Skin Cancer Classification with Machine Learning & Deep Learning

Demarcus Wirsing, Drishti Arora, Nigist Woldeeyesus, Sai Kumar Kanuru ,Yunpeng Li

University of Maryland Baltimore County 1000 Hilltop Circle, Baltimore, MD 21250

Data 606: Capstone project

December 20, 2021

Abstract

Skin cancer is one of the most dangerous forms of cancer. It is caused by unrepaired deoxyribonucleic acid (DNA) in skin cells, which generate genetic defects or mutations on the skin (Ashraf, Rehman & Maqsood, 2020). Skin cancer tends to gradually spread over other body parts, so it is more curable in initial stages, which is why it is best detected at early stages. The increasing rate of skin cancer cases, high mortality rate, and expensive medical treatment require that its symptoms be diagnosed early. Fortunately, the recent rise of machine learning and deep learning technologies have been applied more and more to medical science and cancer detection. In this paper we apply various machine learning and deep learning techniques to skin lesion classification and analyze the performance of various models. We further invent a deep learning hybrid model, which is built on a prebuilt CNN model and can take both image and clinic tabular data to do skin lesion classification. Our hybrid model outperforms all other models.

I. INTRODUCTION

Skin cancer is one of the most common cancers in the United States today. Because the skin is the body's largest organ, it's unsurprising that skin cancer is the most prevalent type of cancer in humans (Ashraf, Rehman & Maqsood, 2020). It is generally classified into two major categories: melanoma and nonmelanoma skin cancer. Melanoma is a hazardous, rare, and deadly type of skin cancer. According to statistics from the American Cancer Society, melanoma skin cancer cases are only 1% of total cases, but they result in a higher death rate (Layode & Rahman, 2019). Based on the article (About melanoma skin cancer. (*n.d.*)., 2021) Melanoma is a type of cancer that develops in cells called melanocytes. It begins when normal melanocytes grow out of control, resulting in a malignant tumor. It has the ability to influence any part of the human body. It commonly arises on sun-exposed areas, such as the hands, face, neck, lips, and so on. Melanoma skin cancer can only be cured if caught early; else, it will spread to other regions of the body and cause a painful death. The biopsy procedure is commonly used by doctors to identify skin cancer. Multiple noninvasive procedures are

offered to investigate skin cancer symptoms and determine whether they reflect melanoma or nonmelanoma (Slamdot, 2021). Melanoma skin cancer accounts for only 1% of all instances, according to the American Cancer Society, but it is associated with a higher fatality rate (About melanoma skin cancer. (*n.d.*)., 2021).

Artificial intelligence (AI) has been the topic of tremendous media frenzy around the world in recent years. Innumerable publications from various fields outside of technology have been published about the use of machine learning, deep learning, and AI (Chollet, 2017). During the last few decades, deep learning has completely transformed the machine learning landscape. It is the most advanced subfield of machine learning that deals with artificial neural network techniques. The function and structure of the human brain inspired these algorithms. On a variety of complex computer vision and image classification tasks, deep learning algorithms have reached human-level performance.

Deep learning techniques are now widely used in the medical imaging sector for a variety of applications, such as illness detection (Rashid, Tanveer & Aqeel Khan, 2019). In recent years, a number of deep learning algorithms have been employed to diagnose skin cancer using computers. Skin cancer detection strategies based on machine learning and deep learning are comprehensively discussed and analyzed in this work. This research focuses on presenting skin cancer classification techniques using several machine learning and deep learning algorithms, and it further reveals that employing a hybrid deep learning model, which can be trained by both clinic tabular data and skin image data, outperforms other approaches. Diagram.1 depicts the overall process and techniques we used to predict skin cancer classification for this project.

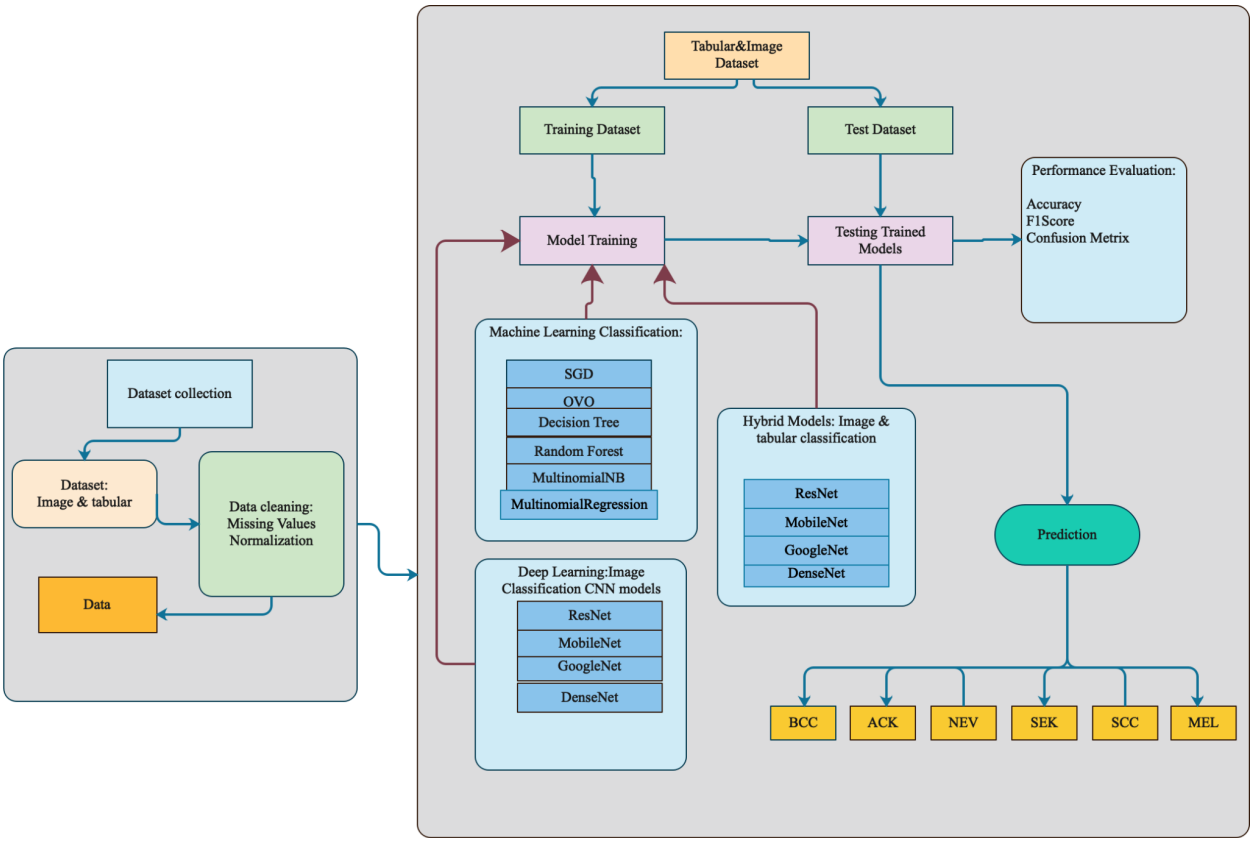

Diagram 1. *Skin Cancer classification prediction major processes*

II. LITERATURE REVIEW

Skin cancer is one of the most frequent cancers among human beings. Diagnosing an unknown skin lesion is the first step to determine appropriate treatment. If there is a suspicion of skin cancer, a visual examination of the lesion is performed first, followed by a clinical study. Deep learning-based image classification, in particular, has lately demonstrated high accuracy in medical picture categorization.

