

PEMBELAJARAN MESIN:
***CONVOLUTIONAL NEURAL NETWORK* UNTUK**
MENGLASIFIKASI DATA MNIST



Vincent Michael Sutanto

16/398531/PA/17492

PROGRAM STUDI ILMU KOMPUTER
DEPARTEMEN ILMU KOMPUTER DAN ELEKTRONIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS GADJAH MADA
YOGYAKARTA
2018

CONVOLUTIONAL NEURAL NETWORK UNTUK MENGLASIFIKASI DATA MNIST

Link Github : <https://github.com/vincentmichael089/CNN-MNIST>

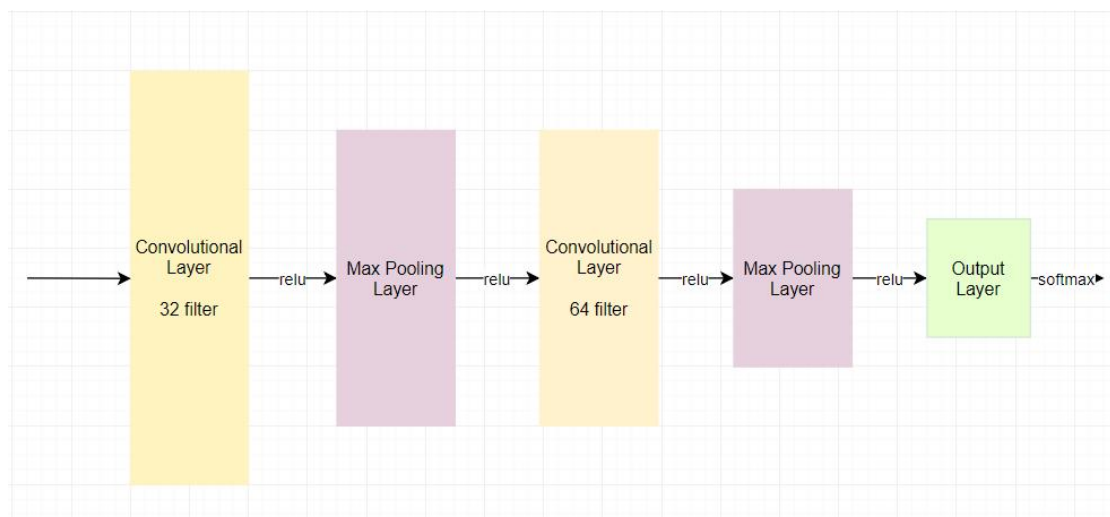
Bahasa Pemrograman : Python (*Jupyter Notebook*)

Library : Keras (*CNN*), Matplotlib (*Plotting*)

PENJELASAN

Convolutional Neural Network merupakan suatu prosedur dalam Pembelajaran Mesin yang digunakan ketika data berupa gambar menjadi masukan dari model. Prosedur ini muncul karena *Neural Network* klasik akan menghilangkan fitur keterkaitan antar pixel ketika merepresentasikan datanya, sehingga dinilai kurang baik ketika dihadapkan dengan data gambar.

Pada tugas kali ini, kembali digunakan MNIST dataset, dimana *Convolutional Neural Network* akan digunakan dalam mengklasifikasi dataset tersebut ke dalam 10 kelas yang berbeda (angka 1 sampai dengan 10). Arsitektur model yang digunakan adalah sebagai berikut:



Langkah-langkah yang dilakukan adalah sebagai berikut:

1. Seluruh data gambar diubah menjadi 28x28 dengan format *grayscale*
2. Seluruh elemen pixel pada seluruh data gambar di *re-scale* dalam rentang 0 sampai 1
3. Siapkan 10 class dengan *One-Hot Encoding*
4. Data di *Split* sebesar 54.000 data untuk *training* dan 6000 data untuk *validasi*
5. Membentuk CNN dengan arsitektur yang telah ditentukan, menggunakan fungsi aktivasi ReLU, fungsi *Loss* dengan *Cross-Entropy*, dan *Optimizer* menggunakan *Adam*
6. Model di train dan validasi dengan *batch size* = 128 dan *epochs* = 10
7. Model yang telah terbentuk kemudian dipakai untuk mengklasifikasi data *Testing*

