



Implementasi Arsitektur Inception ResNet V2 Untuk Klasifikasi Citra Pada Data 4 Animal Classification

Latifatuzikra Suhairi
ASIMO

Latar Belakang

- Pada era sekarang, teknologi pengenalan citra menjadi salah satu inputan yang banyak diterapkan pada berbagai bidang, karena bermanfaat, salah satunya mampu mengenali dan mengklasifikasikan objek citra.
- Pengenalan citra menjadi salah satu alat penting yang sangat dibutuhkan bagi dunia penelitian.
- Salah satu permasalahan tersebut adalah banyaknya binatang yang ada dimuka bumi ini. Terdapat sekitar 8,7 Juta spesies, dimana satu spesies ada lebih dari beberapa ribu hingga beberapa juta jenis.
- Banyaknya jenis binatang dengan spesies yang banyak pula mengakibatkan butuh banyak waktu untuk dapat mengenali binatang tersebut secara manual.
- Melalui pengenalan jenis hewan yang dikembangkan secara komputerisasi diharapkan dapat digunakan untuk membantu peneliti dalam mengenali dan mengklasifikasikan makhluk-mahluk yang mereka jumpai.
- Dalam tulisan ini, pengenalan citra dibatasi untuk binatang anjing, kucing, rusa, dan kuda dengan menggunakan metode deeplearning CNN.
- Namun, deeplearning memerlukan jumlah data yang sangat banyak dalam proses pelatihan modelnya. Oleh karena itu, digunakan pendekatan Transfer Learning, model Inception ResNet V2

Rumusan Masalah

- Bagaimana mengembangkan model klasifikasi citra binatang menggunakan pendekatan transfer learning dengan model arsitektur Inception Resnet V2?
- Bagaimana akurasi dan loss model klasifikasi citra binatang yang telah dibangun?

Dataset

Dataset: 4-animal-classification.zip

- Berasal dari kaggle competitions. Link: <https://www.kaggle.com/competitions/4-animal-classification/data>
- Berisikan data berupa citra dan label yang menyatakan jenis binatangnya.
- Jenis binatang yang tersedia: cat, deer, dog, horse.
- File terbagi dalam dua folder: train dan test.

- Data train berjumlah 2800 citra
- Data test berjumlah 729 citra
- Data train terbagi atas 4 subfolder yang menyatakan class: cat, dog, deer, horse.
- Jumlah citra masing-masing class:

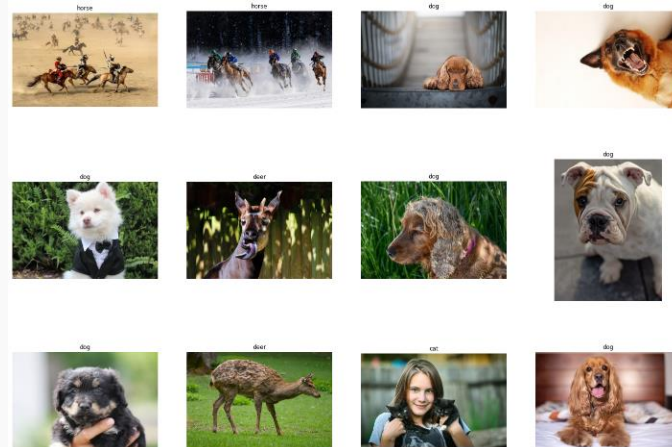
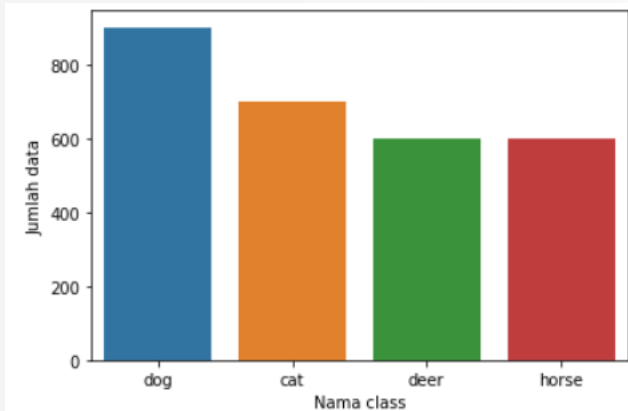
Nama class	Jumlah data
dog	900
cat	700
deer	600
horse	600

Preprocessing Data

1. Visualisasi Data

- Dilakukan untuk mengenali data citra.
- Hasil: mengetahui jumlah citra masing-masing class, mengetahui jumlah data train dan test, melihat beberapa citra untuk mendapatkan sense yang lebih baik.

There are 2800 number of images in train directory.
Total test images: 729



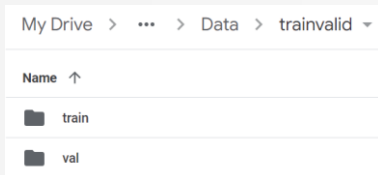
Preprocessing Data

2. Split data train dan data valid

- Dilakukan karena suatu pemodelan akan dibangun berdasarkan data train dan akan diuji validitasnya dengan data valid.
- Proses split data train dan data valid memiliki perbandingan 80:20
- Proses split menggunakan library split-folders

```
import splitfolders
input_folder = data_path + "Data/train"
output_folder = data_path + "Data/trainvalid"
splitfolders.ratio(input_folder, output=output_folder, seed=42, ratio=(0.8,0.2), group_prefix=None)
```

- Hasil:



Data train: 2240 untuk empat class

Data valid: 560 untuk empat class

Preprocessing Data

3. Augmentasi dan Normalize Data

- Augmentasi data bertujuan agar mesin dapat belajar dan mengenali dari berbagai citra yang berbeda-beda sekaligus bisa dimanfaatkan untuk memperbanyak data.
- Normalize data karena data akan diproses oleh CNN untuk improve performance secara keseluruhan. Disini kita akan menggunakan `rescale` parameter untuk scale our image pixel values dari $[0, 255]$ ke $[0,1]$.

