A DEEP LEARNING BASED CLINICAL DECISION SUPPORT SYSTEM FOR MALARIA DIAGNOSIS AND DETECTION

ABIODUN ADEYINKA

ACETEL-NOUN

AABIODUN@NOUN.EDU.NG

OUTLINE

- Background
- Motivation
- Methodology
- Results
- Study relevant

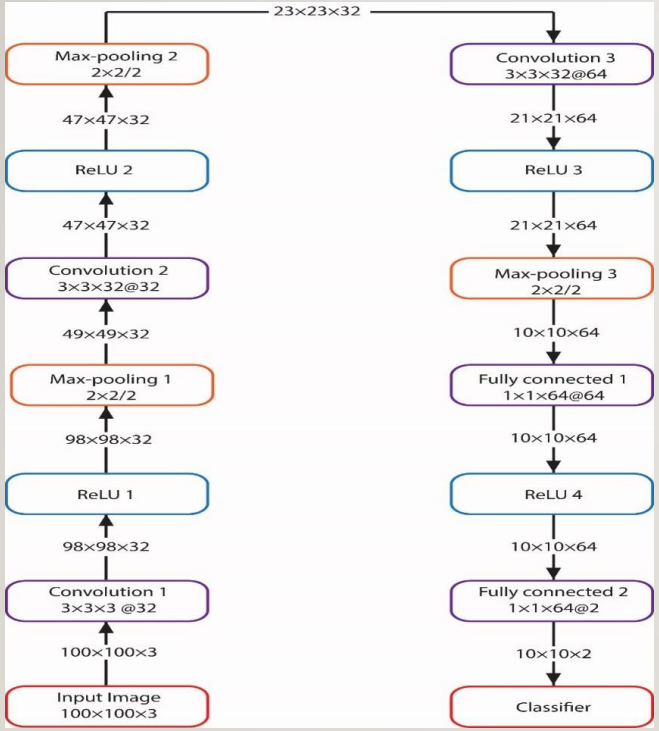
BACKGROUND

- In recent years, there has been a substantial resurgence in the optimistic disposition in the use of artificial intelligence tools and techniques in healthcare
- There are several challenging health problems that need addressing utilizing public health rules in order to improve stability.

MOTIVATION

- A report on non-communicable diseases by the World Health Organization stated that Nigeria and other developing countries are the worst impacted with a high mortality rate from non-communicable diseases
- Wrong diagnosis will lead to wrong treatment and eventually lead to more mortality coupled with the fact that the need for experts far outmatches the available supply in the developing nations There is therefore a need is to implement a deep learning-based clinical decision support system that is capable of recognizing patterns in malaria patient symptoms to make a possible diagnosis so as to assist and hasten a suitable course of action.

