

LAMPIRAN – LAMPIRAN

Lampiran 1. *Source Code* Fusi Informasi

```
# XOR gerbang logika
def XOR(x, y):
    if x != y:
        return '1'
    else:
        return '0'

# ciri difusikan menjadi 1 binary
def fusi_informasi(x):
    for i in range(len(x) - 1):
        temp = []
        for j in range(len(x[0])):
            if i == 0:
                temp += XOR(x[i][j], x[i + 1][j])
            else:
                temp += XOR(information_fusion[0][j], x[i + 1][j])
        information_fusion = ''.join(temp).split()
    return information_fusion
```

Lampiran 2. *Source Code* Preprocessing Data

```
# menambahkan comma setelah 2 angka
def add_dot_separator(binary):
    if len(binary) <= 1:
        for all_values in range(len(binary)):
            new_value = list(binary[all_values])
            new_value.insert(2, '.')
            new_value = ''.join(new_value).split()
    elif len(binary) > 1:
        new_value = list(binary)
        new_value.insert(2, '.')
        new_value = ''.join(new_value).split()
    return new_value

# convert comma binary menjadi angka desimal
def convert_decimal(binary):
    decimal = ''
```

```

positif = negatif = 0
value = 0.0
for i in range(len(binary)):
    decimal = binary[0]
decimal = list(decimal)
index_dot = decimal.index('.')
size_dot = decimal.index('.')
for i in range(len(decimal)):
    if i < index_dot:
        positif = positif + 1
        if decimal[i] == '1':
            value = value + pow(2, (size_dot - 1))
    elif i > index_dot:
        negatif = negatif + 1
        if decimal[i] == '1':
            value = value + pow(2, -negatif)
    size_dot -= 1
return value

```

Lampiran 3. Source Code Hamming Distance

```

def hammingDistance(value1, value2):
    distance = 0
    for i in range(len(value1)):
        for j in range(len(value1[0])):
            if value1[i][j] != value2[i][j]:
                distance += 1
    return distance

def get_min_hd(value1, value2):
    min_distance = 0
    value_min_distance = ''
    current_min_distance = hammingDistance(value1, value2[0])
    for i in range(len(value2) - 1):
        for j in range(len(value2[i])):
            for k in range(len(value2[i][j])):
                if value1[j][k] != value2[i + 1][j][k]:
                    min_distance += 1
            if min_distance < current_min_distance:
                current_min_distance = min_distance

```

```

        value_min_distance = value2[i + 1][j]
        min_distance = 0
    return value_min_distance

```

Lampiran 4. Source Code Backpropagation

```

import numpy as np
import time

def sigmoid(x):
    return 1 / (1 + np.exp(-x))

def deriv_sigmoid(x):
    return x * (1.0 - x)

class Neural_Network:
    def __init__(self, input, hidden, output, learningRate):
        self.n_input = input
        self.n_hidden = hidden
        self.n_output = output
        self.learning_rate = learningRate
        self.bias_hidden = np.ones((hidden, 1))
        self.bias_output = np.ones((output, 1))

        self.weight_hidden = np.random.uniform(low=-0.2, high=0.2,
size=(self.n_hidden, self.n_input))
        self.weight_output = np.random.uniform(low=-0.2, high=0.2,
size=(self.n_output, self.n_hidden))

    def forward(self, input_list):
        inputs = np.array(input_list, ndmin=2).T
        self.hiddenLayer = np.dot(self.weight_hidden, inputs) +
self.bias_hidden
        self.Net_hiddenLayer = sigmoid(self.hiddenLayer)

        self.outputLayer = np.dot(self.weight_output,
self.Net_hiddenLayer) + self.bias_output
        self.Net_outputLayer = sigmoid(self.outputLayer)
        return self.Net_outputLayer

