## Data Description

## Preprocessing

The image paths from the dataset were loaded from the default train and test splits according to the different classes and reshuffled to ensure randomness of the data. The shuffled paths and corresponding labels were then split into three data frames as train, validation and test. The training data consists of 80% of the entire dataset, with 20% of it set aside for validation during training.

The images were then loaded as 3D arrays with the OpenCV library. The arrays were then configured to be passed through a data generator using the Tensorflow library. The data generator allows for on-demand image processing and data randomization as well as augmentation. To ensure better model training, the pixel data was normalized by dividing the arrays by 255. In addition, the generator function was set up to optionally enable image augmentations such as shearing, zooming, horizontal flips, etc.

## Model Training

The different models trained showed varying accuracy in classifying the different species of malaria parasite the red blood cells.

Data keypoints:

* It was observed that extra augmentation save for pixel scaling, resulted in poorer model performance. This might be due to the fact that the dataset isn’t very large and so each augmented image distorts the reality of the data rather than to enhance variation.

Architecture Key points:

Objective one has been achieved, although more experimentation is still in the works

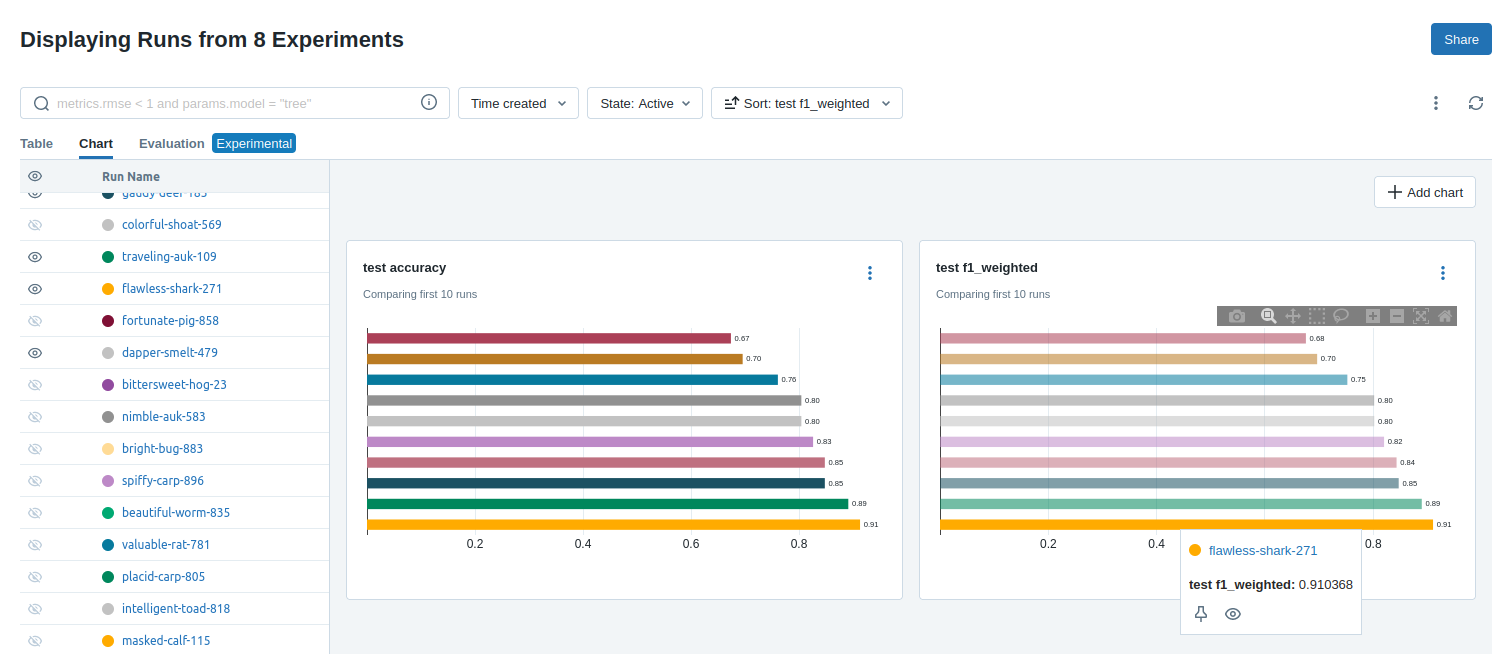
* CNN dilation rate was varied between (2,2) and (5,5): increases from 2 to 3 and 3 to 4 showed positive effect on model performance
* Several attention mechanisms were experimented with, including global attention, self attention, squeeze and excitation attention block, as well as convolution block attention mechanism (CBAM): self attention appears to be the most promising attention mechanism
* Transfer learning for feature extraction with VGG16 model significantly improves model performance compared to training from scratch

Performance metrics:

Objectives two and four have been achieved, however progress with the architecture may improve outcomes

* Accuracy and weighted F1 scores are used to measure model performance: This allows tracking the performance of the model based on each class, to more easily mitigate class imbalance and bias.
* The models are trained with early stopping to prevent overfitting
* The models are trained with checkpointing, to ensure that the model weights from the most optimal training epoch are saved.

