, et al 2022), which helps check for model generalization as well. Following this, we also study several oversampling methods to see how they're able to remove the bias present in the datasets. Finally, we also perform an ablation study on the best performing architecture, which

can help identify the most important model components and reduce unnecessary model complexity.

2 Related Work

There is a lot of research done in the literature on AI-assisted skin lesion diagnosis. Lopez-Labraca et al 2022 proposed a CNN-based Computer-aided melanoma diagnosis system with dermoscopic images, which helps with extracting the most relevant lesion regions. However, it doesn't say much about whether the lesion is benign or malignant, which is important. Ahmed et. al 2020 proposed a DL method for skin lesion classification using Deep CNN ensembles, making use of pre-trained XCeption, Inception-ResNet-V2 and NasNetLarge architectures. However, they donot study how well their model generalizes to other datasets, which is really important. Jayalakshmi et al 2019 propose a custom CNN model using Batch Normalization for skin-lesion classification, however, the motive for choosing the final architecture is not clear. Pham et al 2018 propose a Deep CNN combined with data augmentation to improve the classification accuracy. However, it uses traditional flip and rotate transformations to augment the training data, which doesn't introduce any new information to the data sets. Goceri et al 2020 compare several pre-trained CNN-based networks such as VGG16/19, GoogleNet, ResNet101, InceptionV3 on the task of skin lesion diagnosis. However, they just compare pre-trained networks blindly, which doesn't do much good.

Thus, it can be clearly seen that comprehensive comparison of several different CNN-based model architectures hasn't been performed in literature. Moreover, the justification as to why a particular architecture has been adopted hasn't been made clear as well. In addition to this, studies in literature haven't made use of oversampling methods to augment the training dataset, which unlike traditional flip and rotate transformations, introduces some new information to the dataset. Therefore, it becomes really important to carry out a comprehensive study addressing these issues.

Moreover, literature also discusses about the unbalanced skin-lesion datasets, which is also of concern (Yuan et. al 2022, Wu et. al 2022). Thus, it is necessary to address this in order to build "fair" AI models using these datasets.

3 Methodology

3.1 Data pre-processing

As mentioned previously, multiple datasets have been used in this study – HAM10000, DDI. HAM10000 consists of 10015 images comprising of several skin conditions, including both benign and malicious lesions, whereas DDI dataset consists of only two truth labels (benign or malignant). Since both datasets have different labels, it is necessary to first group the skin conditions in HAM10000 dataset into benign and malignant conditions, in order to train and test the same DL models using both datasets. Out of the several skin conditions covered in the dataset, benign keratosis-like lesions (bkl) and dermatofibroma (df) were considered to be benign (assigned a truth label of '0'), whereas Bowen's disease (akiec), basal cell carcinoma (bcc) and melanoma (mel) were taken to be malignant (assigned a truth label of '1'). Other skin conditions such as melanocytic nevi (nv) and vascular lesions (vasc) were neglected since they can be both benign and malignant, depending on a case by case basis.

Moreover, it was observed that the image resolutions in HAM10000 dataset were 600×450, whereas DDI dataset consisted of images with different resolutions. So, it is necessary to make sure all images are of the same resolution in order to use them. First, all images in the DDI dataset were cropped to 450×450. Similarly, it was not possible to crop the images from HAM10000 to 450×450, as some of the images had a lesser width than 450 pixels. Thus, we tried to crop all images to 250×250. However, using this resolution lead to Out Of Memory (OOM) errors during the training process, due to insufficient memory to store large arrays resulting from using large images. Thus, we downsampled all images to a resolution of 100×100, which removed the memory issues faced during training process.

3.2 Oversampling methods

It was observed that the reduced HAM10000 dataset obtained from pre-processing was

unbalanced, and hence, it was necessary to perform some form of data-augmentation to balance the dataset. Traditional data augmentation methods such as flip and rotate transformations do not introduce any new information to the already existing dataset. Hence, we used three different oversampling methods, provided by 'imbalanced learn' sub-package of 'sklearn' – random, SMOTE, and ADASYN to oversample the minority labels.

Naïve random oversampling is the first approach in which minority classes are oversampled by picking samples at random with replacement. Apart from this, there are two other popular methods to over-sample minority classes – the Synthetic Minority Oversampling Technique (SMOTE) & the Adaptive Synthetic (ADASYN) sampling methods.