```

```

def calc_loss(self, target_list):
    targets = np.array(target_list, ndmin=2).T
    m = targets.shape[0]
    cost = 0
    cost = cost + ((self.Net_outputLayer - targets) ** 2).sum() / m
    return cost

def backprop(self, input_list, target_list):
    inputs = np.array(input_list, ndmin=2).T
    targets = np.array(target_list, ndmin=2).T
    m = targets.shape[0]

    self.output_error = self.Net_outputLayer - targets
    self.output_error_weight = np.dot(self.output_error *
deriv_sigmoid(self.Net_outputLayer),
self.Net_hiddenLayer.T) * (1 /
m)
    self.output_error_bias = np.sum(self.output_error, axis=1,
keepdims=True) * (1 / m)

    self.hidden_error = np.dot(self.weight_output.T,
self.output_error)
    self.hidden_error_weight = np.dot(self.hidden_error *
deriv_sigmoid(self.Net_hiddenLayer), inputs.T) * (1 / m)
    self.hidden_error_bias = np.sum(self.hidden_error, axis=1,
keepdims=True) * (1 / m)

    def update_params(self):
        self.weight_output = self.weight_output - self.learning_rate *
self.output_error_weight
        self.weight_hidden = self.weight_hidden - self.learning_rate *
self.hidden_error_weight
        self.bias_output = sigmoid(self.bias_output - self.learning_rate *
self.output_error_bias)
        self.bias_hidden = sigmoid(self.bias_hidden - self.learning_rate *
self.hidden_error_bias)

    def predict(self, input_list, weight_hidden, weight_output,
bias_hidden, bias_output):
        inputs = np.array(input_list, ndmin=2).T

```

```

hiddenLayer = np.dot(weight_hidden, inputs) + bias_hidden
Net_hiddenLayer = sigmoid(hiddenLayer)

outputLayer = np.dot(weight_output, Net_hiddenLayer) + bias_output
self.Net_outputLayer = sigmoid(outputLayer)
return self.Net_outputLayer

def train_data(self, data_train, epochs):
    accu_train = []
    all_label = []
    for i in range(epochs):
        loss = 0
        startTime = time.time()
        for record in data_train:
            all_values = record.split(',')
            inputs = np.asfarray(all_values[1:])
            targets = np.zeros(self.n_output) + 0.01
            targets[int(all_values[0])] = 0.99
            correct_label = int(all_values[0])
            self.forward(inputs)
            loss = self.calc_loss(targets)
            self.backprop(inputs, targets)
            self.update_params()
            outputs = self.predict(inputs, self.weight_hidden,
self.weight_output, self.bias_hidden, self.bias_output)
            label = np.argmax(outputs)
            all_label.append(label)
            if (label == correct_label):
                accu_train.append(1)
            else:
                accu_train.append(0)
        if loss < 0.0001:
            break
        print("Epochs = ", i)
        print("Loss =", loss)
    if i % 10 == 0:
        print("Epochs = ", i)
        print("Loss =", loss)
    accu_array = np.asarray(accu_train)
    self.accu_train = accu_array.sum() / accu_array.size * 100

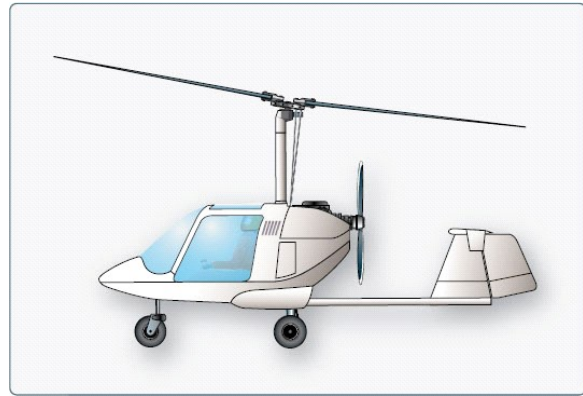
```

```
self.final_loss = loss
print("Epochs = ", i)
print("Loss =", loss)
print("Accu Train: ", self.accu_train, "%")
print('The script took {0} second !'.format(time.time() -
startTime))
```

