

## 2nd Round User Interview Invitation - Machine Learning Engineer INDICO - Problem set 2 Anas Nafis Almustofa

- Image Classification

- a. Data prep and setup env

Download data from the kaggle competition

<https://www.kaggle.com/duttadebadri/image-classification>

Activate new python environment using conda or venv

Install requirements.txt using ``pip install -r requirements.txt``

Split image data into train and validation folder

Load data into .ipynb

- b. Data Analysis

The dataset has image 31775, which contains:

- Architecture: 7570
- Art and culture: 6744
- Food and drinks: 8800
- Travel and adventure: 31775

- c. Data preprocessing

Data augmentation for training data. The augmentation includes:

- Rotation
- Width shift
- Height shift
- Shear
- Zoom

Image rescaling, reduce pixel value into 0 - 1 value

- d. Model training

We used CNN to classify the model. Below are the architecture of the our CNN model:

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d_3 (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_4 (Conv2D)	(None, 72, 72, 64)	18,496
max_pooling2d_4 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_5 (Conv2D)	(None, 34, 34, 64)	36,928
max_pooling2d_5 (MaxPooling2D)	(None, 17, 17, 64)	0
flatten_1 (Flatten)	(None, 18496)	0
dense_2 (Dense)	(None, 64)	1,183,808
dense_3 (Dense)	(None, 4)	260

Hyperparameter that we use for this model are below:

- Optimizer: Adam

- Loss: Categorical crossentropy
- Metrics: accuracy
- Image input size: (150, 150)
- Epoch: 30

e. Training result

i. Best result

- accuracy: 0.7405
- loss: 0.6448
- val\_accuracy: 0.7444
- val\_loss: 0.7092



f. Possible Improvement

- Research more on suitable CNN architecture and optimizer
- Use bigger epoch to gain better result
- Use data batching to gain stable training metrics result
- Use callbacks function to stop training if the result of training is getting worst