In SMOTE (Chawla et al 2002), first the minority class is identified. Following this, k-nearest neighbors are identified for each of the minority samples, and straight lines are drawn between the neighbors to generate random samples along these lines, as illustrated in Figure 1 below.

On the other hand, the ADASYN over-sampling technique (He et al 2008) is an improved version of SMOTE, in which small random values are added to the over-sampled points to make it more realistic. In other words, instead of all the over-sampled points to be linearly correlated with their parent points, small random values are added to introduce some variance to the over-sampled dataset.

Moreover, it is important to note that oversampling should be applied only on the training data set and not the test set. This is because, we want to test our model on real data, to ensure that the model performance is indeed on real data, and not on the synthetic data. That is, the motive of using synthetic data should solely be to improve the model performance on real data.

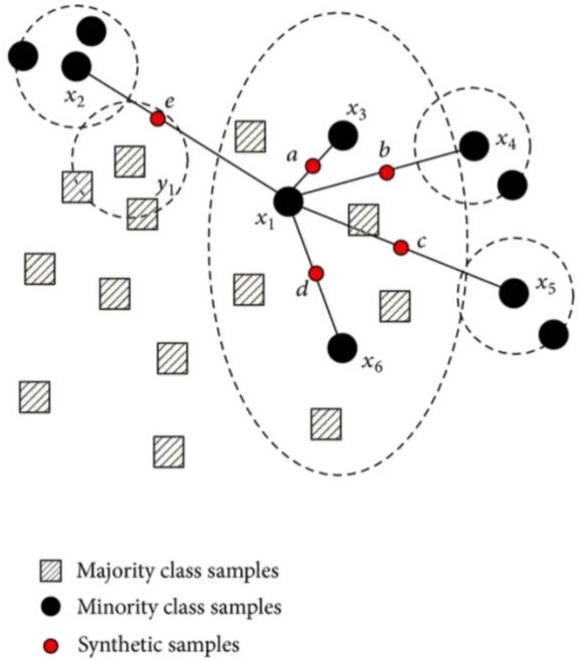

Figure 1: Schematic illustrating SMOTE oversampling technique (Source - https://medium.com/coinmonks/smote-and-adasyn-handling-imbalanced-data-set-34f5223e167)

3.3 Bayesian Optimization

Bayesian Optimization (Frazier 2018) is an approach that uses Bayes theorem to direct the search in an optimization process. For each of the model architectures tested in this project, Bayesian Optimization was used to optimize the hyperparameters. For instance, Figure 2 shows the plain CNN model. The hyperparameters here are the number of convolutional blocks, number of dense blocks, number of convolutional filters in Conv2D layer, number of dense units in each Dense layer. Apart from this, batch size and learning rate were also set as hyperparameters with ADAM optimizer.

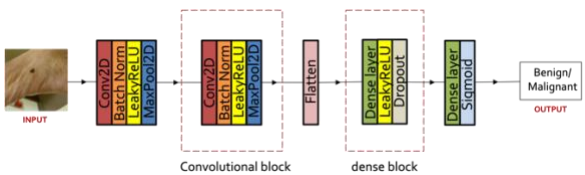

Figure 2: Plain Convolutional Neural Network (CNN) architecture

Similarly, Figure 3 shows the model architecture for CNN with residual blocks. The new hyperparameters here are the number of residual

blocks as well as the number of residual convolution layers, in addition to the hyperparameters in plain CNN architecture.

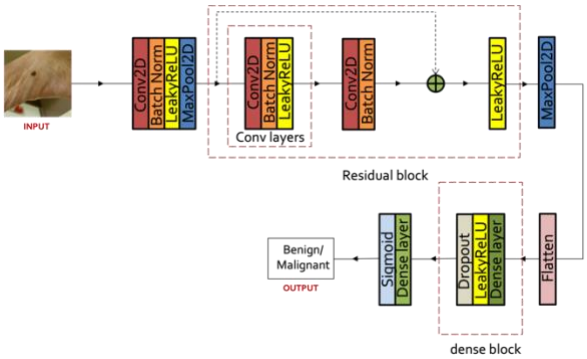

Figure 3: CNN with residual blocks (res-CNN) architecture

Next, Figure 4 shows the architecture of an inception block, proposed by Szegedy et. al 2017, in which convolution operations of different filter sizes $1 \times 1, 3 \times 3$ & 5×5 as well as 3×3 max pooling are performed. We then use the inception block in the incept CNN architecture shown in Figure 5. In this, the new hyperparameter is the number of inception blocks.

