

# 2016

## Deteksi Okupansi Ruang dengan Jaringan Saraf Tiruan

---

*Laporan Tugas Program Kecerdasan Artifisial*

Dosen Pengampu: Eko Prasetyo

Nama: Febrian Imanda Effendy

NIM: 1103134334

*4/30/2016*

---

## 1. Deskripsi Kasus

Mengimplementasikan Jaringan Saraf Tiruan (JST) untuk klasifikasi dalam mengidentifikasi okupansi ruangan. Arsitektur JST yang digunakan adalah Multi-Layer Propagation Feedforwards Network dengan algoritma pembelajaran Backpropagation.

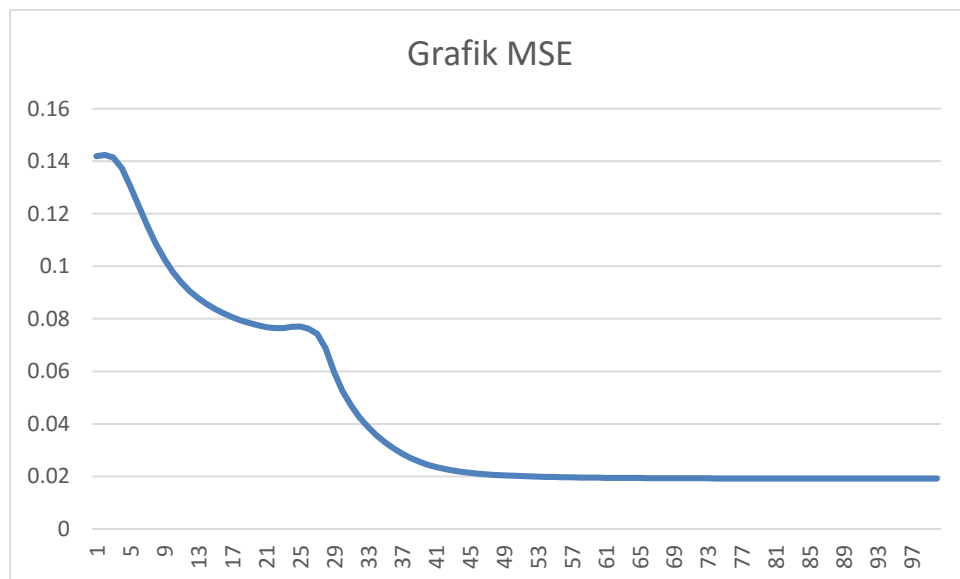
## 2. Data yang Digunakan

Data yang digunakan dalam studi kasus ini adalah data `occupancy.csv` dan `occupancy_normal.csv` yang didalamnya terdapat parameter “Temperature”, “Humidity”, “Light”, “CO2” dengan rentang nilai antara 0-1 dan kelas labelnya adalah “Occupancy” yang memiliki nilai 0 dan 1 saja. *Occupancy* = 0 menunjukkan ruangan yang kosong dan *occupancy* = 1 untuk ruangan yang sedang digunakan.

## 3. Desain Metode dan Implementasi Algoritma

### a. Parameter

- Jumlah Neuron = 4, karena setelah beberapa percobaan dengan hanya 4 neuron sudah menghasilkan akurasi yang cukup baik
- Learning Rate = 0.1, setelah beberapa percobaan untuk 0.2 hasil MSET tidak sebaik ketika 0.1
- Grafik untuk MSE



- Akurasi yang didapatkan adalah 98.35%
- JST untuk yang normal hasilnya lebih baik karena nilai MSE selalu turun untuk okupansi normal

## 4. Petunjuk Penggunaan Program

- Library yang digunakan adalah numpy, math dan csv.
- Install NumPy terlebih dahulu
- Untuk dataset yang dibaca dalam bentuk .csv
- Dari file csv akan diubah dalam bentuk dictionary dengan struktur data adalah { **n:[temp, humidity, light, co2]** } dan target dalam bentuk array dari occupancy [**occupancy1, occupancy2, ..., occupancyN**]