Sheha, Mariam A., et al. offer an automated approach for diagnosing melanoma using a series of dermoscopic pictures (Sheha, 2012). To identify Melanocytic Nevi and Malignant Melanoma, features retrieved are based on gray level Co-occurrence matrix (GLCM) and Multilayer perceptron classifier (MLP). Automatic MLP and Traditional MLP were proposed as two separate training and testing strategies for the MLP classifier.

Layode Oyebisi., et al., proposed an integrated decision support system for the automatic skin cancer recognition of pigmented skin lesions (Layode, 2019). They used a deep learning model based on U-Net architecture that has been adapted for the segmentation of skin lesions which involves linking each pixel of an image to a class label. Then different features extracted from the pre-trained CNNs were fused together in all possible combinations using the Partial Least Square Canonical Correlation Analysis (CCA).

Salian, Abhishek C., et al. proposed a custom CNN model in their study (Salian, 2020). They used publicly available and augmented labeled images of PH2 and HAM10000 dataset for evaluation. Using data augmentation, they created more samples by rotating the images by a little angle, flipping the image to the side or flipping an image upside down. They mainly used MobileNet and VGG-16 which are pre-trained on Imagenet dataset and custom model to achieve results for comparisons. All major studies mainly focus on classification of image data, but they lack considering other clinic data in tabular format that may play an important role for skin cancer diagnosis.

III. DATASET

We have retrieved the dataset for this analysis from https://data.mendeley.com/. The dataset is publicly available and includes both image and metadata about some patient detail. The PAD-UFES-20 dataset was collected along with the Dermatological and Surgical Assistance Program (in Portuguese: Programa de Assistência Dermatológica e Cirúrgica - PAD) at the Federal University of Espírito Santo (UFES-Brazil). The dataset consists of 2,298 samples of six different types of skin lesions. Each sample consists of a clinical image and up to 22 clinical features including the patient's age, skin lesion location, Fitzpatrick skin type, and skin lesion diameter. The skin lesions are: Basal Cell Carcinoma (BCC), Squamous Cell Carcinoma (SCC), Actinic Keratosis (ACK), Seborrheic Keratosis (SEK), Melanoma (MEL), and Nevus (NEV).

The metadata associated with each skin lesion is composed of up to 26 features. All features are available in a CSV document in which each line represents a skin lesion

and each column specifies a feature. The "diagnostic" column shows the skin lesion types, which is the main target column where we have applied machine learning and deep learning methods to predict values. In total, there are 1,373 patients, 1,641 skin lesions, and 2,298 images present in the dataset. Each image/sample has a reference to the patient and the skin lesion in the metadata.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
patient_id	lesion_id	smoke	drink	backgroun	backgroun	age	pesticide	gender	skin_cance	cancer_his	has_piped	has_sewa	fitspatrick	region	diameter_	diameter_	diagnostic
PAT_1516	1765					8								ARM			NEV
PAT_46	881	FALSE	FALSE	POMERAN	POMERAN	55	FALSE	FEMALE	TRUE	TRUE	TRUE	TRUE	3	NECK	6	5	BCC
PAT_1545	1867					77								FACE			ACK
PAT_1989	4061					75								HAND			ACK
PAT_684	1302	FALSE	TRUE	POMERAN	POMERAN	79	FALSE	MALE	TRUE	FALSE	FALSE	FALSE	1	FOREARM	5	5	BCC
PAT_1549	1882					53								CHEST			SEK
PAT_778	1471	FALSE	TRUE	GERMANY	ITALY	52	FALSE	FEMALE	FALSE	TRUE	TRUE	TRUE	3	FACE	15	10	BCC
PAT_117	179	FALSE	FALSE	POMERAN	POMERAN	74	TRUE	FEMALE	FALSE	FALSE	FALSE	FALSE	1	FACE	15	10	BCC
PAT_1995	4080					68								FOREARM			ACK
PAT_705	4015	FALSE	TRUE	GERMANY	GERMANY	58	TRUE	FEMALE	TRUE	TRUE	TRUE	TRUE	1	FOREARM	9	7	ACK
PAT_2140	4726					45								NECK			ACK
PAT_967	1827	FALSE	FALSE	POMERAN	POMERAN	34	TRUE	FEMALE	TRUE	FALSE	FALSE	FALSE	2	NOSE	5	4	BCC
PAT_2088	4524					9								CHEST			NEV
PAT_636	1204	FALSE	FALSE	BRAZIL	BRAZIL	78	FALSE	FEMALE	TRUE	TRUE	TRUE	TRUE	2	CHEST	20	18	BCC

***FIG.1**: Sample tabular data*

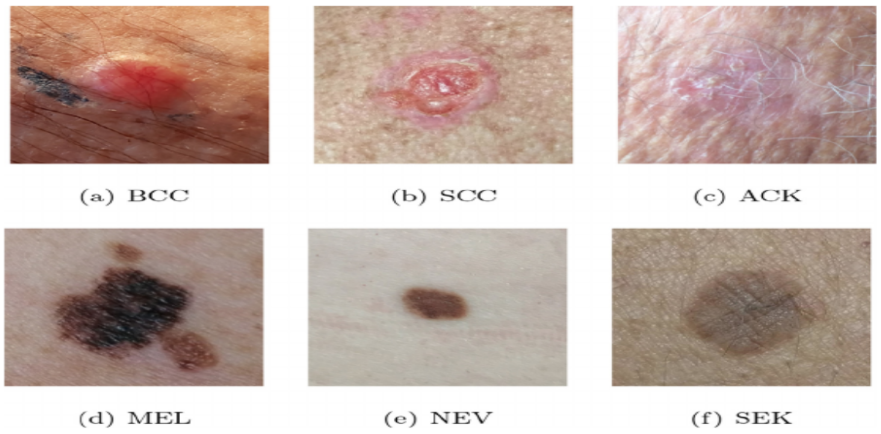

***FIG.2**: Image data sample of the six lesion types*

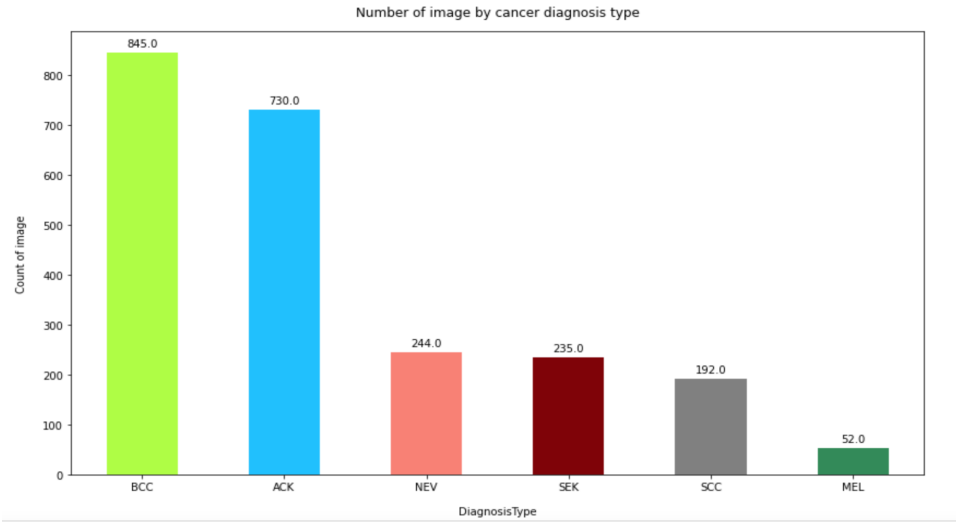

FIG.3: *The number of samples for each type of skin lesion present in the PAD dataset.*