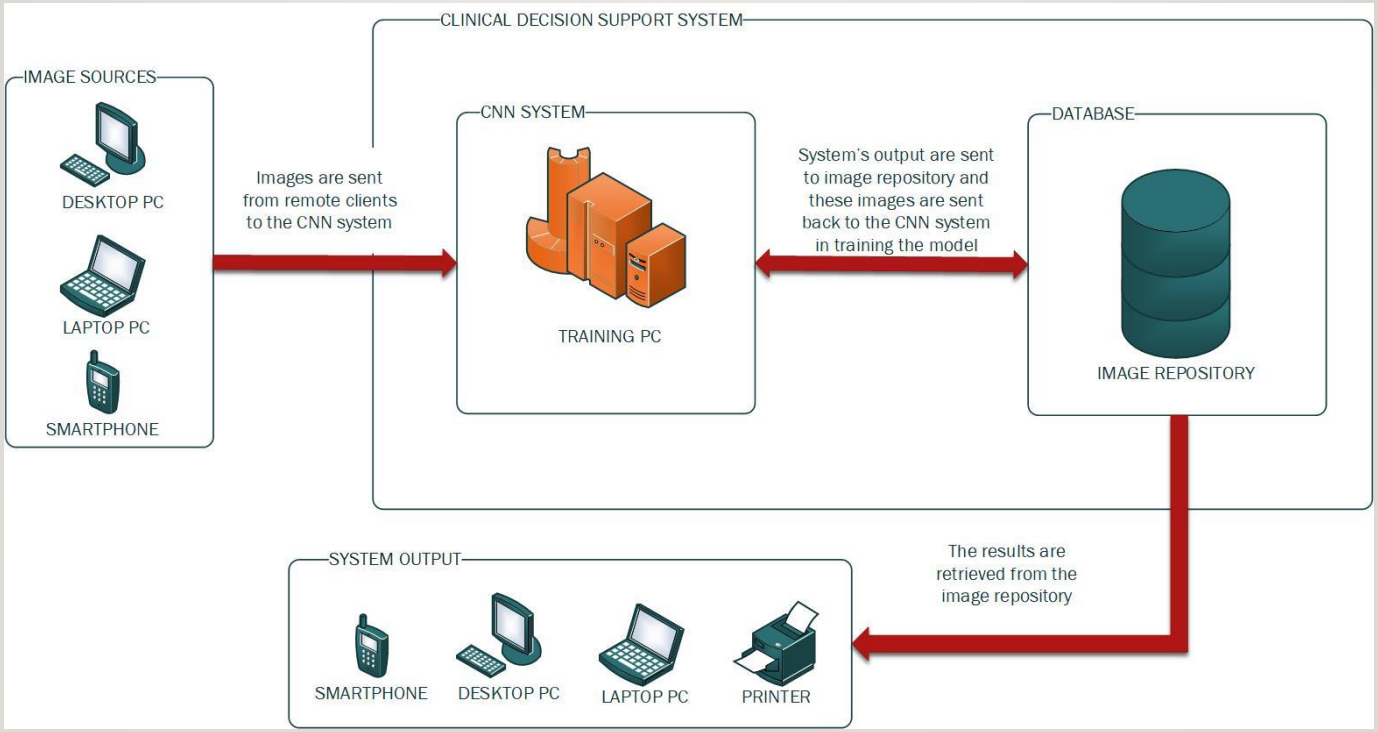
METHODOLOGY



This shows various layers in the CNN, their input dimensions and output dimensions, the Rectified linear unit (ReLU) and the max-pooling layers used for the convolutional layers.

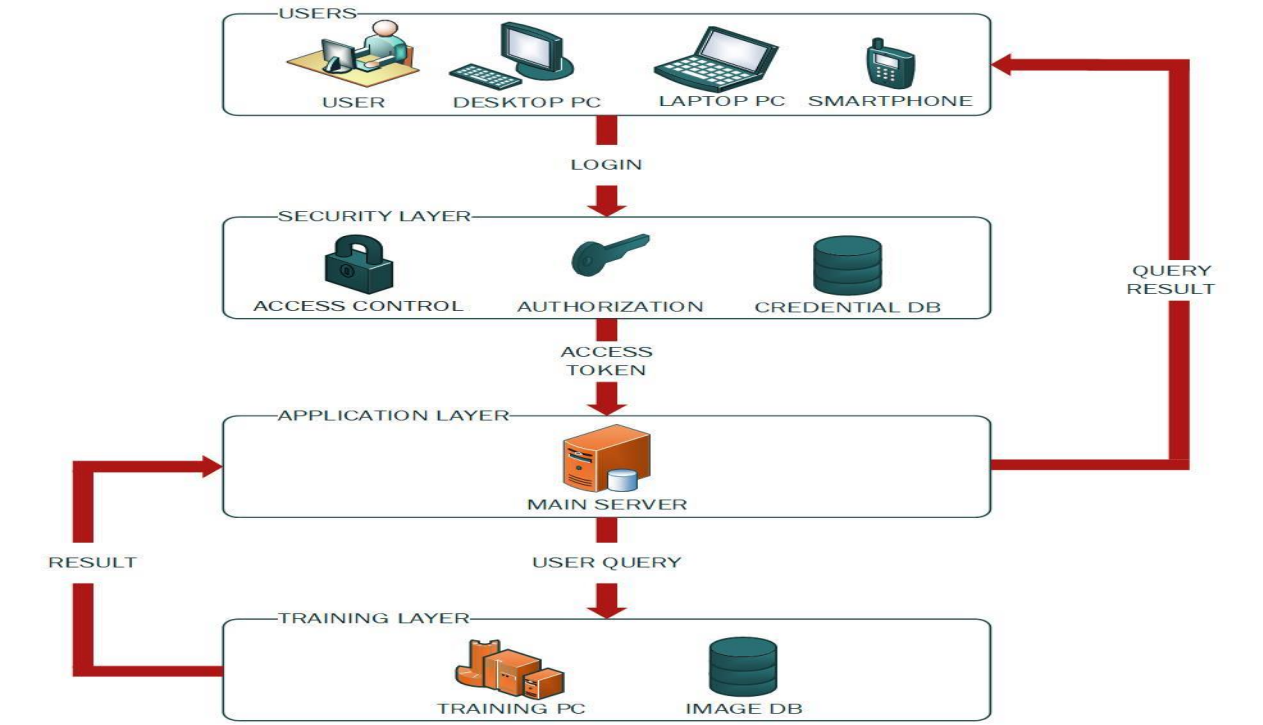
The Clinical Decision Support System (CDSS) consists of the main server that is comprised of a database for storing the classified images and a central PC for training the CNN model. The input and output of the system are connected to the main server through the internet. The patients digitalize blood smear images were used as input to the CDSS. The images were re-sampled to 100 × 100 pixels to match the dimensions of the dataset images that were used in training the CNN model