- Ketika ingin menjalankan program, hal yang dieksekusi adalah memanggil kelas JST dengan parameter inisiasiasinya adalah data, target, dan jumlah epochnya. **JST(data, target, nEpoch)**
- Kemudian panggil fungsi **learning()** untuk proses learning
- Ketika ingin dengan data test maka panggil fungsi **testing(data\_testing, target\_testing)** dengan memasukkan parameter data testing, dan target testing
- Setelah itu akan mendapatkan nilai akurasi dari program

## 5. Luaran Program

Output yang ditampilkan adalah :

- Proses Learning akan mengeluarkan hasil MSE setiap epochnya dan dibagian akhir akan mengeluarkan hasil w (bobot) yang paling terakhir.
- Proses testing akan mengeluarkan hasil akurasi di setiap pembacaan datanya.

Jelaskan output yang akan ditampilkan setelah program Anda dijalankan. Tulis dan jelaskan pula (contoh) solusi yang dihasilkan dari program Anda, sesuai studi kasus yang dihadapi.

## 6. Screenshot Program

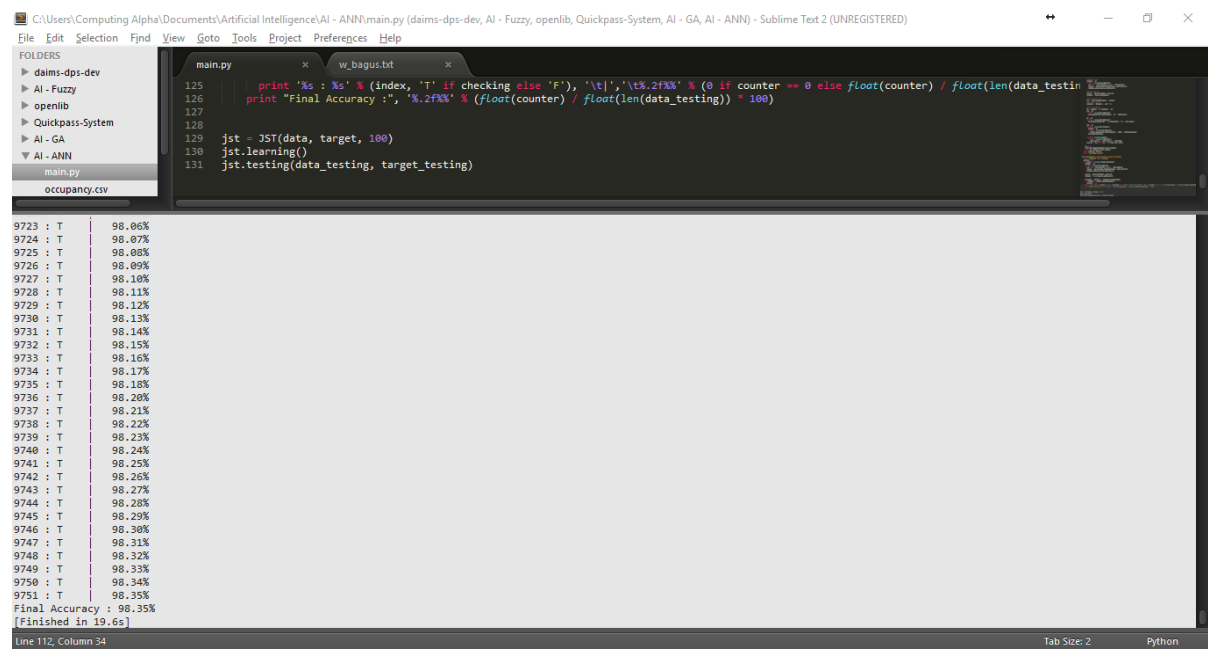
Bagian membaca dataset

```

C:\Users\Computing Alpha\Documents\Artificial Intelligence\AI - ANN\main.py (daims-dps-dev, AI - Fuzzy, openlib, Quickpass-System, AI - GA, AI - ANN) - Sublime Text 2 (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
└─ daims-dps-dev
└─ AI - Fuzzy
└─ openlib
└─ Quickpass-System
└─ AI - GA
└─ AI - ANN
  └─ main.py
    └─ occupancy_normal.csv
    └─ testing_occupancy_normal.csv
    └─ w_bagus.txt

main.py
1 import csv, math
2 import numpy as np
3
4 data = {}
5 target = []
6 with open('occupancy_normal.csv', 'rb') as csvfile:
7     reader = csv.DictReader(csvfile, delimiter=',')
8     i = 0
9     for row in reader:
10         data[i] = [float(row['Temperature']), float(row['Humidity']), float(row['Light']), float(row['CO2'])]
11         target.append(int(row['Occupancy']))
12         i += 1
13
14 data_testing = {}
15 target_testing = []
16 with open('testing_occupancy_normal.csv', 'rb') as csvfile:
17     reader = csv.DictReader(csvfile, delimiter=',')
18     i = 0
19     for row in reader:
20         data_testing[i] = [float(row['Temperature']), float(row['Humidity']), float(row['Light']), float(row['CO2'])]
21         target_testing.append(int(row['Occupancy']))
22         i += 1
23
9737 : T | 98.21%
9738 : T | 98.22%
9739 : T | 98.23%
9740 : T | 98.24%
9741 : T | 98.25%
9742 : T | 98.26%
9743 : T | 98.27%
9744 : T | 98.28%
9745 : T | 98.29%
9746 : T | 98.30%
9747 : T | 98.31%
9748 : T | 98.32%
9749 : T | 98.33%
9750 : T | 98.34%
9751 : T | 98.35%
Final Accuracy : 98.35%
[Finished in 19.6s]
Line 112, Column 34 Tab Size: 2 Python
  