IV. METHODOLOGY

To achieve the goal of this project we have applied both machine learning and deep learning methodologies consisting of several phases or steps summarized below. All our source code can be found at https://github.com/yunpengliDataScience/Skin_Cancer_ML_DL

1. Preprocess and analyze the metadata and construct various machine learning models to predict and evaluate the results of skin lesion classifications.
2. Use image processing techniques and experiment with various prebuilt deep learning models on skin image data to evaluate the results of skin lesion classifications.
3. Use prebuilt Convolutional Neural Networks (CNN) to inference missing values in columns of clinic tabular data.
4. Construct a hybrid model, which utilizes prebuilt CNN models and can take both image and tabular data through a deep learning approach to do skin lesion classifications.
5. Evaluate the accuracy and performance of various models and find the best model for skin lesion classifications.
6. Construct a data web application using streamlit to visualize our Google Colab notebooks, for a more uniform/seamless experience when presenting our project and findings.

Machine learning Methodology:

The data was pre-processed in order to prepare it for modeling. Because the results of machine learning classification algorithms are dependent on the quality of raw data, pre-processing is critical for data improvement. As a result, attributes with missing values and 'UNK' were imputed and replaced by mean for continuous data and mode for nominal data during this process. The continuous data was normalized as well.

All of the categorical variables in the dataset are converted to numerical values using the One-hot encoding technique in the first stage of building predictive models. We used this to train several machine learning algorithms. We have trained the SGDClassifier, OVO, Random Forest, and Decision tree algorithms. All the accuracy scores are less than 75% initially.

The goal of the second stage was to improve on the initial results. We used a variety of preprocessing and feature engineering techniques to achieve this. First, we used the dython library to determine the correlation coefficient of all variables, and we discovered the strongly correlated variables against the target variable as shown in the Figure.4. In the heat map features with coefficients over 0.5 were "biopsed", "Fitzpatrick", and "age", which had values of 0.83, 0.57, and 0.55 respectively.

Second, for the dataset that was fed into the machine learning models, we utilized a Stepwise regression test to discover the best predictor variables. Out of a total of 20 variables, six had a p-value of 0.05 and were found to be strong predictors of skin cancer diagnostic variables.

Our third approach involved Chi-squared test which is a non-parametric statistical test that shows the relationship between categorical variables and the relationship of categorical variables to the target variable. The test is about hypothesis testing if the independent variable contributes in predicting the target variable (Dr. Goswami,2020). We got a chi-square statistic score for each variable, and the variables with highest scores were selected.

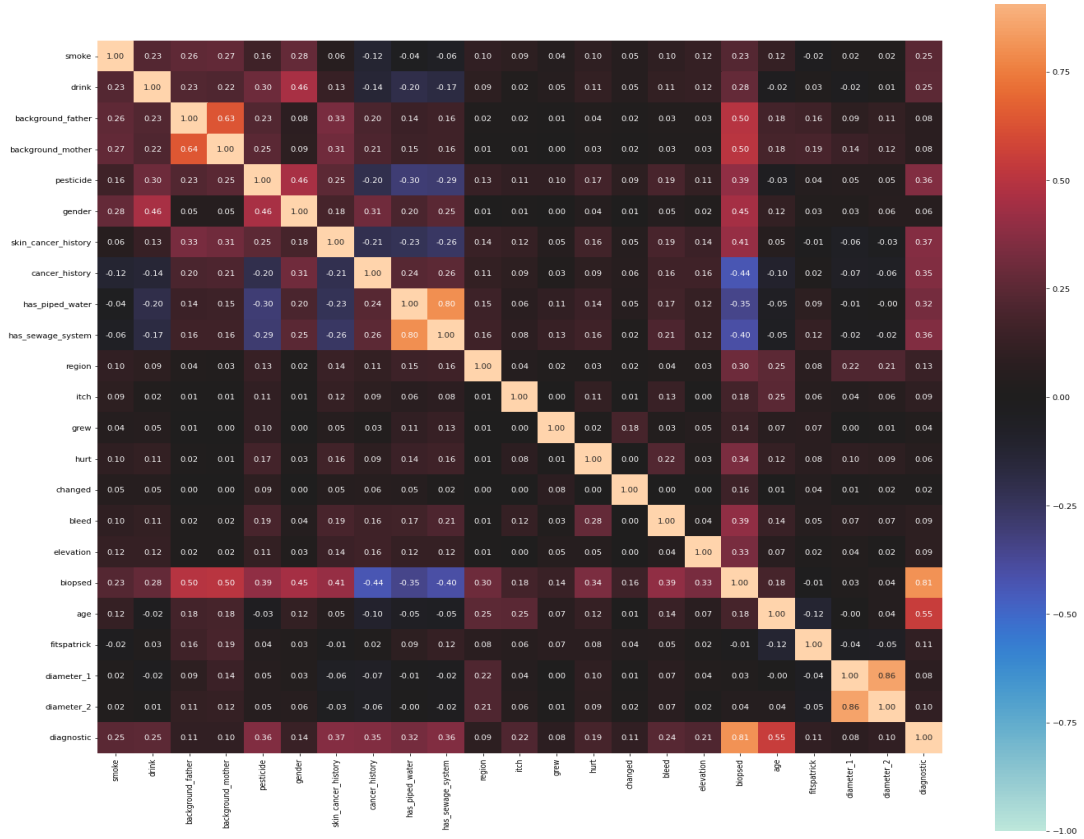

FIG.4 *correlation matrix associated with the variables used in this analysis.*

Considering the outcome of Correlation coefficient, Chi square test and Stepwise regression, the best predictors to train machine learning classification models are: 'region','itch','grew','changed','bleed','elevation','biopsed','age','smoke','drink','back ground_father','pesticide','skin_cancer_history','cancer_history','has_swage_system' and 'fitspatrick'.

Deep Learning Methodology

Prebuilt CNN Model Experimentation and Selection

During data exploration analysis stage, images of six types of skin lesion are programmatically copied into six image sub-folders to comply with the directory structure that Pytorch's torchvision.datasets.ImageFolder favors, so that prebuilt CNN models can easily access them. To deal with imbalanced data, we have applied sampling

weights and augmentation techniques, such as resizing, random flipping, random rotation, cropping, and color jittering transformations on the image data when data is being loaded for training.

We have utilized Pytorch library API to experiment with several prebuilt CNN models, such as ResNet18, ResNet50, GoogleNet, DenseNet121, DenseNet161, DenseNet201, and MobileNet V2, to train skin lesion image data.

