

**PERSONALISASI PENYEIMBANGAN SKENARIO BERBASIS PROFIL
PEMAIN PADA *GAME* PENDIDIKAN ISLAM DAN LINGKUNGAN HIDUP
MENGUNAKAN *FEEDFORWARD NEURAL NETWORK* (FFNN)**

SKRIPSI

Oleh:
NURLAILI KHAIRANI
NIM. 18650057



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022**

**PERSONALISASI PENYEIMBANGAN SKENARIO BERBASIS PROFIL
PEMAIN PADA *GAME* PENDIDIKAN ISLAM DAN LINGKUNGAN
HIDUP MENGGUNAKAN *FEEDFORWARD NEURAL NETWORK*
(FFNN)**

SKRIPSI

Oleh :
NURLAILI KHAIRANI
NIM. 18650057

**Diajukan Kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri (UIN) Malang
untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2022**

HALAMAN PERSETUJUAN

**PERSONALISASI PENYEIMBANGAN SKENARIO BERBASIS PROFIL
PEMAIN PADA *GAME* PENDIDIKAN ISLAM DAN LINGKUNGAN
HIDUP MENGGUNAKAN *FEEDFORWARD NEURAL NETWORK* (FFNN)**

SKRIPSI

Oleh :
NURLAILI KHAIRANI
NIM. 18650057

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal : 2 Desember 2022

Dosen Pembimbing I



Dr. Fresy Nugroho, M.T.
NIP. 19710722 201101 1 001


Dosen Pembimbing II



Dr. Muhammad Faisal, M.T.
NIP. 19740510 200501 1 007

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Fachrul Kurniawan, M.MT., IPM
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

**PERSONALISASI PENYEIMBANGAN SKENARIO BERBASIS PROFIL
PEMAIN PADA *GAME* PENDIDIKAN ISLAM DAN LINGKUNGAN HIDUP
MENGUNAKAN *FEEDFORWARD NEURAL NETWORK* (FFNN)**

SKRIPSI

Oleh:

NURLAILI KHAIRANI

NIM. 18650057

Telah Dipertahankan di Depan Dewan Penguji
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Pada Tanggal: 09 Desember 2022

Susunan Dewan Penguji

Ketua Penguji : Dr. Yunifa Miftahul Arif, M.T
NIP. 19830616 201101 004

Anggota Penguji I : Juniardi Nur Fadila, M.T
NIP. 19920605 201903 1 015

Anggota Penguji II : Dr. Fresy Nugroho, M.T
NIP. 19710722 201101 1 001

Anggota Penguji III : Dr. Muhammad Faisal, M.T
NIP. 19740510 200501 1 007



Mengetahui,
Ketua Proram Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Fachrul Kurniawan, M.MT., IPM
NIP. 19771020 200912 1 001

PERSYARATAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

Nama : Nurlaili Khairani
NIM : 18650057
Jurusan : Teknik Informatika
Fakultas : Sains dan Teknologi
Judul Skripsi : Personalisasi Penyeimbangan Skenario Berbasis Profil Pemain Pada *Game* Pendidikan Islam Dan Lingkungan Hidup Menggunakan *Feedforward Neural Network* (FFNN)

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 16 Desember 2022
Yang membuat pernyataan,



Nurlaili Khairani
NIM. 18650057

HALAMAN MOTTO

“Tidaklah mungkin bagi matahari mengejar bulan dan malam pun tidak dapat mendahului siang. Masing-masing beredar pada garis edarnya.”

(QS. Yasin ayat 40)

HALAMAN PERSEMBAHAN

Skripsi ini saya persembahkan untuk

Orang tua, Saudara, Keluarga,

Seluruh guru, dosen, Sahabat,

Teman-teman seperjuangan dan

Diri saya sendiri

Terima kasih..

KATA PENGANTAR

Assalamualaikum, Wr, Wb.