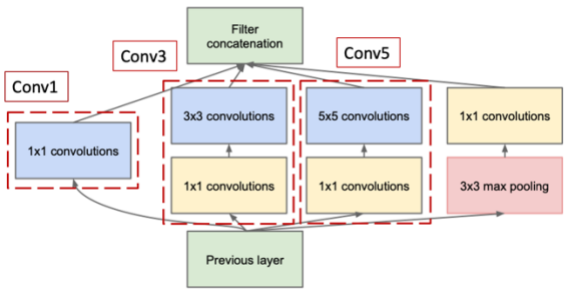

Figure 4: Inception block (Szegedy et. al 2017)

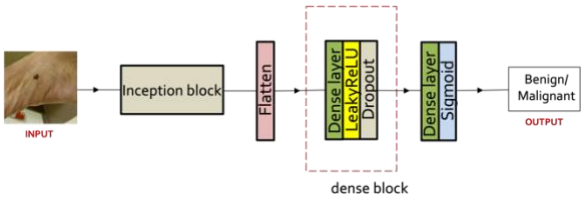

Figure 5: CNN with inception blocks (incept-CNN) architecture

Next, we report the hyperparameters used in different DL architectures tested in this project in Tables 1, 2 and 3. We also provide the range of values tested for these hyperparameters, and also report the optimal value found using Bayesian Optimization.

Hyperparameter	Range	Optimal value
No. of Conv blocks	[1,5]	1
No. of Conv layers in a Conv block	[1,4]	3
No. of Conv filters	[16,128]	128
No. of dense blocks	[1,3]	3
No. of dense units	[16,128]	64
Dropout ratio	[0.0,0.5]	0.0

Table 1: Hyperparameters in plain CNN model

Hyperparameter	Range	Optimal value
No. of Conv filters	[16,128]	64
No. of res-block Conv layers	[1,3]	2
No. of residual blocks	[1,5]	4
No. of dense blocks	[1,3]	3
No. of dense units	[16,128]	80
Dropout ratio	[0.0,0.5]	0.0

Table 2: Hyperparameters in res-CNN model

Hyperparameter	Range	Optimal value
No. of Conv filters	[16,48]	48
No. of inception blocks	[1,5]	5
No. of dense blocks	[1,3]	1
No. of dense units	[16,48]	16
Dropout ratio	[0.0,0.5]	0.5

Table 3: Hyperparameters in incept-CNN model

3.4 Model Training

All models were trained on a single CPU

3 Evaluation

4.1 Overall Performance

Of all the different architectures, it was observed that incept-CNN architecture has the best accuracy and also generalizes the best to the DDI dataset, after being trained using HAM10000 dataset. Also, note that incept-CNN architecture is the most heavy in terms of number of parameters as well as the training time.

The model parameters/training times as well as the relative performance of the optimized versions of the three architectures studied in this work are presented in Tables 4 & 5 below.

Model	Number of parameters	Training time
Best CNN	785,793	0.71hrs
Best res-CNN	3,661,377	1.74hrs
Best incept-CNN	31,376,673	9.6 hrs

Table 4: Number of parameters/training time for different architectures

Model	HAM10000 test accuracy (%)	DDI dataset accuracy (%)
Best CNN	70.82%	40.85%
Best res-CNN	67.98%	40.55%
Best incept-CNN	71.61%	57.77%

Table 5: Performance of different architectures on HAM10000 & DDI datasets

4.2 Comparison of Oversampling methods

Following this, we report the effect of oversampling methods on the accuracy. Table 6

below shows how oversampling balances the datasets. As it can be seen, oversampling leads to almost equal representation of the two labels (benign/malignant) in the HAM10000 dataset.

Oversampling Method	Number of Ones	Total samples	Percentage (%)
No oversampling	1564	2534	60.54%
Random	1564	3128	49.04%
SMOTE	1564	3128	49.04%
ADASYN	1564	3104	49.42%

Table 6: Effect of oversampling on distribution of dataset

Next, we combine the various oversampling methods with the optimized CNN, res-CNN and incept-CNN architectures, and study their effect on accuracy.

As it can be seen from Table 7 below, random oversampling performs the best both in terms of generalization as well as accuracy with the optimized CNN architecture, followed by SMOTE oversampling. In comparison, as can be observed from Table 8, random oversampling only shows the best generalization, followed by SMOTE. However, in case of incept-CNN architecture, SMOTE and ADASYN oversampling methods perform better than random oversampling.

