Final Project JKT1-A

Group members:

Azhari Haris Al Hamdi

Bimantara Hanumpraja

Irfani Sakinah

R. Achmad Dadang Nur Hidayanto

Ocular Disease Recognition Using Multilabel Classification

Consideration of Dataset Choice

1. It is challenging, neither of us ever created machine learning model for multilabel classification before, mostly the case was multiclass problem where each item only belongs to one class.

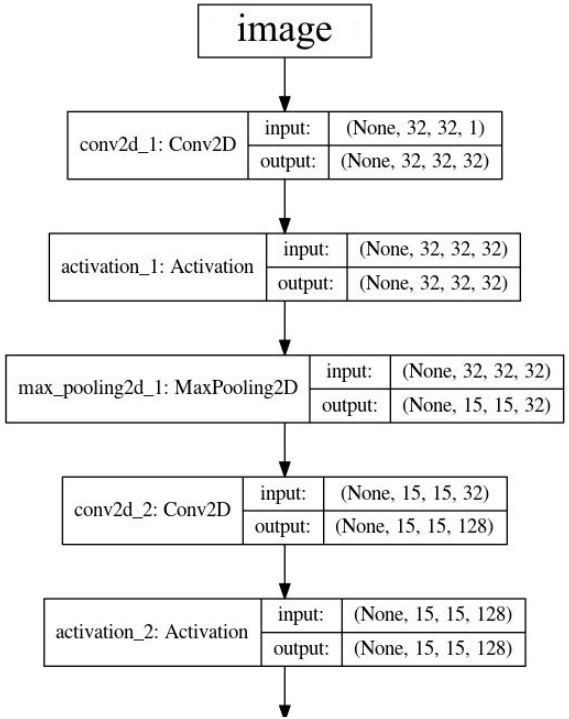
2. Ocular Disease Recognition is still needed to be improved because its recognition metric is still not approaching 99% in accuracy or F-measurement score based on recent research.

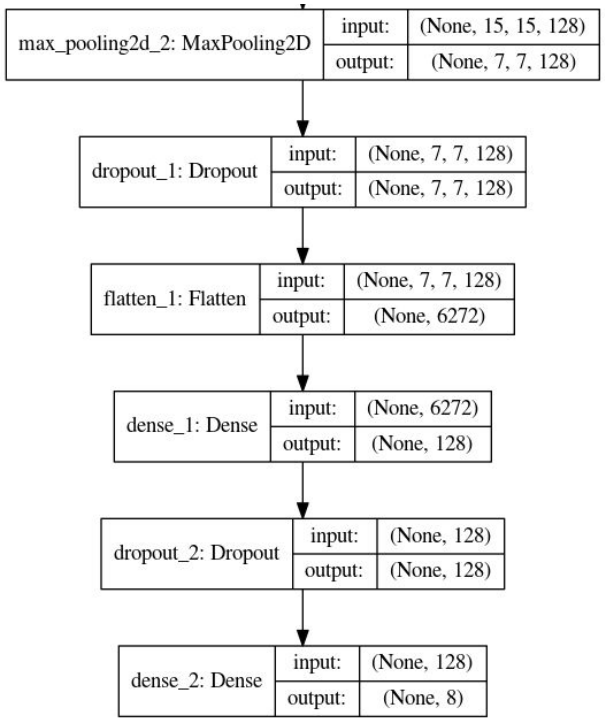
Baseline CNN Implementation

Conference Paper

- Title : Source and Camera Independent Ophthalmic Disease Recognition from Fundus Image Using Neural Network
- Authors : Islam et al.
- Date : November, 2019
- Kernel : https://www.kaggle.com/vinaypnair/odir-5k

Schematic Diagram of Model





Performance

- F-score : 0.85
- Kappa score : 0.31
- AUC value : 0.805

Consideration of Specific Improvement

- Speed of training
- Better image data for training
- Training Accuracy

Improvement

- Data Preprocessing
  - Separating unique keywords for each label
  - Contrast Limited Adaptive Histogram Equalization method for contrast enhancement
  - Image cropping and resize to reduce feature size for training
  - Image augmentation to generate larger dataset

- Training stream method

- Training time

Experiment Table

Experiment		1	2	3
Parameter	Batch size	32	32	32
	Learning rate	0.00005	0.001	1e-4
	Model architecture	VGG19, 256 neuron, 128 neuron,64 neuron, 8 neuron output	Base model - MobileNetV2, Layer 1 - 1024 neurons, Layer 2 - 8 neurons	VGG19, 256 neuron, 128 neuron,64 neuron, 8 neuron output
	Epoch	300	50	50
	Loss function	binary_crossentropy	binary_crossentropy	binary_crossentropy

Experiment		1	2	3
Metric (Validation)	Multi-label Accuracy	0.8842	0.8858	0.8952
	F1-score(Top-5)	0.5195	0.4998	0.4453
	AUC Value	0.7923	0.8171	0.7987