METHODOLOGY



Conceptual frameworks of the Clinical Decision Support System

METHODOLOGY



Data flow in the Clinical Decision Support System. This shows the information one layer to another in the Clinical Decision Support System. The security layer controls access into the application system. The application layer sends user queries to the training layer and receives the results from the training layer. This result is subsequently sent to the privileged user by the application layer.

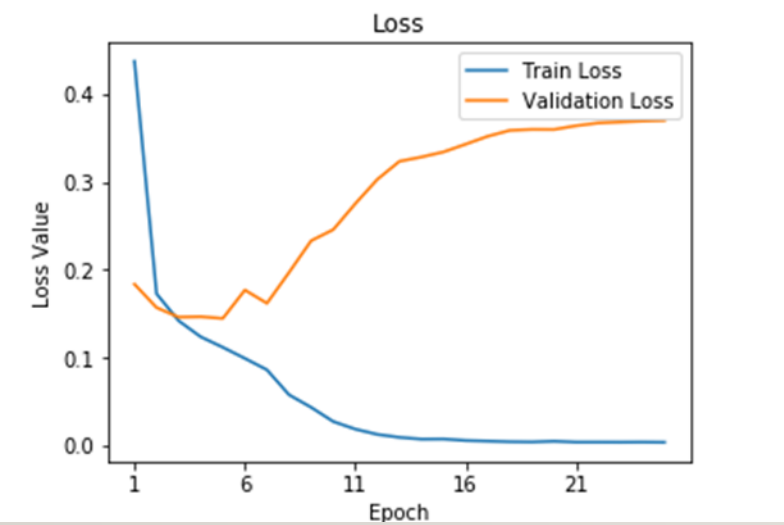
RESULTS

```
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
input_1 (InputLayer)         [(None, 125, 125, 3)]     0
_________________________________________________________________
conv2d (Conv2D)              (None, 125, 125, 32)      896
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 62, 62, 32)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 62, 62, 64)        18496
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 31, 31, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 31, 31, 128)       73856
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 15, 15, 128)       0
_________________________________________________________________
flatten (Flatten)            (None, 28800)             0
_________________________________________________________________
dense (Dense)                (None, 512)               14746112
_________________________________________________________________
dropout (Dropout)            (None, 512)               0
_________________________________________________________________
dense_1 (Dense)              (None, 512)               262656
_________________________________________________________________
dropout_1 (Dropout)          (None, 512)               0
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 513
=================================================================
Total params: 15,102,529
Trainable params: 15,102,529
Non-trainable params: 0
```

The layers of the CNN model were built using Keras which runs on Tensorflow. All layer specifications were done using Keras and the CNN model was compiled after.