```

## Bagian inialisasi



The screenshot shows a Sublime Text 2 editor window with the following content:

```
C:\Users\Computing Alpha\Documents\Artificial Intelligence\AI - ANN\main.py (daims-dps-dev, AI - Fuzzy, openlib, Quickpass-System, AI - GA, AI - ANN) - Sublime Text 2 (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
FOLDERS
  ▶ daims-dps-dev
  ▶ AI - Fuzzy
  ▶ openlib
  ▶ Quickpass-System
  ▶ AI - GA
  ▼ AI - ANN
    main.py
    occupancy.csv
main.py x w_bagus.txt x
125     print '%s : %s' % (index, 'T' if checking else 'F'), '\t', '%.2f%%' % (0 if counter == 0 else float(counter) / float(len(data_testin
126     print "Final Accuracy :", '%.2f%%' % (float(counter) / float(len(data_testing)) * 100)
127
128
129 jst = JST(data, target, 100)
130 jst.learning()
131 jst.testing(data_testing, target_testing)
```

The output of the script is displayed in the console:

```
9723 : T | 98.06%
9724 : T | 98.07%
9725 : T | 98.08%
9726 : T | 98.09%
9727 : T | 98.10%
9728 : T | 98.11%
9729 : T | 98.12%
9730 : T | 98.13%
9731 : T | 98.14%
9732 : T | 98.15%
9733 : T | 98.16%
9734 : T | 98.17%
9735 : T | 98.18%
9736 : T | 98.20%
9737 : T | 98.21%
9738 : T | 98.22%
9739 : T | 98.23%
9740 : T | 98.24%
9741 : T | 98.25%
9742 : T | 98.26%
9743 : T | 98.27%
9744 : T | 98.28%
9745 : T | 98.29%
9746 : T | 98.30%
9747 : T | 98.31%
9748 : T | 98.32%
9749 : T | 98.33%
9750 : T | 98.34%
9751 : T | 98.35%
Final Accuracy : 98.35%
[Finished in 19.6s]
```

Line 112, Column 34 Tab Size: 2 Python