Puji dan syukur atas kehadiran Tuhan semesta alam Allah SWT yang telah memberikan Rahmat dan hidayah-Nya, sehingga peneliti diberikan kemudahan dan keberkahan dalam setiap menyelesaikan skripsi ini. Penyusunan skripsi ini bertujuan untuk memenuhi syarat kelulusan bagi mahasiswa Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang. Keberhasilan penulisan skripsi ini tidak lepas dari dorongan dan bimbingan dari berbagai pihak. Untuk itu dalam kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. H.M. Zainuddin, MA selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim.
2. Dr. Sri Hariani, M.Si selalu dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim.
3. Dr. Fachrul Kurniawan, M.MT., IPM selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Ibrahim Malang
4. Dr. Fresy Nugroho, M.T dan Dr. Muhammad Faisal, M.T selaku Dosen pembimbing yang telah bersedia meluangkan waktunya dalam membimbing dan memberikan dorongan dan arahan kepada peneliti sehingga dapat menyelesaikan skripsi ini.
5. Dr. Yunifa Miftachul Arif, M.T dan Juniardi Nur Fadila, M.T selaku Dosen penguji yang telah memberikan saran dan arahan kepada penulis dalam proses penyelesaian skripsi.

6. Ayah saya yaitu Bapak Usman dan Ibu saya yaitu Ibu Selamah yang telah memberikan dukungan dari segi ekonomi, moral, spiritual, dan doa yang tiada hentinya sehingga pengerjaan skripsi ini bisa berjalan lancar.
7. Kepada Kakak dan Kakak Ipar saya yang selalu memberikan motivasi dan nasihat.
8. Seluruh dosen dan staff Jurusan Teknik Informatika yang telah memberikan ilmu dan pengalaman yang berharga.
9. Teman sebangunan saya Seta Murdha Pamungkas dan Nurul Khafidoh yang selalu memberikan penjelasan, arahan dan pemahaman tentang penelitian skripsi ini.
10. Sahabat-sahabat saya, Asfilia Nova Anggraini, Maulida Dwi Ariani, dan Aulia Ananda Salsabila yang selalu membantu dan memberikan informasi terkait pengerjaan skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih jauh dari kata sempurna. Maka dari itu penulis akan menerima saran dan kritik yang membangun. Terlepas dari itu semua, penulis berharap semoga skripsi ini dapat memberikan manfaat kepada para pembaca.

Wassalamualaikum Wr. Wb.

Malang, 26 September 2022

Penulis

DAFTAR ISI

HALAMAN JUDUL	ii
HALAMAN PERSETUJUAN	Error! Bookmark not defined.
HALAMAN PERSETUJUAN	Error! Bookmark not defined.
HALAMAN PENGESAHAN	Error! Bookmark not defined.
PERSYARATAN KEASLIAN TULISAN	Error! Bookmark not defined.
HALAMAN MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiii
ABSTRAK	xiv
ABSTRACT	xv
مستخلص البحث	xvi
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	5
1.3 Tujuan Penelitian	5
1.4 Batasan Masalah	5
1.5 Manfaat Penelitian	6
BAB II STUDI PUSTAKA	7
2.1 Penelitian Terkait	7
2.2 Personalisasi	8
2.3 Penyeimbangan Skenario	9
2.4 Profil Pemain	13
2.5 <i>Game</i> Edukasi	14
2.6 Pendidikan Islam dan Lingkungan Hidup	15
2.7 Edukasi	16
2.8 <i>Artificial Neural Network</i>	23
2.8.1 Arsitektur Jaringan	24
2.8.2 Fungsi Aktivasi	25
2.8.3 <i>Multi Layer Perceptron</i>	29
2.8.4 <i>FeedForward Neural Network</i>	29
BAB III DESAIN DAN RANCANGAN SISTEM	32
3.1 Fokus Penelitian	32
3.2 Desain <i>Game</i>	34
3.2.1 Storyboard	34
3.2.2 Desain Penyeimbangan Skenario	37
3.2.3 Desain Alur <i>Game</i>	39
3.3 Data Yang Digunakan	40
3.4 Preparasi Data	41
3.5 <i>Artificial Neural Network-Feedforward</i>	42
BAB IV HASIL DAN PEMBAHASAN	48