Preprocessing Data

A. Augmentasi pada data train

Parameter augmentasi:

rescale = 1./255,
rotation_range=20,
width_shift_range=0.2,
height_shift_range=0.2,
horizontal_flip=True,
fill_mode='nearest'

B. Augmentasi pada data valid

Parameter augmentasi:

rescale = 1./255,

Penerapan augmentasi →

```
train_gen = train_datagen.flow_from_directory(  
    train_dir,  
    target_size=(img_height, img_width),  
    batch_size=32,  
    shuffle=True)  
  
val_gen = val_datagen.flow_from_directory(  
    valid_dir,  
    target_size=(img_height, img_width),  
    batch_size=20,  
    shuffle=False)
```


Model : Inception-ResNet V2

01

Model Inception-ResNet V2

Inception-ResNet-V2 adalah salah satu model transfer learning.

Kelebihan menggunakan transfer learning:

1. Menghemat waktu karena existing deeplearning model yang digunakan untuk Transfer Learning sudah memiliki weight dimana didalamnya ada data training hasil training sebelumnya.
2. Dataset yang digunakan berjumlah tergolong sedikit untuk dilakukan pembelajaran model yang mampu menghasilkan akurasi yang baik

Inception Resnet V2 : arsitektur CNN yang dibangun di atas keluarga arsitektur Inception tetapi menggabungkan koneksi residual (menggantikan tahap rangkaian filter dari arsitektur Inception).

Model : Inception-ResNet V2

02

Parameter

```
base_model = InceptionResNetV2(include_top=False, weights='imagenet', input_tensor=None, input_shape=(400, 400, 3))
```

```
# Flatten the output layer to 1 dimension
```

```
x = layers.Flatten()(base_model.output)
```

```
# Add a dropout rate of 0.5
```

```
x = layers.Dropout(0.5)(x)
```

```
# Add a final sigmoid layer with 1 node for classification output
```

```
output_layer = layers.Dense(4, activation='softmax')(x)
```

```
model = tf.keras.models.Model(base_model.input, output_layer)
```

```
for layer in base_model.layers:
```

```
    layer.trainable = False
```

```
# tell the model what cost and optimization method to use
```

```
model.compile(optimizer = Adam(learning_rate=1e-5),
```

```
              loss='categorical_crossentropy',
```

```
              metrics=['accuracy'])
```

```
img_height = 400
```

```
img_width = 400
```

```
#batchsize
```

```
batch_train = 32
```

```
batch_val = 20
```

Epoch=15

Ukuran Kebaikan Model

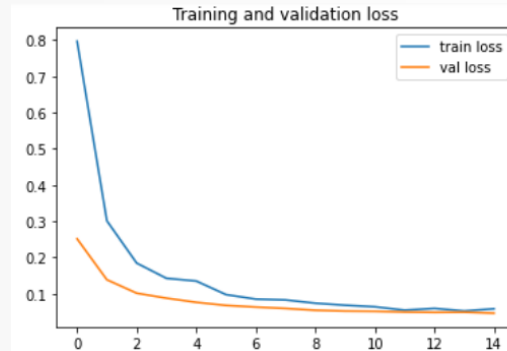
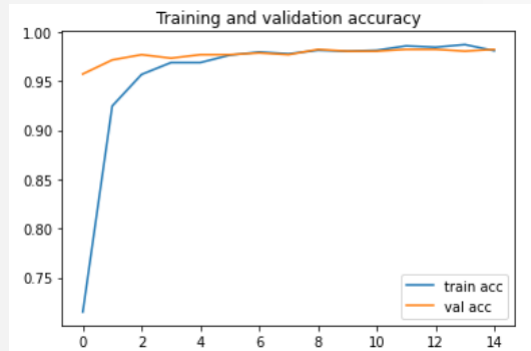
Menggunakan perhitungan skor akurasi data train dan data valid

Loss dan akurasi data train


```
70/70 [=====] - 91s 1s/step - loss: 0.0352 - accuracy: 0.9902  
[0.035230349749326706, 0.9901785850524902]
```

Loss dan akurasi data valid

```
28/28 [=====] - 10s 350ms/step - loss: 0.0461 - accuracy: 0.9821  
[0.046059977263212204, 0.9821428656578064]
```



Leaderboard



Create

Home

Competitions

Datasets

Code

Discussions

Learn

More

Your Work

RECENTLY VIEWED

View Active Events

Search

Overview

Data

Code

Discussion

Leaderboard

Rules

Team

My Submissions

Submit Predictions

...

Kesimpulan

01

Dari **4-animal-classification.xlsx**, dapat dibuatkan model klasifikasi citra menggunakan transfer learning model Inception Resnet V2. Dengan langkah: load data-preprocessing data (data split, augmentasi, normalize)- modelling (memanggil based model Inception Resnet V2, freeze layer, menambah layer dropout dan dense, model optimizer Adam)-evaluasi(perhitungan acc dan loss data train dan data valid)-prediksi data test

02

Klasifikasi citra menggunakan model Inception Resnet V2 menghasilkan model yang dapat dinyatakan goodfit dengan akurasi data train 99% dan data validation 98%. Sedangkan untuk loss data train 3% dan data validation 4%