SKIN CANCER SELF-DIAGNOSIS USING MOBILE DEVICE DERMATOSCOPIC ATTACHMENTS

(Computer Vision; Healthcare)

Chafik Taiebennefs
tchafik@stanford.edu; SUNet ID: tchafik

PROJECT GOAL

Develop a mobile device optimized ML model for the self-diagnosis of skin cancer using mobile device dermatoscopic attachments.

PROBLEM STATEMENT

Skin cancer is the most common form of cancer in the United States, with the annual cost of care exceeding $8 billion. With early detection, the 5-year survival rate of the deadliest form, melanoma, can be up to 99%; however, delayed diagnosis causes the survival rate to dramatically decrease to 23%. furthermore, inaccurate screening for skin cancer can lead to numerous unnecessary, biopsies and excisions of benign skin lesions.

Visual inspection of suspicious skin lesions is usually the first of a series of 'tests' to diagnose skin cancer. The diagnostic accuracy of visual inspection alone is important to decide whether additional tests, such as a biopsy, are needed [4]. Recently, NN ML implementations have shown great promise in matching and even beating dermatologist's visual diagnosis accuracy when using dermatoscopic images for training [6]

Also, with the availability of affordable mobile phone skin magnifier attachments, people can now capture highly magnified dermatologist-grade photos of moles or other skin lesions. This works in favor of both the patient and the doctor, as it increases accessibility to convenient skin health assessment, while reducing unnecessary in-clinic visits and imaging times. Studies are already demonstrating that patients find the devices easy to use and encourage a greater level of engagement in their skin health [5]

Mobile phone skin magnifier attachment: https://dermlite.com/products/dermlite-hud

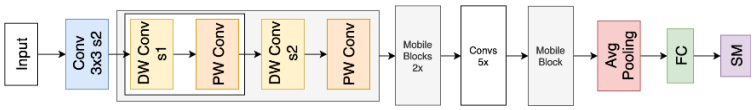
SUMMARY

After conducting more than 30 experiments, I was able to reach a dev set Multi-Class Accuracy (MCA) that beats the Human error (Bayes error proxy) and the average ML model performance achieved in the 2018 ISIC skin cancer detection challenge:

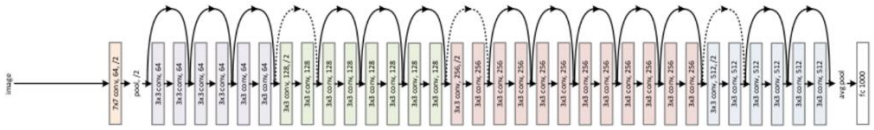
Mean Human Reader (Dermatologists & general practitioners)	**Mean - Human Reader Expert** (more than 10 years of experience)	**Mean - Top 3 ML Models** (2018 ISIC challenge)	**Mean – ML Models** (2018 ISIC challenge)	**My Best Performing Model (MobileNetv2) - so far**
0.60	0.63	0.86	0.66	0.724

MODELS AND APPROACH

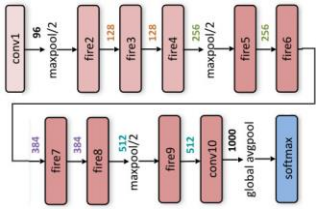
Due to the small size of my data set, my strategy was to leverage transfer learning using CNN models pretrained on the ImageNet data set. Specifically, I experimented with the following CNN architectures:



MobileNetv2



Resnet50



SqueezeNet

My transfer learning strategy was as follow:

- Transfer and freeze all layers and weights except the last fully connected (FC) layer and the Softmax layer
- Replace the removed FC layer with different FC architectures
- Replace the original softmax layer (1000 ImageNet classes) with a new Softmax layer (7 HAM1000 classes)
- Only train the weights of the new FC layers
- Use ReLU activation in the fully connected (FC) layers

Throughout my experiments I tried different performance tuning configurations:

- Multiclass cross entropic loss function vs. multiclass weighted loss function
- Different FC layer architectures: 80, 160+80,, 240+160.
- Batch normalization
- Mini batch sizes: 16,32,64, 128
- # of epochs: 20, 30, 50, 100
- Learning rates: 2E-01 --- 2E-05
- Optimizer: Used ADAM only

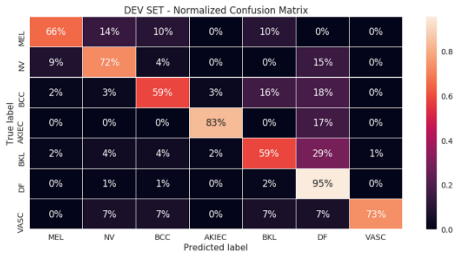
RESULTS

Below is a sample of the conducted experiments that illustrate the impact of different architecture configurations and hyperparameter selections on models' performances.