4.1 Implementasi	48
4.1.1 <i>UI Game</i>	48
4.1.2 <i>Gameplay</i>	49
4.1.3 Perubahan kabut.....	50
4.2 Pencarian Arsitektur Neural Network.....	61
4.2.1 Data	61
4.2.2 Arsitektur <i>Neural Network</i>	64
4.2.2.1 Arsitektur jaringan 7 – 5 – 4.....	64
4.2.2.2 Arsitektur jaringan 7 – 12 – 4.....	65
4.2.2.3 Arsitektur jaringan 7 – 20 – 4.....	66
4.2.2.4 Arsitektur jaringan 7 – 4 – 3 – 4.....	67
4.2.2.5 Arsitektur jaringan 7 – 8 – 6 – 4.....	68
4.2.2.6 Arsitektur jaringan 7 – 15 – 9 – 4.....	69
4.2.2.7 Arsitektur jaringan 7 – 20 – 10 – 4.....	69
4.2.2.8 Arsitektur jaringan 7 – 12 – 6 – 2 – 4.....	70
4.2.2.9 Arsitektur jaringan 7 – 16 – 12 – 8 – 4.....	71
4.2.2.10 Arsitektur jaringan 7 – 18 – 13 – 9 – 4.....	72
4.3 Hasil Arsitektur Jaringan.....	73
4.4 Skenario Uji	80
4.4.1 Pengujian 80:20	80
4.4.2 Pengujian 70:30	82
4.4.3 Hasil Skenario Uji.....	83
4.5 Integrasi Islam.....	84
BAB V KESIMPULAN DAN SARAN	86
5.1 Kesimpulan	86
5.2 Saran.....	86
DAFTAR PUSTAKA	

DAFTAR GAMBAR

Gambar 2. 1 Penyeimbangan Skenario	10
Gambar 2. 2 Jaringan Syaraf Manusia VS ANN	23
Gambar 2. 3 <i>Single Layer Network</i>	24
Gambar 2. 4 <i>Multilayer Network</i>	25
Gambar 2. 5 Struktur ANN	25
Gambar 2. 6 Fungsi Aktivasi <i>Linier</i>	27
Gambar 2. 7 Fungsi Aktivasi <i>ReLU</i>	27
Gambar 2. 8 Fungsi Aktivasi <i>Biner</i>	28
Gambar 2. 9 Fungsi Aktivasi <i>Bipolar</i>	29
Gambar 2. 10 Fungsi Aktivasi <i>Softemax</i>	28
Gambar 2. 11 Fungsi Aktivasi <i>Tanh</i>	29
Gambar 3. 1 Alur Penelitian.....	32
Gambar 3. 2 <i>Splash screen</i>	34
Gambar 3. 3 <i>Main Menu</i>	35
Gambar 3. 4 Tampilan kuis	35
Gambar 3. 5 Menang.....	36
Gambar 3. 6 Kalah	36
Gambar 3. 7 <i>Flowchart</i>	38
Gambar 3. 8 Blok Diagram FFNN.....	40
Gambar 3. 9 <i>Confusion Matrix</i>	43
Gambar 3. 10 Arsitektur Jaringan 7 8 6 4	44
Gambar 4. 1 <i>Main Menu</i>	48
Gambar 4. 2 Kuis	49
Gambar 4. 3 <i>Game World</i>	50
Gambar 4. 4 Kabut 1	51
Gambar 4. 5 Kabut 2	52
Gambar 4. 6 Kabut 3	53
Gambar 4. 7 Kabut 4	54
Gambar 4. 8 Kabut 5	55
Gambar 4. 9 Kabut 6	56
Gambar 4. 10 Kabut 7	57
Gambar 4. 11 Kabut 8	58
Gambar 4. 12 Kabut 9	59
Gambar 4. 13 Kabut 10	60
Gambar 4. 14 Arsitektur 7 8 6 4.....	75
Gambar 4. 15 <i>Confusion Matrix</i> 80 : 20	81
Gambar 4. 16 <i>Confusion Matrix</i> 70 : 30	82