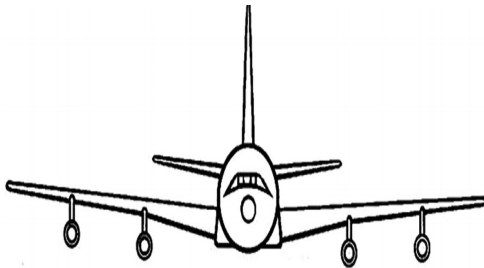
Lampiran 5. Gambar Fitur Pesawat



Fixed Wings



Rotary Wings



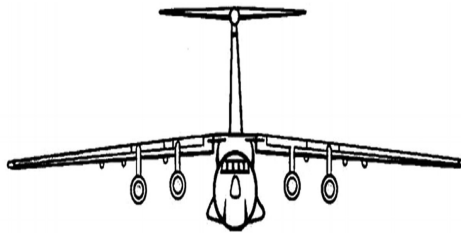
Low wing

Low Wings



Mid wing

Mid Wings



High wing

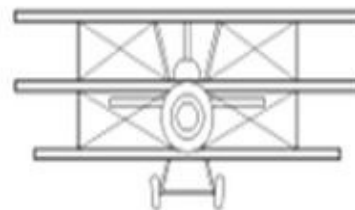
High Wings



Monoplane



Biplane



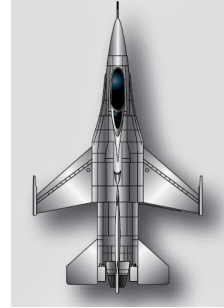
Triplane

Triplane

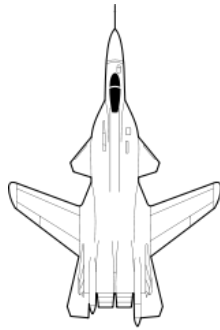


Straight wing

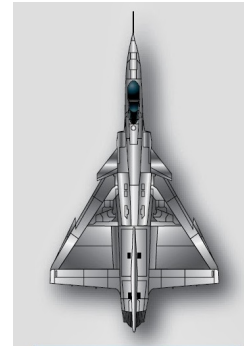
Straight Wings



Sweptback Wings



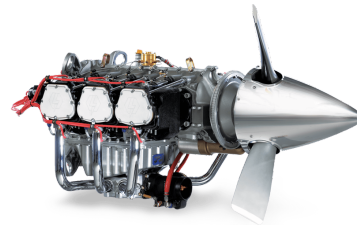
Forward Swept Wings



Delta Wings



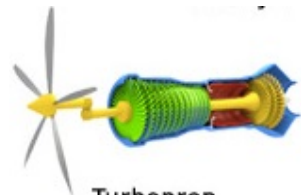
Rotor Wings



Piston

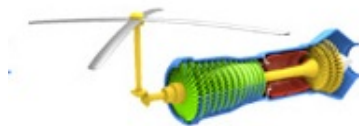


Turbojet



Turboprop

Turboprop



Turboshaft

Turboshaft



Turbofan



One Engine



Double Engine



Three Engine



Four Engine



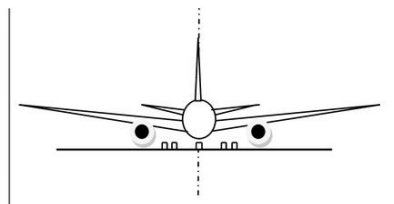
Five Engine



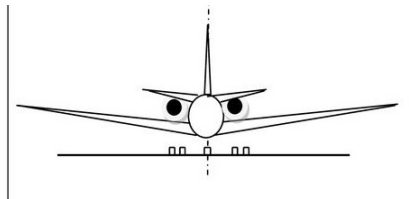
Six Engine



On The Wings



Behind Fuselage



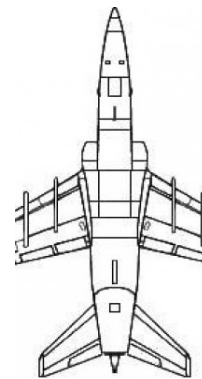
Above Fuselage



Behind Cabin



Above Cabin



In Fuselage



Front Fuselage



Close To The Fuselage



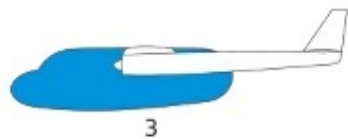
1

Subsonic Fuselage

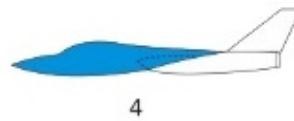


2

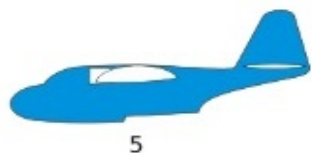
Supersonic Fuselage



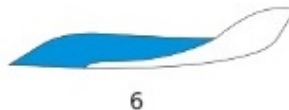
High Capacity Subsonic Fuselage



High Manuverability Fuselage



Flying Boat Fuselage



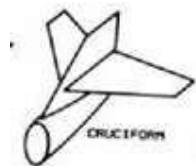
Hypersonic Fuselage



Dragonfly Fuselage



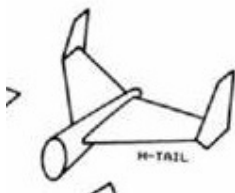
Conventional Tail



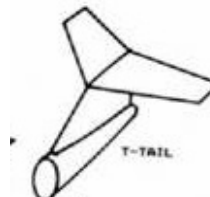
Cruciform Tail



No Tail (Double Main Rotor)



H Tail



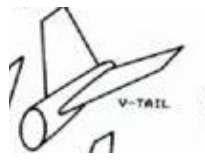
T Tail



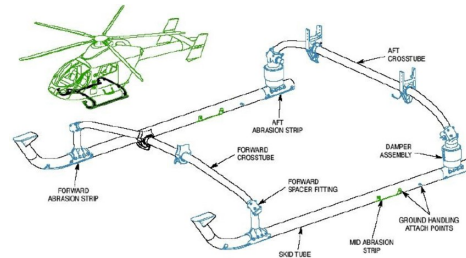
Rotor Tail



Twin Tail



V Tail



Skids Landing Gear



Canard

Lampiran 6. Log Bimbingan

8/21/21, 1:24 PM

Skripsi JTI



Sistem Skripsi JTI
Polinema v2.0 (Beta)

Ujian Akhir	Akun
Pembimbing	Proposal

Log Bimbingan

Catatlah selalu materi bimbingan maupun progress setiap kali Anda menemui dosen pembimbing atau ketika bimbingan secara daring.