ResNet is built on deep residual networks and pre-trained on ImageNet, and it can be optimized to gain accuracy from considerably increased depth (He et al., 2015). GoogleNet is based on a deep convolutional neural network architecture codenamed "Inception", and it is designed with computational efficiency, so that inference can be run on individual devices including even those with limited computational resources, especially with low-memory footprint (Szegedy et al., 2014). The network is 22 layers deep when counting only layers with parameters (or 27 layers if we also count pooling). DenseNet Connects each layer to every other layer in a feed-forward fashion, so it is substantially deeper, more accurate, and efficient to train if they contain shorter connections between layers close to the input and those close to the output (Huang et al., 2016). DenseNet alleviates the vanishing-gradient problem, strengthens feature propagation, encourages feature reuse, and substantially reduces the number of parameters (Huang et al., 2016). DenseNets naturally scale to hundreds of layers while posing no optimization challenges and show a steady increase in accuracy as the number of parameters increases, with no symptoms of performance degradation or overfitting (Huang et al., 2016). MobileNet are efficient networks optimized for speed and memory, and they are based on an inverted residual structure where the shortcut connections are between the thin bottleneck layers which are opposite to traditional residual models (Sandler et al., 2018). MobileNet Uses lightweight depth wise convolutions to filter features in the intermediate expansion layer (Sandler et al., 2018).

We have used the stratified strategy, which propositionally takes data from each category, to split the dataset in 80% for training and 20% for testing. After 50 epochs of

training the prebuilt CNN models, the validation accuracy scores of four prebuilt CNN models, ResNet18, GoogleNet, DenseNet121, and MobileNet V2, have reached the top four spots. Therefore, these four models become the best four candidates for our further model construction and analysis. The validation accuracy scores are shown in the following table.

Prebuilt Model	Validation Accuracy Score
ResNet18	0.667
ResNet50	0.559
MobileNet V2	0.680
GoogleNet	0.637
DenseNet121	0.704
DenseNet161	0.598
DenseNet201	0.636

Table 1. *Accuracy scores of prebuilt CNN models*

We have also trained the best four prebuilt CNN models, ResNet18, GoogleNet, DenseNet121, and MobileNet V2, to experiment with data inferencing on other columns besides the "diagnostic" target column. The main purpose of column inference is to use deep learning models trained by image data to predict missing values in columns. The main strategy is to treat a particular tabular column as a class label, and then train images in deep learning models to predict the value of that column. The performance results are discussed in the Analysis section.

Customized Hybrid Models

We think both patients' clinic tabular data and skin image data may all contribute to the diagnosis of a skin cancer, so we decide to integrate both types of data into our model training. We have defined a dataset class named SkinImageTabularDataset that extends Pytorch's torch.utils.data.Dataset class to extract tabular data and retrieve image data referenced in an encoded metadata file in the format of comma-separated value (CSV). The metadata CSV file contains the encoded tabular columns, an img_id column that references the image names in a skin lesion image directory, where all skin lesion

images reside, and the encoded diagnostic column, the target column representing the diagnostic skin cancer types in a numerical format (ACK: 0, BCC: 1, MEL: 2, NEV: 3, SCC 4, SEK: 5). We have also experimented with both One-hot encoding and Ordinal encoding techniques on categorical columns of the tabular data.

In order to train a customized deep learning model with the combined tabular and image data, we have constructed a hybrid model named ImageTabularHybridModel, which extends Pytorch's nn.Module class. The hybrid model can be configured and initialized by a user parameter to train image only data, tabular only data, or combined image and tabular data. Another user parameter specifies what the underlying prebuilt CNN model that the hybrid model should utilize to train image data. Currently, it supports our four best candidates: ResNet18, GoogleNet, DenseNet121, and MobileNet V2. When the hybrid model is initialized to train combined image and tabular data, it pushes image data through the prebuilt CNN model specified by the user parameter, and the last layer of the original prebuilt CNN model is hijacked and extended to connect 3 fully connected layers with Relu activation. The tabular data is pushed through another 3 fully connected layers with Relu activation. Then the output data of the image layer is concatenated with the output data of the tabular layer, and such concatenated data is further pushed through additional 3 fully connected layers with Relu activation. The last layer outputs 6 nodes that represent the 6 types of skin lesions. The below figure shows the architecture overview of the hybrid model. The loss function for training the model is Pytorch's nn.CrossEntropyLoss.

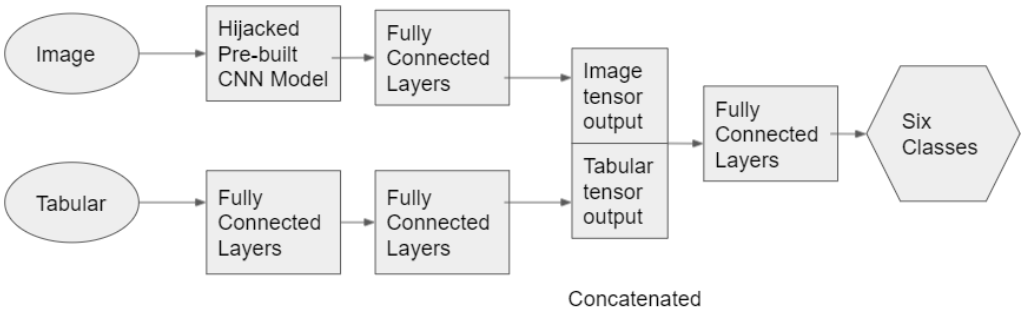

Diagram 2. *Architecture of hybrid model*

For training and testing, we have used the stratified strategy, which propositionally takes data from each skin lesion category, to split the dataset in 80% for training and 20% for testing. To deal with imbalanced data, we have applied upsampling and augmentation techniques, such as resizing, random flipping, random rotation, cropping, and color jittering transformations on the image data when data is being loaded for training.

In order to determine what encoding technique on categorical data is better for our models, we have experimented with the One-hot encoding and Ordinal encoding techniques on the tabular columns to compare the performance of the hybrid models built on the four prebuilt CNN models. The results are discussed in the Analysis section.

We have trained our models with data including all categorical tabular columns and data excluding several tabular columns, such as “backgroud_mother”, “gender”, “cancer_history”, “has_piped_water”, “diameter_1”, and “diameter_2”, which we think have relatively low correlations to the target column during EDA and machine learning phase, to see if there are significant differences. The results are discussed in the Analysis section.

We suspect that the “biopsed” column may have significant correlation or bias to our skin lesion classification, so we have retrained the hybrid models that exclude the “biopsed” column to see if there are any performance differences from those of hybrid models including the “biopsed” column. The results are discussed in the Analysis section.

V. ANALYSIS

A. **Preprocessing**:

Since all machine learning algorithms cannot simply function on literal data, the input and output variables for machine learning models need to be converted into numeric. There are several methods for converting categorical variables to numeric

variables, but we chose ordinal encoding in machine learning models since it does not add any new columns to the dataset, unlike One-hot encoding, and assigns a number to each unique value in the feature (Reddy, 2019). To fit and transform the target variable, we utilized OrdinalEncoder and the output was a transformed array. However, we applied and compared both One-hot encoding and ordinal encoding in the deep learning phase.

When numerical input variables are scaled to a standard range before being used in machine learning models, the models perform better and are less biased. It translates each input variable individually to the 0-1 range, which provides higher precision (Brownlee, 2020). We used StandardScaler to standardize the data, resulting in a distribution with a mean of 0 and a standard deviation of 1.