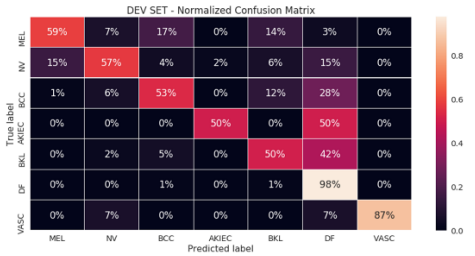
Model Configurations	**ResNet50 (size~100Mb)**		**MobelNetv2 (size~10Mb)**		**SqueeseNet (Size ~4Mb)**	
	Training MCA	Dev MCA	Training MCA	Dev MCA	Training MCA	Dev MCA
Baseline: cross entropy loss, no batch norm, 120FC+softmax, 20 epochs, bs=16, lr=2e-05	0.694	0.453	___	___	___	___

Weighted loss, no batch norm, 120FC+softmax, 20 epochs, bs=16, lr=2e-05	0.673	0.492	___	___	___	___
Weighted loss, batch norm, 120FC+softmax, 20 epochs, bs=16, lr=2e-05	0.657	0.491	___	___	___	___
Weighted loss, batch norm, 120FC+Softmax, 100 epochs, bs=16, lr=2e-05	0.977	0.648	0.632	0.476	0.142	0.142
Weighted loss, batch norm, 240FC+80FC+Softmax, 100 epochs, bs=16, lr=2e-05	0.835	0.532	0.93	0.724	0.142	0.142

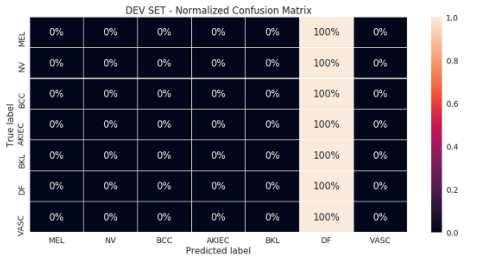
The confusion matrices clearly show the impact of the unbalanced data set on the models' prediction performances. Also, the SqueezeNet confusion matrix clearly shows that the SqueezeNet model is overwhelmed by the unbalanced data set:



Mobilnetv2 Confusion Matrix



Resnet50 Confusion Matrix

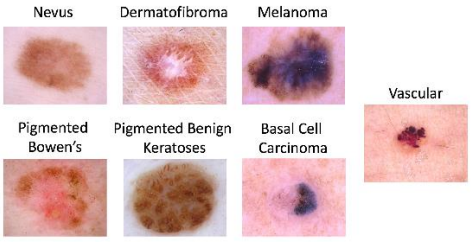


SqueezeNet Confusion Matrix

DATA

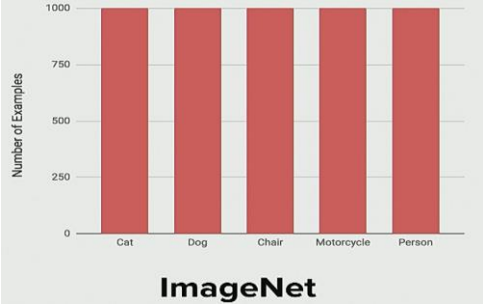
The HAM1000 dataset is a large collection of multi-source dermatoscopic images of common skin lesions. The data set consists of 10,015 JPEG images which were made public through the International Skin Imaging Collaboration (ISIC) archive.

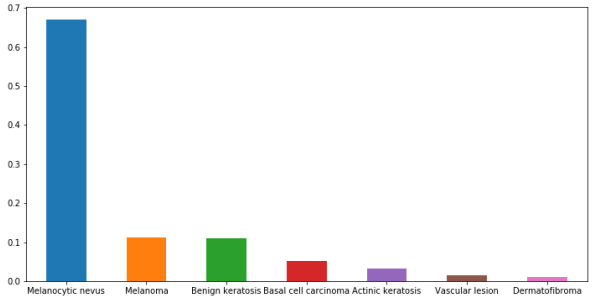
The images labels are stored in a CSV file and classified into 7 different disease categories: Actinic keratosis(akiec); Basal cell carcinoma(bcc); Benign keratosis(bkl); Dermatofibroma(df); Melanoma(mel); Melanocytic nevus(nv); Vascular lesion(vasc).



Sample data set images

The HAM1000 dataset is heavily unbalanced with about 70% of the images belonging to the Melanocytic Nevus (NV) class. Below is a comparison between a sample IMAGENET data set distribution and the HAM1000 data set distribution:





About 70% of the HAM1000 data set images belong to the Melanocytic Nevus class

To prepare the data set for model training and testing, I processed the images and labels as follow:

- Normalized and resized images to 224 x 224 x 3 dimensions
- Shuffled images then stored them in a Numpy array on disk for faster data loading
- Split data set: 90% training set; 10% dev set
- Converted labels to stacked transposes of one-hot vectors.