Catatan:

- Jenis log bimbingan ada dua: **pra_proposal** dan **pasca_proposal**
- Ketika Anda **belum mengunggah revisi sempro**, maka jenis log bimbingan akan secara otomatis disimpan sebagai **pra_proposal**.
- Setelah revisi sempro dan seterusnya, jenis log bimbingan akan otomatis disimpan sebagai **pasca_proposal**
- Untuk dapat maju **seminar proposal**, minimal Anda harus melakukan 3 (tiga) kali bimbingan **pra_proposal** terverifikasi ke dosen **pembimbing utama** Anda.
- Untuk dapat maju **sidakng akhir** tahap 1, minimal Anda harus melakukan 10 (sepuluh) kali bimbingan yang **disetujui** ke dosen pembimbing utama dan 4 (empat) kali ke dosen pembimbing pendamping.
- Log bimbingan yang masih dalam tahap 'diajukan' masih bisa diedit maupun dihapus.
- Log bimbingan yang ditolak masih bisa diedit tetapi tidak bisa dihapus.
- Log bimbingan yang sudah diverifikasi tidak dapat dihapus maupun diedit.

Id	Dibangkitkan otomatis
Jenis	pasca_proposal
Nim mahasiswa	1741720101
Id dosen pembimbing	-- Pilih salah satu -- ▼
Tanggal	21 / 08 / 2021
Materi bimbingan dan/atau progres	

8/21/21, 1:24 PM

Skripsi JTI

Status	Diajukan
	Tambah Baru

Id	Jenis	Nama dosen pembimbing	Tanggal	Materi bimbingan dan/atau progres	Status	Aksi
7782	pasca_proposal	Arie Rachmad Syulistyo, S.Kom., M.Kom	2021-05-29	mengirimkan laporan skripsi terbaru tanggal 24 Mei (mengisi ulang karena beberapa log hilang sehabis pindah hosting)	disetujui	
7781	pasca_proposal	Arie Rachmad Syulistyo, S.Kom., M.Kom	2021-05-29	bimbingan offline tanggal 20 Mei (progress laporan skripsi) > memperbaiki tulisan yang salah > memperbaiki latar belakang > menghilangkan space pada sub bab yang terlalu lebar > memperbaiki bab 6 (mengisi ulang karena beberapa log hilang sehabis pindah hosting)	disetujui	
7780	pasca_proposal	Arie Rachmad Syulistyo, S.Kom., M.Kom	2021-05-29	mengirimkan progress laporan skripsi tanggal 8 mei (mengisi ulang karena beberapa log hilang sehabis pindah hosting)	disetujui	