B. Machine Learning Models

Stochastic Gradient Descent (SGD) Classifier is an efficient approach used in sparse machine learning problems which supports multi-class classification and implements a first-order SGD learning routine which iterates over training examples and updates the model parameters (Tsuruoka,Tsujii & Ananiadou, *n.d*.).

It also combines multiple binary classifiers in a ‘one versus one’ (OVO) scheme. For each of the k and k-1 classes, one dataset is made for each class versus every other class. The number of models generated will be k(k-1)/2. At testing time, each classification is given one vote for the winning class and highest votes determine which class test data belongs to (Fuchs, 2019). Splitting the data into train and test sets (0.8 of the total instances is the train data and 0.2 the test data), the accuracy score obtained with both SGD Classifier and OVO Classifier is 65%.

Random Forest fits a large number of individual decision tree classifiers that operate as an ensemble in which each decision tree gives a class prediction and the class with the most votes becomes our model’s prediction (Yiu, 2019) . The accuracy obtained for Random Forest Classifier with criterion as entropy (measurement of uncertainty) and

max_depth (maximum depth of tree) as 50 is 81%, and it performs better than other classification models. Confusion matrix for this best performing model is shown in the figure below (FIG. 5).

Also, the feature importance has been calculated using the notion of Gini impurity. The result is presented in figure (FIG. 6). The most dominant features are 'biopsied' and 'age', and the least important are 'has_piped_water' and 'gender'.

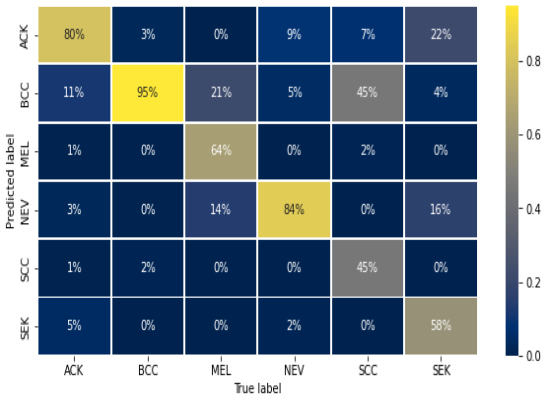

FIG 5. *Confusion matrix for Random Forest Classifier*

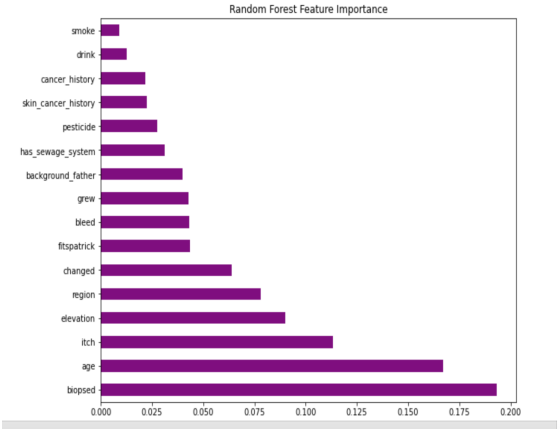

FIG 6*. Feature importance using Random Forest*

Decision Tree Classifier represents all possible solutions to a decision based on certain conditions. At each node in the model, conditions are formed on the features to separate all classes in the dataset to the fullest purity, and the algorithm assigns one class to the data points in each leaf node (Chakure, 2019). The result for Decision Tree Classifier with entropy and max_depth of 50 is 74%. The most dominant features obtained after calculating feature importance using a decision tree are 'biopsed' and 'age', and the least important are 'drink' and 'smoke'.

Multinomial Naive Bayes Classifier is robust and is based on Bayes' theorem, which shows that the features in the dataset are mutually independent, and the probability of features given class is a multinomial distribution. To deal with the zero-probability problem, we used alpha which is the Laplace smoothing parameter and handles zero probability. We chose alpha=1, and the accuracy obtained is 68%.

Multinomial Logistic Regression is the generalization of logistic regression algorithms to implement for multi-classification tasks as it consists of two functional layers: a) Linear Prediction Function (logit layer) b) SoftMax Function (SoftMax layer). It combines the two layers, takes a vector of features as the input and on the basis of features computes the probabilities for suitable outcome (Sharma, 2020).

After splitting the data in train and test sets, the accuracy score obtained for Multinomial logistic regression with 'Multinomial' class and 'newton-cg' solver is 72%.

The result of Machine Learning models is summarized in Table 3

Machine Learning Model	Accuracy score
SGD Classifier	63%
OVO Classifier	65%
Random Forest Classifier	81%
Decision Tree Classifier	74%
Multinomial NB Classifier	68%
Multinomial Logistic Regression	72%

Table 2. *Result of Machine Learning models*

C. Prebuilt CNN Models for Column Inferencing

The column inferencing results are illustrated in the below figures. After 50 epochs of training on the ReNet18 model, the 5 columns with inferencing scores reaching accuracy score 0.7 and above are "biopsed", "smoke", "bleed", "elevation", and "drink". After 50 epochs of training on the MobileNet V2 model, the 7 columns with inferencing scores reaching accuracy score 0.7 and above are "biopsed", "changed", "smoke", "bleed", "elevation", "drink", and "hurt". After 50 epochs of training on the GoogleNet model, the 8 columns with inferencing scores reaching accuracy score 0.7 and above are "biopsed", "changed", "smoke", "bleed", "elevation", "drink", "hurt", and "itch". After 50 epochs of training on the DenseNet121 model, the 8 columns with inferencing scores

reaching accuracy score 0.7 and above are “biopsed”, “smoke”, “elevation”, “drink”, “changed”, “bleed”, “itch”, and “hurt”. Both GoogleNet and DenseNet121 have relatively better overall performance on inferencing column values of our tabular data. All results show the column “biopsed” has the highest inferencing scores, so it indicates column “biopsed” has strong correlation with skin lesion types. A possible explanation is that a doctor may intend to order a skin biopsy when the patient is suspected to have skin cancer after viewing the patient’s skin image.