Analysis And Findings

Performance metric selection

Given that the HAM1000 is heavily unbalanced, I chose to use the balanced Multi-Class Accuracy (MCA), i.e. balanced recall, as a model evaluation metric:

$$\text{balanced accuracy} = \frac{\text{sensitivity} + \text{specificity}}{2} \quad \text{Specificity} = \frac{TN}{TN + FP} \quad \text{TPR /Recall / Sensitivity} = \frac{TP}{TP + FN}$$

The balanced MCA avoids inflated performance estimates due to unbalanced data sets. Also, the balanced MCA is the standard metric used to evaluate human based performance (i.e. Bayes error proxy) [6] and by skin cancer detection ML Models [1]

Loss function selection

When using a normal cross-entropy loss function, the models predicted the dominating class (*Melanocytic Nevus class*) more often suggesting that models were overfitting to that class. To alleviate this problem, I used a weighted loss function that multiplies the cross-entropy loss function by the frequency of classes. As a result of this new loss function, the classes occurring more often are penalized with a higher loss whereas those that occur less often are rewarded with a lower loss. The impact of using the new loss function on models' performances was significant: *the dev set MCA of my best performing Mobilenetv2 model jumped from 0.59 to 0.72!*

$$J = (\sum_{j=1}^{b} w_i Y_{ij}).(\frac{1}{b}\sum_{j=1}^{b}\sum_{i=0}^{n-1} Y_{ij} \log \hat{Y}_{ij})$$

Class weighted loss function

- **wi** is the weight i
- **Yij** is the true label of class i in example j
- **b** is the batch size
- **Ŷij** is the softmax probability of class i predicted for example j

Key Findings

- The baseline Resnet50 model achieved an MCA performance of about 70%. That's is to be expected as it is easy for the model to achieve close to 70% accuracy by simply predicting all images to be of the dominant class (70% of data set images belong to the *Melanocytic Nevus* class*).*
- Squeezenet model performance does not improve regardless of whatever architecture and hyperparameter values I tried. It seems the model is too small/simple to fit my data set. It is also clear that this model architecture is overwhelmed by unbalanced data.
- Using a weighted loss function caused the most dev set performance improvement across all models, except the SqueezeNet model.
- MobileNetv2 is a highly performant model that beat Resnet50 with 1/10th of the model size!
- Unbalanced data has a significant impact on the models performances, preventing models from generalizing well.

FUTURE WORK

- While I did manage to eliminate the avoidable bias (MobileNetv2 training performance = 93% MCA; expert doctors performance=63%MCA), there is still a considerable variance between my best model's training MCA (93%) and dev MCA(72%) even after using the weighted loss function and trying various model configurations. I believe DEV performance could be further improved using different regularization techniques (e.g. L1/L2 regularization).
- Also, given the limitation of my AWS ML machine, I could not experiment with different data augmentation techniques as that significantly increased the aws machine storage and memory requirements. As I was approaching my Stanford aws credit limit, I chose not to further explore that route. In the future, I would like to conduct data augmentation (e.g. mirroring, cropping, etc.) experiments as I believe data augmentation has great potential to further improve the models' ability to generalize.
- Given that the MobileNetv2 CNN architecture performs much better on my skin cancer detection task than the rest of the other architectures and with a much smaller storage footprint (~10Mb), It is definitely the best suited for deployment on mobile and small factor devices. As such, going forward I will focus on the MobileNetv2 architecture while also evaluating other mobile optimized architectures: i.e. the EfficientNet CNN architecture.

REFERENCES

- *[1] Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC); https://arxiv.org/abs/1902.03368*
- *[2] The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions; https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6091241/*
- *[3] Pruning Convolutional Neural Networks for Resource Efficient Inference; https://arxiv.org/abs/1611.06440*
- *[4] Visual inspection for diagnosing cutaneous melanoma in adults; https://www.cochrane.org/CD013194/SKIN_how-accurate-visual-inspection-skin-lesions-naked-eye-diagnosis-melanoma-adults)*
- *[5] Consumer acceptance of patient-performed mobile teledermoscopy for the early detection of melanoma; https://onlinelibrary.wiley.com/doi/full/10.1111/bjd.14630#bjd14630-fig-0001*
- *[6] Comparison of the accuracy of human readers versus machine-learning algorithms for pigmented skin lesion classification; https://www.thelancet.com/journals/lanonc/article/PIIS1470-2045(19)30333-X/fulltext*
- *[7] Tensornets:* High level network definitions with pre-trained weights in TensorFlow; *https://github.com/taehoonlee/tensornets*