tugasakhir.jti.polinema.ac.id/v2/pembimbing/log-bimbingan

2/6

8/21/21, 1:24 PM

Skripsi JTI

Id	Jenis	Nama dosen pembimbing	Tanggal	Materi bimbingan dan/atau progres	Status	Aksi
7408	pasca_proposal	Arie Rachmad Syulistyo, S.Kom., M.Kom	2021-04-29	mempresentasikan progress skripsi sampai saat ini dan menambahkan evaluasi untuk membandingkan hasil data	disetujui	
7315	pasca_proposal	Arie Rachmad Syulistyo, S.Kom., M.Kom	2021-04-26	mengajukan progres skripsi dalam bentuk laporan	disetujui	
6448	pasca_proposal	Arie Rachmad Syulistyo, S.Kom., M.Kom	2021-03-11	1. melaporkan progres skripsi paling baru 2. pengubahan form input menjadi dropdown / list gambar 3. penambahan kode untuk validasi 4. pengarahannya untuk menguji data training untuk memperoleh error terkecil	disetujui	
6949	pasca_proposal	Arie Rachmad Syulistyo, S.Kom., M.Kom	2021-04-07	mengajukan progress dalam bentuk laporan	disetujui	
6985	pasca_proposal	Arie Rachmad Syulistyo, S.Kom., M.Kom	2021-04-08	1. mempresentasikan progress program skripsi 2. mencari titik optimal data dan kombinasi data jumlah node dihidden layer dan learning rate	disetujui	
7783	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-05-29	menyerahkan laporan skripsi terbaru > menambahkan pengujian dengan data pesawat yang tidak ada di dataset > menambahkan cross validation > membuat aplikasi skripsi menjadi desktop (mengisi ulang karena beberapa log hilang sehabis pindah hosting)	disetujui	

Id	Jenis	Nama dosen pembimbing	Tanggal	Materi bimbingan dan/atau progres	Status	Aksi
5144	pra_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2020-12-10	1. menambahkan dan mengganti beberapa penelitian terdahulu 2. menambahkan manfaat dari penelitian yang dilakukan 3. menambahkan penjelasan tentang radar 4. perubahan cara sitasi jika lebih dari 3 orang	disetujui	
7458	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-05-01	menyerahkan laporan skripsi	disetujui	
7153	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-04-17	1. presentasi progress program skripsi 2. penambahan cara identifikasi pesawat dengan hamming distance untuk mencari bit yang terdekat	disetujui	
7151	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-04-17	Menyerahkan laporan progress skripsi	disetujui	
6948	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-04-07	mengajukan progress dalam bentuk laporan	disetujui	

8/21/21, 1:24 PM

Skripsi JTI

Id	Jenis	Nama dosen pembimbing	Tanggal	Materi bimbingan dan/atau progres	Status	Aksi
6779	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-03-30	--> Progress 1. mempresentasikan progress --> Next Progress 1. melakukan beberapa uji coba pada metode	disetujui	
6447	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-03-11	1. melaporkan progres skripsi paling baru 2. pengubahan form input menjadi dropdown / list gambar 3. penambahan kode untuk validasi 4. pengarahan untuk menguji data training untuk memperoleh error terkecil	disetujui	
6408	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-03-05	melaporkan progress skripsi	disetujui	
6363	pasca_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-02-16	Penambahan Dataset dan pengarahan untuk isi dari dataset	disetujui	
6108	pra_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-01-08	1. perbaikan pada slide presentasi sempro 2. penambahan perhitungan metode	disetujui	