The reasons for these observations can be multi-fold. First, naïve random oversampling is random in nature, and hence, is not guaranteed to yield the same behavior in different trials. However, SMOTE and ADASYN sampling methods are more or less deterministic in nature, except for the fact that small noise is added in ADASYN, which could also be the reason for the relatively worse performance of ADASYN in comparison with SMOTE in cases of both plain CNN as well as res-CNN architectures. Even though the addition of noise in ADASYN brings the oversampled data closer to the real-world, this by itself could've made it harder for the models to learn properly. Thus, we conclude that SMOTE oversampling method has the most consistent performance across all the models.

Model	HAM10000 test accuracy (%)	DDI dataset accuracy (%)
Best CNN	70.82%	40.85%
Best CNN + Random Oversampling	72.40%	49.24%
Best CNN + SMOTE Oversampling	69.56%	46.65%
Best CNN + ADASYN Oversampling	67.51%	45.58%

Table 7: Effect of oversampling on accuracy of optimized CNN architecture

Model	HAM10000 test accuracy (%)	DDI dataset accuracy (%)
Best res-CNN	67.98%	40.55%
Best res-CNN + Random Oversampling	64.03%	47.71%
Best res-CNN + SMOTE Oversampling	69.24%	42.38%
Best res-CNN + ADASYN Oversampling	68.14%	32.01%

Table 8: Effect of oversampling on accuracy of optimized res-CNN architecture

Model	HAM10000 test accuracy (%)	DDI dataset accuracy (%)
Best incept-CNN	71.61%	57.77%
Best incept-CNN + Random Oversampling	74.29%	53.51%
Best incept-CNN + SMOTE Oversampling	74.60%	58.69%
Best incept-CNN + ADASYN Oversampling	73.97%	61.89%

Table 9: Effect of oversampling on accuracy of optimized incept-CNN architecture

4.3 Incept-CNN architecture ablation study

Next, since incept-CNN architecture has been found to be the best in terms of performance for the skin-lesion diagnosis problem in this study, we wanted to study the relative contributions of the different Convolutions in the inception block towards its performance. The different convolution blocks namely, Conv1, Conv3, Conv5, labeled in Figure 4, needn't contribute equally towards the performance of the incept-CNN architecture, and hence, we wanted to study this further.

In order to do this, we removed each of the Convolutions, one at a time and studied its effect, as shown in Tables 10 & 11 below. As it can be seen, Conv1 seems to be the most important convolution operation, followed by Conv3, as removing it leads to significantly lesser accuracy, as can be seen from Table 11. Moreover, Conv1 is the cheapest convolution operation of all and doesn't capture spatial relationships between pixels. Nevertheless, it seems to be the most important for the performance in skin-lesion diagnosis task, which says that probably for the skin-lesion diagnosis task, spatial relationships are probably not that important.

Model	Number of parameters	Training time
Best incept-CNN	31,376,673	9.6 hrs
Best incept-CNN, no Conv1	23,615,601	8 hrs
Best incept-CNN, no Conv3	23,490,897	6.86 hrs
Best incept-CNN, no Conv5	23,269,713	6.09 hrs

Table 10: Number of parameters/training time for different incept-CNN architectures

Model	HAM10000 test accuracy (%)	DDI dataset accuracy (%)
Best incept-CNN	71.61%	57.77%
Best incept-CNN, no Conv1	69.40%	44.36%
Best incept-CNN, no Conv3	75.08%	49.70%
Best incept-CNN, no Conv5	72.24%	53.81%

Table 11: Performance of different incept-CNN architectures on HAM10000 & DDI datasets

4 Conclusions

Skin-lesion diagnosis is important as it can assist clinicians prior to diagnosis. Deep Learning models have been used in the recent past in automating this task. Nevertheless, the datasets used in the field have been found to be biased. Moreover, DL models have also struggled with generalization, as the models don't show good performance on new datasets. In this study, we compared three different CNN-based DL architectures on the task of skin-lesion diagnosis and found that CNN architecture with inception blocks (or incept-CNN) shows the best performance. Moreover, we also study the effect of oversampling methods and found that oversampling in general, improves accuracy and generalization, and the SMOTE oversampling approach shows the most consistent performance. Following this, we also performed an ablation study on the incept-CNN architecture and found that 1×1 convolution operation contributes the most to accuracy, which raises an important question – "Isn't spatial relationships using 3×3 and 5×5 convolutions supposed to improve model performance?"