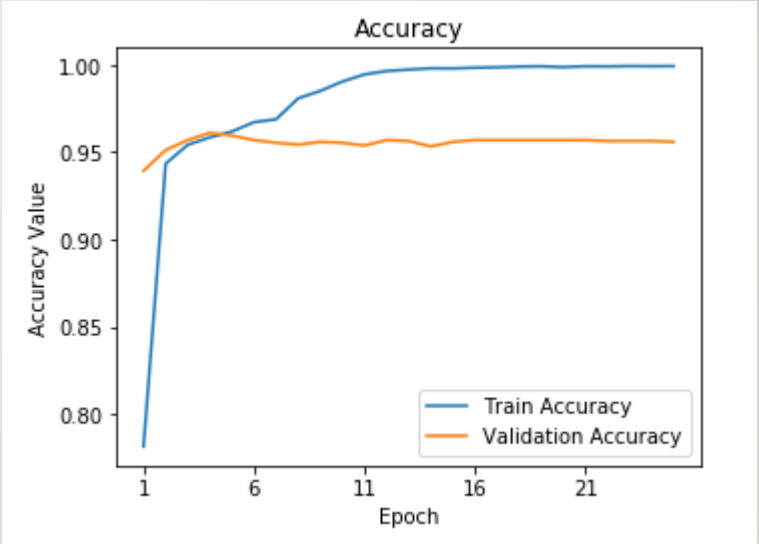
The CNN model architecture after all the layers were built using their respective specifications and compiled. The model was trained on the dataset using keras. Using specified batch size, epoch, class numbers and input dimensions. The model is the validated using the validate dataset and evaluate its performance.