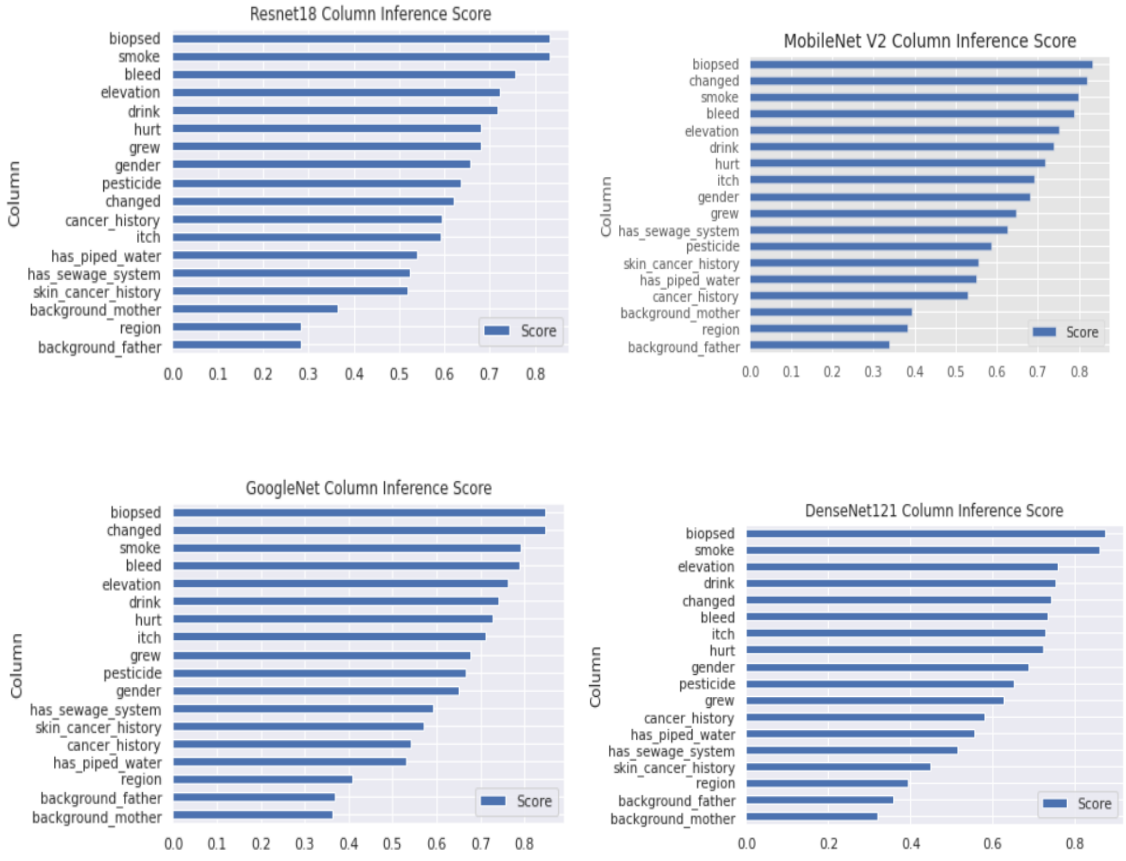

FIG 7. *Accuracy scores of column inferencing*

D. Hybrid Models

One-Hot Encoding vs. Ordinal Encoding

The following table shows the best accuracy scores of hybrid models after 100 epochs of training on combined image data with One-hot Encoded tabular data or Ordinal

Encoded tabular data. One-hot Encoding has significantly better accuracy scores than the accuracy scores of Ordinal Encoding yielded by all the 4 hybrid models.

Underlying Models	Best Accuracy Score of One-Hot Encoding	Best Accuracy Score of Ordinal Encoding
ResNet18	0.833	0.798
GoogleNet	0.828	0.796
MobileNet V2	0.837	0.815
DenseNet121	0.830	0.791

Table 3. *Accuracy scores of one-hot Encoding vs Ordinal Encoding*

Full Tabular Columns vs. Reduced Tabular Columns

The following table shows the best accuracy scores of hybrid models after 100 epochs of training on combined image data and tabular data with reduced columns versus scores of training data with full columns. Reducing tabular columns seems not to have significant performance improvement. It may make sense because deep learning can figure out what features are important and ignore what are less important after a sufficiently big number of epochs of training. So, the original clinic columns may all have some contributions to the classification.

Underlying Models	Best Accuracy Score of Reduced Column	Best Accuracy Score of Full Column
ResNet18	0.835	0.830
GoogleNet	0.826	0.826
MobileNet V2	0.837	0.828
DenseNet121	0.826	0.843

Table 4. *Accuracy scores of models with reduced tabular columns vs full tabular columns*

With "Biopsed" Column vs. Without "Biopsed" Column

The following table shows the best accuracy scores of hybrid models after 100 epochs of training on combined image data and tabular data without "biopsed" columns versus scores of data with "biopsed" column. Although we suspect the "biopsed" column may introduce bias to skin lesion classifications, the results do not show the missing

"biopsed" column has significant influence on our hybrid validation accuracy scores. A possible explanation is that other clinic tabular columns could make up for the missing influence of "biopsed" columns and still contribute significantly to the classifications during deep learning.

Underlying Models	Best Accuracy Score of Model without Biopsed Column	Best Accuracy Score of Model with Biopsed Column
ResNet18	0.833	0.830
GoogleNet	0.828	0.826
MobileNet V2	0.837	0.828
DenseNet121	0.830	0.843

Table 5. *Accuracy scores of models without "biopsed" column vs with "biopsed" column*

Image Data Only vs. Tabular Data Only vs. Image Tabular Data Combined

Regardless of whether including or excluding certain clinic tabular columns, after training the 4 hybrid models, all results consistently show that models trained by image tabular combined data have higher validation accuracy scores than the scores of the models only trained by tabular data or only trained by image data. Such finding confirms our hypothesis that combining image data and clinic tabular data in deep learning for skin lesion classification can reach higher accuracy. The results are illustrated in the following figures. The green curve represents the validation accuracy score of models trained by image-tabular combined data in 100 epochs of training. The orange curve represents the validation accuracy score of models trained by tabular only data in 100 epochs of training. It is slightly lower than the green curve. The lowest blue curve represents the validation accuracy score of models trained by image only data in 100 epochs of training.

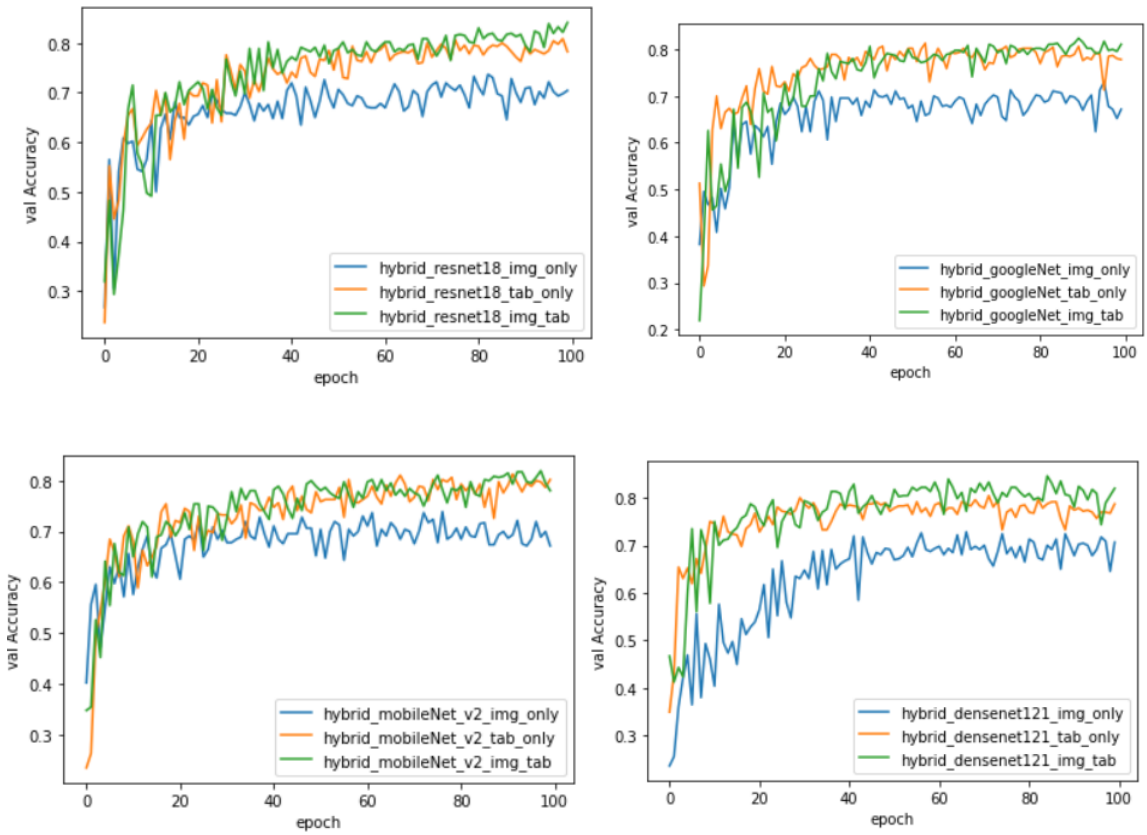

FIG 8. *Validation accuracy score curves of hybrid models in 100 epochs of training*