Future work entails addressing other biases reported in literature in DL-assisted skin-lesion diagnosis, such as racial and gender bias (Yuan et. al, Wu et. al). In order to address racial/gender bias using oversampling, in addition to generating the truth labels (benign/malignant), we also need to generate the race/gender of the new samples. Hence, we can use MLSMOTE (Charte et al 2015), which is a multilabel variant of SMOTE, to augment the data in such cases. In addition to this, we could also use CycleGANs (Zhu et al 2017) to generate synthetic data of minority samples. Moreover, we also need to test our models on other important datasets in this field, such as the Fitzpatrick17k dataset, since we want to ensure that our models generalize well.

References

Lopez-Labraca, J., Gonzalez-Diaz, I., Diaz-de-Maria, F. and Fueyo-Casado, A., 2022. An interpretable CNN-based CAD system for skin lesion diagnosis. *Artificial Intelligence in Medicine*, *132*, p.102370.

Ahmed, S.A.A., Yanikoğlu, B., Göksu, Ö. and Aptoula, E., 2020, October. Skin lesion classification with deep CNN ensembles. In *2020 28th Signal Processing and Communications Applications Conference (SIU)* (pp. 1-4). IEEE.

Jayalakshmi, G.S. and Kumar, V.S., 2019, February. Performance analysis of convolutional neural network (CNN) based cancerous skin lesion detection system. In *2019 International Conference on Computational Intelligence in Data Science (ICCIDS)* (pp. 1-6). IEEE.

Pham, T.C., Luong, C.M., Visani, M. and Hoang, V.D., 2018, March. Deep CNN and data augmentation for skin lesion classification. In *Asian Conference on Intelligent Information and Database Systems* (pp. 573-582). Springer, Cham.

Goceri, E. and Karakas, A.A., 2020, July. Comparative evaluations of cnn based networks for skin lesion classification. In *14th International conference on computer graphics. visualization, computer vision and image processing (CGVCVIP), Zagreb, Croatia* (pp. 1-6).

Fitzpatrick17k,https://github.com/mattgroh/fitzpatrick17k

Tschandl, P., Rosendahl, C. and Kittler, H., 2018. The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, *5*(1), pp.1-9.

Daneshjou, R., Vodrahalli, K., Novoa, R.A., Jenkins, M., Liang, W., Rotemberg, V., Ko, J., Swetter, S.M., Bailey, E.E., Gevaert, O. and Mukherjee, P., 2022. Disparities in dermatology AI performance on a diverse, curated clinical image set. *Science advances*, *8*(31), p.eabq6147.

Oversampling methods, https://imbalanced-learn.org/stable/over_sampling.html

Chawla, N.V., Bowyer, K.W., Hall, L.O. and Kegelmeyer, W.P., 2002. SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, pp.321-357.

He, H., Bai, Y., Garcia, E.A. and Li, S., 2008, June. ADASYN: Adaptive synthetic sampling approach for imbalanced learning. In *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*(pp. 1322-1328). IEEE.

Frazier, P.I., 2018. Bayesian optimization. In *Recent advances in optimization and modeling of contemporary problems* (pp. 255-278). Informs.

Szegedy, C., Ioffe, S., Vanhoucke, V. and Alemi, A.A., 2017, February. Inception-v4, inception-resnet and

the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*.

Yuan, H., Hadzic, A., Paul, W., de Flores, D.V., Mathew, P., Aucott, J., Cao, Y. and Burlina, P., 2022. EdgeMixup: Improving Fairness for Skin Disease Classification and Segmentation. arXiv preprint arXiv:2202.13883.

Wu, Y., Zeng, D., Xu, X., Shi, Y. and Hu, J., 2022. FairPrune: Achieving Fairness Through Pruning for Dermatological Disease Diagnosis. arXiv preprint arXiv:2203.02110.

Charte, F., Rivera, A.J., del Jesus, M.J. and Herrera, F., 2015. MLSMOTE: Approaching imbalanced multilabel learning through synthetic instance generation. *Knowledge-Based Systems*, *89*, pp.385-397.

Zhu, J.Y., Park, T., Isola, P. and Efros, A.A., 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision* (pp. 2223-2232).