Sample metrics:



Maximum observed F1 score so far was ~0.91, although the more stable result with re-runs is around ~0.84 and ~0.86

Model explainability (localizing classification):

This covers objective number three and will subsequently be implemented, as previous efforts have been channeled towards achieving feasible model architecture.

## To-Do:

* ~~Implementation of Grad-Cam mapping system to localize classification~~
* ~~Experimentation with other pretrained models for feature extraction with a focus on modelt trained medical data~~
* ~~Experiementation with the convolution blocks, as well as regularization and dropout to prevent data memorization.~~

## Experiment Schedule

Experiment | Dilation Rate | Attn Mechanism | Attention Units / Ratio

------------ |---------------- |---------------------- |--------------------------

* ~~Experiment 1 | 2 | Self-Attention | 128~~
* ~~Experiment 2 | 2 | Self-Attention | 256~~
* ~~Experiment 3 | 2 | Self-Attention | 512~~
* ~~Experiment 4 | 3 | Self-Attention | 128~~
* ~~Experiment 5 | 3 | Self-Attention | 256~~
* ~~Experiment 6 | 3 | Self-Attention | 512~~
* ~~Experiment 7 | 4 | Self-Attention | 128~~
* ~~Experiment 8 | 4 | Self-Attention | 256~~
* ~~Experiment 9 | 4 | Self-Attention | 512~~
* ~~Experiment 10 | 5 | Self-Attention | 128~~
* ~~Experiment 11 | 5 | Self-Attention | 256~~
* ~~Experiment 12 | 5 | Self-Attention | 512~~
* ~~Experiment 13 | 2 | Squeeze and Excite | 4~~
* ~~Experiment 14 | 2 | Squeeze and Excite | 8~~
* ~~Experiment 15 | 2 | Squeeze and Excite | 16~~
* ~~Experiment 16 | 3 | Squeeze and Excite | 4~~
* ~~Experiment 17 | 3 | Squeeze and Excite | 8~~
* ~~Experiment 18 | 3 | Squeeze and Excite | 16~~
* ~~Experiment 19 | 4 | Squeeze and Excite | 4~~
* ~~Experiment 20 | 4 | Squeeze and Excite | 8~~
* ~~Experiment 21 | 4 | Squeeze and Excite | 16~~
* ~~Experiment 22 | 5 | Squeeze and Excite | 4~~
* ~~Experiment 23 | 5 | Squeeze and Excite | 8~~
* ~~Experiment 24 | 5 | Squeeze and Excite | 16~~
* ~~Experiment 25 | 2 | CBAM | 4~~
* ~~Experiment 26 | 2 | CBAM | 8~~
* ~~Experiment 27 | 2 | CBAM | 16~~
* ~~Experiment 28 | 3 | CBAM | 4~~
* ~~Experiment 29 | 3 | CBAM | 8~~
* ~~Experiment 30 | 3 | CBAM | 16~~
* ~~Experiment 31 | 4 | CBAM | 4~~
* ~~Experiment 32 | 4 | CBAM | 8~~
* ~~Experiment 33 | 4 | CBAM | 16~~
* ~~Experiment 34 | 5 | CBAM | 4~~
* ~~Experiment 35 | 5 | CBAM | 8~~
* ~~Experiment 36 | 5 | CBAM | 16~~

### Naming Conventions

Experiment naming convention:

{experiment number}\_{dilation rate}\_{attention mechanism}\_{attention units/ratio}

Eg "exp36\_5\_CBAM\_16"

Numbers between 128 and 512 at the end represent attention units for SA mechanism whilst numbers between 4 and 16 represent ratio for CBAM and SE mechanisms

Attention mechanisms follow this convention

SA = self attention

CBAM = convolution block attention mechanism

SE = squeeze and excite (attention mechanism)

Tracked with this coding:

mechs = {

'SA': 0,

'SE': 1,

'CBAM': 2

}

### Best Performing experiments:

CBAM:

* Exp 28
* Exp 27

SE:

* Exp 15
* Exp 16
* Exp 18

SA:

* Exp 7
* Exp 8
* Exp 9

### Parameters

input\_shape = (224, 224, 3)

input\_transform = 0

num\_classes = 4

num\_epochs = 100

lr = 0.001

batch\_size = 16

patience = 20

val\_split = 0.2

dilation\_rate = (dil, dil) // changes

ratio = 16 // changes

attention\_mechanism = mechs['CBAM'] //changes

attention\_units = 0 // changes

### Metrics (Measured with K-fold -> 5 folds)

Weighted F1 scores

Weighted Precision

Weighted Recall

Accuracy