Best Hybrid Models

We have applied up-sampling to balance underrepresented skin lesion types in the training dataset and kept recording the best validation accuracy score of each hybrid model in each training epoch. The state and parameters of the best performance model are persisted in a file so that models can be easily loaded instead of being retrieved by retraining. The highest accuracy score is 0.846 of the hybrid models that utilize DenseNet121, so it is our best hybrid model for predicting skin lesion types from combined image and tabular input data. The following table shows the best validation accuracy scores for each hybrid model.

Underlying Models	Best Accuracy Score of Image Only	Best Accuracy Score of Tabular Only	Best Accuracy Score of Image Tabular Combined
ResNet18	0.737	0.809	0.841
GoogleNet	0.730	0.813	0.824
MobileNet V2	0.739	0.813	0.820
DenseNet121	0.728	0.804	0.846

Table 6. *Best accuracy scores of hybrid models*

The following figures illustrate the confusion matrix of each hybrid model.

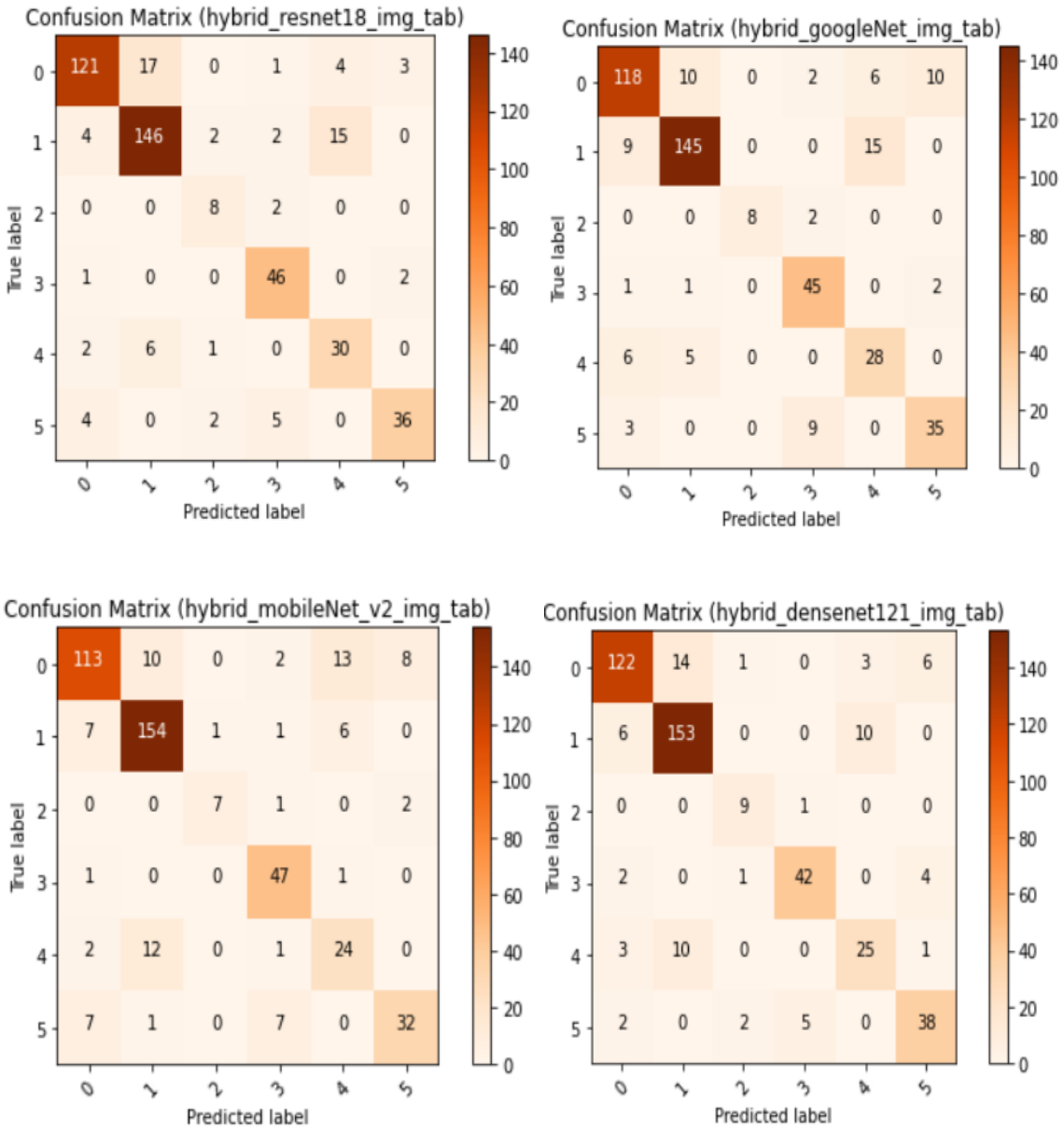

FIG9. *Confusion matrices of hybrid models*

VI. DATA APPLICATION

To display our group's progression along with our findings in our topic we have elected to create a data application using the streamlit python library to accomplish this goal. The python library streamlit is a relatively new library that has given the power to Data analyst and Data Scientist the ability to create, standup, as well as deploy fully working data applications within minutes. We have chosen to use this medium for our final presentation for the project because we wanted to go with a more elegant/streamlined approach than the conventional jupyter notebook and powerpoint slides.

With the use of this application, we are able to combine both slides and coded notebooks into one entity. Our group has decided that we will split up the application into three sections. The first section contains all three phases of our project's PowerPoint presentation structured in three expanded vertical fields. These three expanded fields will give the user of the application the ability to hide/show different phases of the project they would like to view. By default, all phases are expanded to view, but later can be hidden/collapsed if the user clicks on the plus icon (+) in the right of the field. Under the phases of the project, you will then see the reference and related work portion of the page where you will see all the academic articles/books we reviewed for our project.

VII. CONCLUSION

In this systematic review study, we have experimented and reviewed machine learning and deep learning models for skin cancer classification. Each algorithm or model comes with its own set of advantages and disadvantages. So far, our customized hybrid model built on DenseNet121 outperforms other models. In a practical clinic situation, a doctor not only examines skin images but also considers other formats of clinic data before a diagnosis is made. Our hybrid models can simulate such clinic situations because they are capable of taking both image data and clinic data of tabular format to make relatively good predictions for cancer diagnosis.