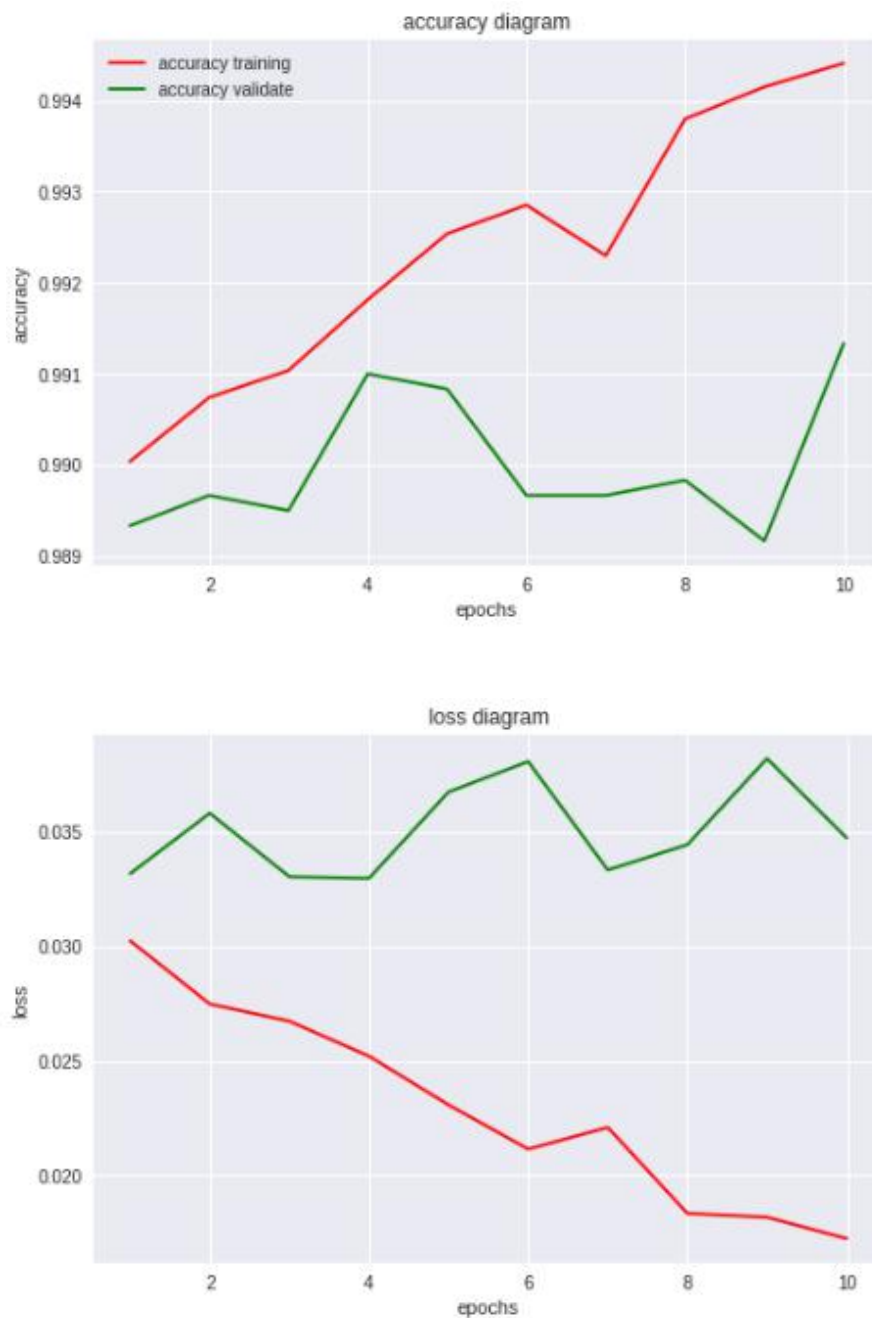
HASIL TRAINING, VALIDASI, DAN TESTING

Berikut disajikan *History Log* hasil *Training* dan *Validasi* model:

```
Train on 54000 samples, validate on 6000 samples
Epoch 1/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0302 - acc: 0.9900 - val_loss: 0.0332 - val_acc: 0.9893
Epoch 2/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0275 - acc: 0.9907 - val_loss: 0.0358 - val_acc: 0.9897
Epoch 3/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0267 - acc: 0.9910 - val_loss: 0.0330 - val_acc: 0.9895
Epoch 4/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0252 - acc: 0.9918 - val_loss: 0.0330 - val_acc: 0.9910
Epoch 5/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0231 - acc: 0.9925 - val_loss: 0.0367 - val_acc: 0.9908
Epoch 6/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0211 - acc: 0.9929 - val_loss: 0.0381 - val_acc: 0.9897
Epoch 7/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0221 - acc: 0.9923 - val_loss: 0.0333 - val_acc: 0.9897
Epoch 8/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0183 - acc: 0.9938 - val_loss: 0.0344 - val_acc: 0.9898
Epoch 9/10
54000/54000 [=====] - 71s 1ms/step - loss: 0.0182 - acc: 0.9941 - val_loss: 0.0382 - val_acc: 0.9892
Epoch 10/10
54000/54000 [=====] - 70s 1ms/step - loss: 0.0172 - acc: 0.9944 - val_loss: 0.0347 - val_acc: 0.9913
```

Dengan hasil *Testing* model menunjukkan tingkat *loss* sebesar 0.022120834507804922 dan tingkat akurasi sebesar 99.35%

DIAGRAM ERROR DAN AKURASI



KESIMPULAN

Tingkat akurasi mencapai 99.4% untuk *training* dan 99.1% untuk *validasi* menunjukkan model sudah termasuk kategori baik dalam mengklasifikasi data MNIST. Dari diagram yang disajikan, terlihat bahwa sebenarnya model mengalami *overfit*, namun hanya sedikit saja (*slightly overfit*).

REFERENSI PEMBELAJARAN

https://github.com/keras-team/keras/blob/master/examples/mnist_cnn.py

<https://medium.com/@mjbhobe/mnist-digits-classification-with-keras-ed6c2374bd0e>