8/21/21, 1:24 PM

Skripsi JTI

Id	Jenis	Nama dosen pembimbing	Tanggal	Materi bimbingan dan/atau progres	Status	Aksi
6058	pra_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2021-01-07	1. penambahan penelitian terdahulu pada state of the art 2. penambahan tabel ekstrasi fitur 3. penambahan cara fusi informasi 4. perbaikan flowchart	disetujui	
5287	pra_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2020-12-16	1. perbaikan penulisan bahasa asing 2. perbaikan pada gambar dan tabel 3. perbaikan pada dasar teori 4. penambahan pada metode dan diagram	disetujui	
5206	pra_proposal	Kolonel Lek. Dr. Ir. Arwin Datumaya Wahyudi Sumari, S.T., M.T., IPM., ASEAN Eng., ACPE.	2020-12-14	1. perbaikan state of the art 2. perbaikan diagram venn 3. perbaikan studi literatur	disetujui	

Copyright(c) 2021 by Tim LA-Skripsi JTI Polinema
 Developed by [YYN](#), powered by [M PHP Framework](#)

Lampiran 7. Log Revisi

8/21/21, 1:28 PM

Skripsi JTI



Sistem Skripsi JTI Polinema v2.0
(Beta)

Proposal	Ujian Akhir	Akun
		Pembimbing

Detail Revisi Ujian Akhir

Berikut ini adalah revisi yang diberikan penguji 1 dan 2 pada saat Anda ujian.

Catatan:

- Untuk dapat mengentrikan hasil revisi, nilai dari semua penguji harus sudah lengkap untuk semua mahasiswa yang terlibat pada LA/Skripsi ini.

[<< Back](#)

Semua nilai dan tandatangan berita acara sudah lengkap, Anda dapat mengentrikan revisi dengan mengklik tombol Unggah Revisi di bawah.

Nomor ujian	Tahap	Nim	Nama	Penguji 1	Revisi penguji 1	Penguji 2	Revisi penguji 2
709	Ujian Akhir D4 Tahap-1 (2021)	1741720101	dimas eka adinandra	Luqman Affandi, S.Kom., MMSI	di tiap gambar, harusnya ada penjelasannya biasakan sub bab ada kalimatnya flowchart, prosesnya tlg diperhatikan. hal 52, 55 terangkan flowchartnya munculnya erd harusnya ada dasar, dari dfd atau normalisasi cek kesimpulan dengan rumusan dan tujuan harusnya dimasukkan/dikelompokkan berdasarkan ciri2 ada contoh perhitungan?	Wilda Imama Sabilla, S.Kom., M.Kom	- perbaikan beberapa typo, misal :dibagian harusnya di bagian, dst - penulisan istilah menggunakan huruf italic, misal: artificial - pada tabel, ada kata yg terpotong, misal: backpropagation menjadi 2 baris, jika demikian harusnya diberi -, misal back-propagation - Gambar 2.1 penelitian saat ini bisa diberikan sedikit keterangan mengenai gambar tersebut - Gambar diagram blok font terlalu kecil, bisa diperbesar agar tidak kesulitan

tugasakhir.jti.polinema.ac.id/v2/ujian-akhir/revisi/detail?nomor_ujian=709

1/2

8/21/21, 1:28 PM

Skripsi JTI

Nomor ujian	Tahap	Nim	Nama	Penguji 1	Penguji 2	Revisi penguji 2
						<p>backpro dan fuzzy??? yg sudah tambah fuzzy tambah lama. sdh ada di penulisan mengapa ditambahkan metode?</p> <p>membaca - 4.5 Perancangan Flowchart: Gambar 4.3, 4.4 tidak jelas, jika ada detailnya digambarkan lg, jika tidak ada tidak perlu dituliskan, diagram alir buat seperti 4.3, ada kelihatan awalnya dan akhirnya kemana - Gambar database juga terlalu kecil - Seperti kata pak luqman, tampilan saat memasukkan fitur bisa dikelompokkan berdasarkan fitur utama agar tidak membingungkan - Di tabel pengujian ada MSE, apakepanjangan MSE? Belum dijelaskan, perhitungannya jg saya cari belum nemu di BAB 2 - Penjelasan jika ada data yang sama persis akan ditampilkan semua apa sudah ada? Kalau belum ada, tambahkan</p>

Unggah Revisi

Copyright(c) 2021 by Tim LA-Skripsi JTI Polinema
 Developed by [YYN](#), powered by [M.PHP Framework](#)