RESULTS

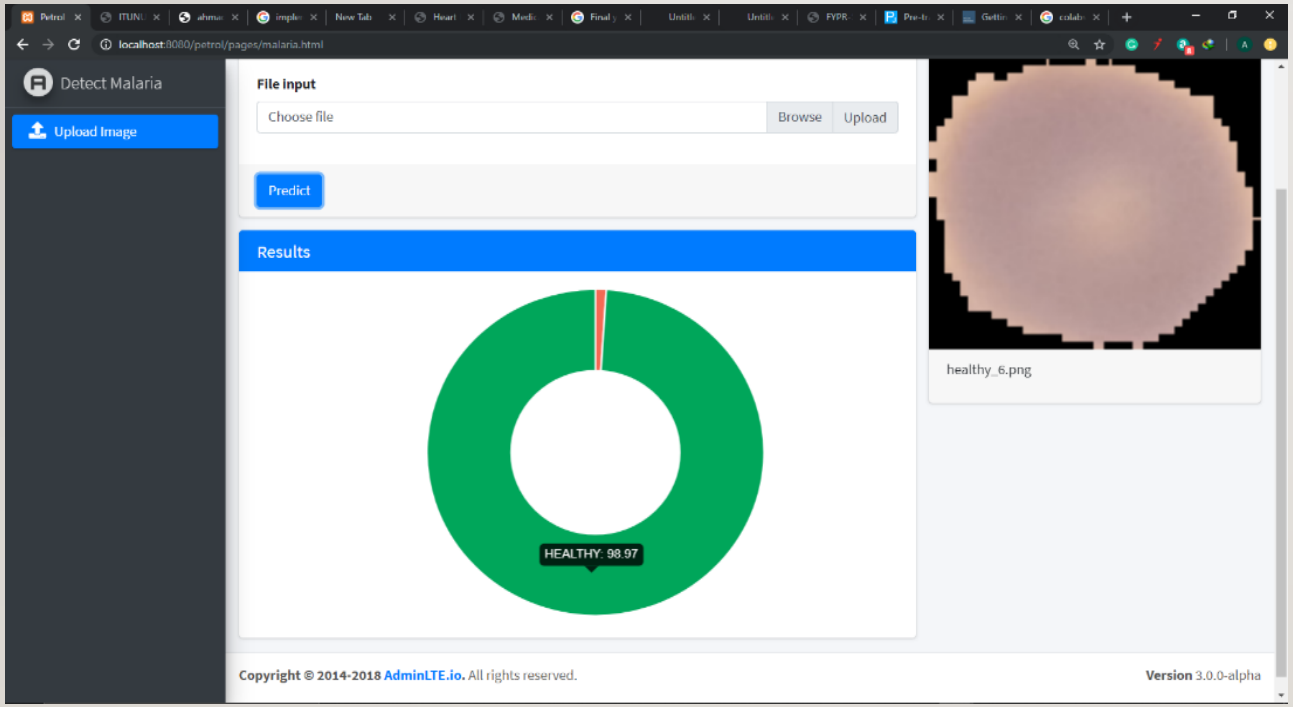


Graph of the accuracy at each epoch for both training and validation datasets

The plot of the accuracy of the model against the number of epoch. This increases as the model learns the dataset. The blue line is the graph for the training dataset and the orange line is for the validation dataset



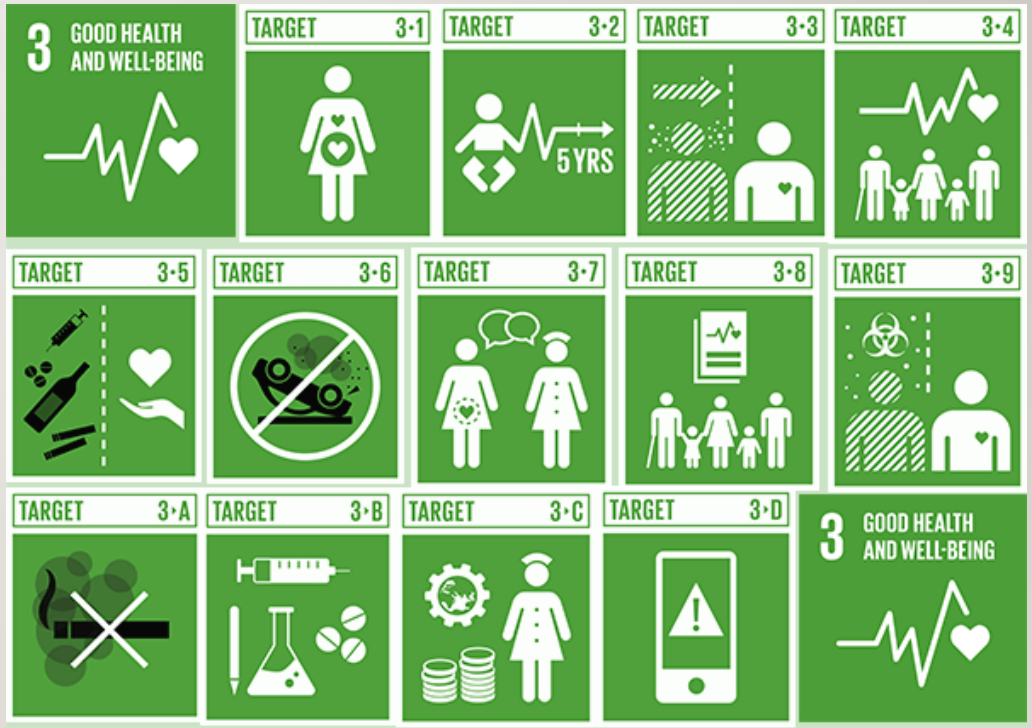
RESULTS



The CNN model was trained by using stochastic gradient descent (SGD) and Nesterov's momentum to optimizing the multinomial logistic regression objective. The proposed model achieved a training accuracy of 99%, validation accuracy of 97%, 40% train loss, 35% validation loss and a 98% prediction.

STUDY RELEVANT THEME

STUDY RELEVANT THEME




Thank you!