Experiment		4	2	3
Parameter	Batch size	32	32	32
	Learning rate	0.001	0.001	1e-4
	Model architecture	Model 4	Base model - MobileNetV2, Layer 1 - 1024 neurons, Layer 2 - 8 neurons	VGG19, 256 neuron, 128 neuron,64 neuron, 8 neuron output
	Epoch	50	50	50
	Loss function	binary_crossentropy	binary_crossentropy	binary_crossentropy
	Image	rescale=1./255, rotation_range=30, width_shift_range=0.2, height_shift_range=0.2,		

Experiment		4	2	3
Metric (Validation)	Multi-label Accuracy	0.8906	0.8858	0.8952
	F1-score(Top-5)	0.4152	0.4998	0.4453
	AUC Value	0.8145	0.8171	0.7987

Schematic Diagram of Chosen Model

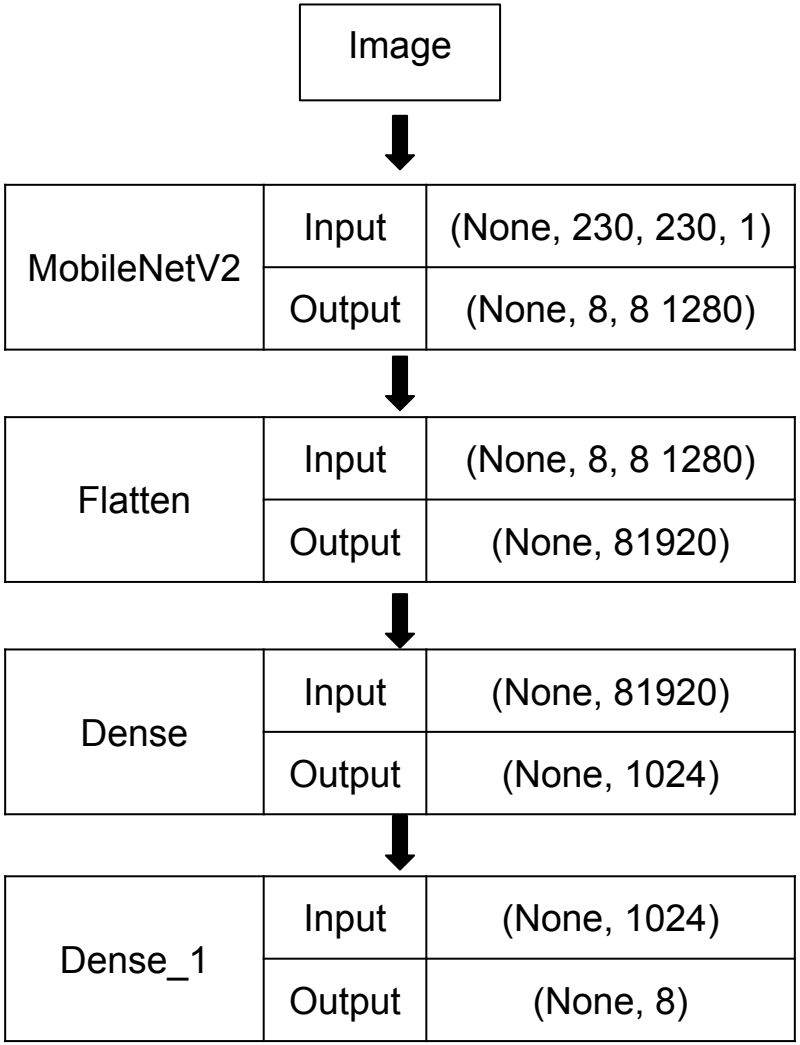


Image
MobileNetV2
Input
(None, 230, 230, 1)
Output
(None, 8, 8 1280)
Flatten
Input
(None, 8, 8 1280)
Output
(None, 81920)
Dense
Input
(None, 81920)
Output
(None, 1024)
Dense_1
Input
(None, 1024)
Output
(None, 8)

Documentation

- Clone this repository:
  ```
  $ git clone git@github.com:bangkitjkt/final-project-ODIR5K.git
  ```
- Run the model to train new data from model directory
  ```
  $ ipython
  $ %run choose_one_model.ipynb
  ```
- Extract the model into Tensorflow.js files:
  ```
  >> !pip install tensorflowjs
  >> import tensorflowjs as tfjs
  >> tfjs.converters.save_keras_model(model, 'path_to_tfjs/')
  ```
- Then, upload the model to the web server
  ```
  $ scp -r path_to_tfjs/ remote_username@server_ip:/path_to_model_site
  ```
- Change model url on index.js file (line 36921)
- Last, upload the web to your server
  ```
  $ scp -r web/ remote_username@server_ip:/path_to_web
  ```

  Demo site: https://odir.simulasikan.com/

Local Implementation

- It can be applied to recognize ocular diseases from fundus images collected in Indonesian eye hospitals.

- This project can be used as a reference to deal with another multilabel classification project.

Thank you!