Besides skin cancer classifications, our concept of combining tabular data with image data in deep learning can also be extended to other fields where both tabular data and image data play important roles. Often, unstructured data may also contain important hidden information. For example, a conversation between a patient and a doctor or descriptions of certain symptoms may provide some clues to a diagnostic analysis; however, such information is not well organized in nature. Therefore, our future research may focus on building a more advanced model that not only can combine tabular data and image data but also be able to integrate with unstructured text data, where Natural Language Processing can be applied to extract other essential information, to further improve skin cancer diagnosis.

Bibliography

About melanoma skin cancer. (*n.d.*). Retrieved December 5, 2021, from https://prod.cancer.org/content/dam/CRC/PDF/Public/8823.00.pdf.

Amer Abdulkader, 10509213 Canada Inc., & Sarmad Tanveer. (2019). PyTorch for Deep Learning and Computer Vision. Packt Publishing.

Ashraf, R., Afzal, S., Rehman, A. U., Gul, S., Baber, J., Bakhtyar, M., Mehmood, I., Song, O.-Y., & Maqsood, M. (2020). Region-of-interest based transfer learning assisted framework for Skin cancer detection. *IEEE Access*, *8*, 147858–147871. https://doi.org/10.1109/access.2020.3014701

Brownlee, J. (2020, August 19). 4 types of classification tasks in machine learning. Machine Learning Mastery. Retrieved November 3, 2021, from https://machinelearningmastery.com/types-of-classification-in-machine-learning/.

Brownlee, J., How to Use StandardScaler and MinMaxScaler Transforms in Python, 2020, https://machinelearningmastery.com/standardscaler-and-minmaxscaler-transforms-in-python/

Chakure, A., Decision Tree Classification, 2019, https://medium.com/swlh/decision-tree-classification-de64fc4d5aac

Chollet, F. (2017). *Deep learning with python*. Manning Publications.

Elgamal, M. (2013). Automatic skin cancer images classification. *International Journal of Advanced Computer Science and Applications*, *4*(3). https://doi.org/10.14569/ijacsa.2013.040342

Fuchs, M., OvO and OvR Classifier, 2019, https://michael-fuchs-python.netlify.app/2019/11/13/ovo-and-ovr-classifier/#background-information-on-ovo-and-ovr

Goswami, S., Using the Chi-Squared test for feature selection with implementation, 2020, https://towardsdatascience.com/using-the-chi-squared-test-for-feature-selection-with-implementation-b15a4dad93f

He, K., Zhang, X., Ren, S., Sun, J., Deep Residual Learning for Image Recognition, (2015), https://arxiv.org/abs/1512.03385

Huang, G., Liu, Z., Maaten, L.V.D., Densely Connected Convolutional Networks, (2016), https://arxiv.org/abs/1608.06993

Islam, M. K., Ali, M. S., Ali, M. M., Haque, M. F., Das, A. A., Hossain, M. M., Duranta, D. S., & Rahman, M. A. (2021). Melanoma Skin Lesions Classification using Deep Convolutional Neural Network with Transfer Learning. 2021 1st International Conference on Artificial Intelligence and Data Analytics (CAIDA), Artificial Intelligence and Data Analytics (CAIDA), 2021 1st International Conference On, 48–53. https://doi-org.proxy-bc.researchport.umd.edu/10.1109/CAIDA51941.2021.9425117

Ivan Vasilev, Daniel Slater, Gianmario Spacagna, Peter Roelants, & Valentino Zocca. (2019). Python Deep Learning : Exploring Deep Learning Techniques and Neural Network Architectures with PyTorch, Keras, and TensorFlow, 2nd Edition: Vol. Second edition. Packt Publishing.

Layode, O., Alam, T., & Rahman, M. M. (2019). Deep Learning Based Integrated Classification and Image Retrieval System for Early Skin Cancer Detection. 2019 IEEE Applied Imagery Pattern Recognition Workshop (AIPR), Applied Imagery Pattern Recognition Workshop (AIPR), 2019 IEEE, 1–7. https://doi-org.proxy-bc.researchport.umd.edu/10.1109/AIPR47015.2019.9174586

Makwana, K. (2021, June 2). Frequent category imputation (missing data imputation technique). Medium. Retrieved November 3, 2021, from https://medium.com/geekculture/frequent-category-imputation-missing-data-imputation-technique-4d7e2b33daf7.

Rashid, H., Tanveer, M. A., & Aqeel Khan, H. (2019). Skin lesion classification using gan based data augmentation. *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. https://doi.org/10.1109/embc.2019.8857905

Reddy, M.A., Encoding Categorical data in Machine Learning, 2019, https://medium.com/bycodegarage/encoding-categorical-data-in-machine-learning-def03ccfbf40

Salian, A.C., Vaze, S., Singh, P., Shaikh, G.N., Chapaneri, S., Jayaswal, D., "Skin lesion classification using deep learning architectures," in 2020 3rd International Conference on Communication System, Computing and IT Applications (CSCITA). IEEE, 2020, pp. 168–173. https://ieeexplore.ieee.org/document/9137810

Sandler, M., Howard A., Zhu, M., Zhigunov, A.,Chen, L.C., MobileNetV2: Inverted Residuals and Linear Bottlenecks, The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2018, pp. 4510-4520 https://arxiv.org/abs/1801.04381

Sharma, A., ML from Scratch-Multinomial Logistic Regression, 2020, https://towardsdatascience.com/ml-from-scratch-multinomial-logistic-regression-6dda9cbacf9d

Sheha, M.A., Mabrouk, M.S., Sharawy, A., Automatic detection of melanoma skin cancer using texture analysis. International Journal of Computer Applica-tions, 2012, 42(20), 22-26, https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.259.2922&rep=rep1&type=pdf

Slamdot, I. (*n.d.*). What's the difference between melanoma and non-melanoma skin cancer? Premier Surgical Associates. Retrieved December 5, 2021, from https://www.premiersurgical.com/01/whats-the-difference-between-melanoma-and-non-melanoma-skin-cancer/.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., Going deeper with convolutions, (2014), https://arxiv.org/abs/1409.4842#

Torchvision.models. torchvision.models - Torchvision 0.11.0 documentation. (n.d.). Retrieved November 7, 2021, from https://pytorch.org/vision/stable/models.html.

Tschandl, P., Codella, N., Akay, B. N., Argenziano, G., Braun, R. P., Cabo, H., Gutman, D., Halpern, A., Helba, B., Hofmann-Wellenhof, R., Lallas, A., Lapins, J., Longo, C., Malvehy, J., Marchetti, M. A., Marghoob, A., Menzies, S., Oakley, A., Paoli, J., … Kittler, H. (2019). Comparison of the accuracy of human readers versus machine-learning algorithms for pigmented skin lesion classification: an open, web-based, international, diagnostic study. Lancet Oncology, 7, 938.

Tsuruoka, Y., Tsujii, J., Ananiadou, S., Stochastic Gradient Descent Training for L1-regularized Log-linear Models with Cumulative Penalty, *n.d*., https://aclanthology.org/P09-1054.pdf

Yiu, T., Understanding Random Forest, 2019, https://towardsdatascience.com/understanding-random-forest-58381e0602d2