DAFTAR TABEL

Tabel 2.1 Nilai <i>max</i> dan <i>min</i> pada atribut.	12
Tabel 2.2 Pertanyaan Kognitif.	17
Tabel 2.3 Pertanyaan Afektif.	19
Tabel 2.4 Pertanyaan Psikomotorik.	21
Tabel 3.1 Penentuan nilai <i>variabel</i> setiap <i>level</i>	40
Tabel 3.2 <i>Normalisasi</i>	41
Tabel 3.3 Arsitektur Jaringan.	42
Tabel 3.4 Nilai bobot dan bias dari <i>input</i> menuju <i>hidden layer 1</i>	44
Tabel 4.1 <i>Dataset</i>	61
Tabel 4.2 Arsitektur 7 5 4.	64
Tabel 4.3 Arsitektur 7 5 4.	64
Tabel 4.4 Arsitektur 7 12 4.	65
Tabel 4.5 Arsitektur 7 12 4.	65
Tabel 4.6 Arsitektur 7 20 4.	66
Tabel 4.7 Arsitektur 7 20 4.	66
Tabel 4.8 Arsitektur 7 4 3 4.	67
Tabel 4.9 Arsitektur 7 4 3 4.	67
Tabel 4.10 Arsitektur 7 8 6 4.	68
Tabel 4.11 Arsitektur 7 8 6 4.	68
Tabel 4.12 Arsitektur 7 15 9 4.	69
Tabel 4.13 Arsitektur 7 15 9 4.	69
Tabel 4.14 Arsitektur 7 20 10 4.	70
Tabel 4.15 Arsitektur 7 20 10 4.	70
Tabel 4.16 Arsitektur 7 12 6 2 4.	71
Tabel 4.17 Arsitektur 7 12 6 2 4.	71
Tabel 4.18 Arsitektur 7 16 12 8 4.	71
Tabel 4.19 Arsitektur 7 16 12 8 4.	72
Tabel 4.20 Arsitektur 7 18 13 9 5.	72
Tabel 4.21 Arsitektur 7 18 13 9 5.	73
Tabel 4.22 Hasil Pengujian.	73
Tabel 4.23 Nilai bobot dan bias dari <i>input</i> ke <i>hidden layer 1</i>	75
Tabel 4.24 Nilai bobot dan bias dari <i>input</i> ke <i>hidden layer 2</i>	77
Tabel 4.25 Nilai bobot dan bias dari <i>input</i> ke <i>hidden layer 3</i>	79
Tabel 4.26 Nilai <i>Precision</i> , <i>Recall</i> , <i>F1-Score</i>	81
Tabel 4.27 Arsitektur Rata <i>Precision</i> , <i>Recall</i> , <i>F1-Score</i>	82
Tabel 4.28 Arsitektur Nilai <i>Precision</i> , <i>Recall</i> , <i>F1-Score</i>	82
Tabel 4.29 Arsitektur Rata <i>Precision</i> , <i>Recall</i> , <i>F1-Score</i>	83

ABSTRAK

Khairani, Nurlaili. 2022. **Personalisasi Penyeimbangan Skenario Berbasis Profil Pemain Pada *Game* Pendidikan Islam Dan Lingkungan Hidup Menggunakan *Feedforward Neural Network* (FFNN)**. Skripsi. Jurusan Teknik Informatika, Fakultas Sains Dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Fresy Nugroho, M.T. (II) Dr. Muhammad Faisal, M.T.

Kata Kunci: *Feedforward Neural Network*, *Neural Network*, *Game* Edukasi, Pendidikan Islam, Penyeimbangan Skenario.

Untuk menentukan penyeimbangan skenario pada *game* maka diperlukan personalisasi yang menjadi ciri khas pemain. Personalisasi pada pemain mempengaruhi keseimbangan skenario pada *game* yang dimainkan oleh pemain. Keseimbangan skenario berbasis profil pemain, yang mana tantangan pada *game* diseimbangkan dengan *skill* dari pemain tersebut. Untuk dapat menentukan keseimbangan skenario yang tepat maka dapat dihitung dengan metode *Feedforward Neural Network* sehingga dapat menemukan hasil akurasi yang baik. Penyeimbangan skenario menjadi tantangan berupa tingkat *level* kabut yang terdapat pada *game*, kabut tersebut merupakan *output* dari metode *Feedforward Neural Network* yang diimplementasikan pada *game* dengan mencari arsitektur jaringan yang optimal. Pengujian arsitektur jaringan menggunakan 10 uji dengan melibatkan perbedaan *hidden layer*, fungsi aktivasi, dan *optimizer* yang berbeda. Kemudian hasil dari arsitektur yang optimal akan diuji dengan skenario 80:20 dan 70:30. Hasil yang didapatkan bahwa skenario uji 70:30 lebih optimal dengan hasil akurasi sebesar 90%, *precision* 85%, *recall* 80%, dan *f1-score* 79%. Terdapat nilai bobot dan bias dari hasil skenario uji 70:30 yang kemudian diimplementasikan pada *game* edukasi.

ABSTRACT

Khairani, Nurlaili. 2022. **Personalization of Balancing Scenarios Based on Player Profiles in Islamic and Environmental Education Games Using Feedforward Neural Networks (FFNN)** Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Counselor: (I) Dr. Fresy Nugroho, M.T. (II) Dr. Muhammad Faisal, M.T.

Keywords: Feedforward Neural Network, Artificial Neural Network, Educational Games, Islamic Education.

To determine the balance of scenarios in the game, personalization is needed which is the hallmark of the player. Player personalization affects the balance of scenarios in games played by players. Scenario balance based on player profile, where the challenge in the game is balanced with the skill of the player. To be able to determine the right scenario balance, it can be calculated using the Feedforward Neural Network method so that good accuracy results can be found. Scenario balancing is a challenge in the form of fog levels contained in the game, the fog is the output of the Feedforward Neural Network method implemented in the game by finding the optimal network architecture. Network architecture testing uses 10 tests involving different hidden layers, activation functions, and different optimizers. Then the results of the optimal architecture will be tested with 80:20 and 70:30 scenarios. The results obtained are that the 70:30 test scenario is more optimal with an accuracy of 90%, 85% precision, 80% recall, and 79% f1 score. There are weight and bias values from the results of the 70:30 test scenario which are then implemented in educational games.

مستخلص البحث

خيراني ، نورليلي. 2022. إضفاء الطابع الشخصي على السيناريوهات الموزنة بناءً على ملفات تعريف اللاعب في ألعاب الإسلاميه والبيئية باستخدام الشبكات العصبية المغذية (FFNN) التريبية. فرضية. قسم هندسة المعلوماتية ، كلية العلوم والتكنولوجيا ، جامعة الولاية الإسلامية مولانا مالك إبراهيم مالانج. مشرفان: فريسي نوغروهو الماجستير (I) برنامج محمد فيصل للماجستير (II)

.الكلمات المفتاحية : شبكة عصبية مغذية ، شبكة عصبية اصطناعية ، ألعاب تعليمية ، تربية إسلامية

لتحديد توازن السيناريوهات في اللعبة ، يلزم التخصيص وهو ما يميز اللاعب. يؤثر تخصيص اللاعب على توازن سيناريوهات اللعبة التي يلعبها اللاعبون. توازن السيناريو على أساس الملف الشخصي للاعب ، حيث يتم موازنة التحدي في اللعبة بمهارة اللاعب. لتكون Feedforward Neural Network قادرًا على تحديد توازن السيناريو الصحيح ، يمكن إجراء الحسابات باستخدام طريقة حتى يتم الحصول على نتائج دقة جيدة. تمثل موازنة السيناريو تحديًا في شكل مستويات الضباب الموجودة في اللعبة ، والضباب هو المطبقة في اللعبة من خلال إيجاد بنية الشبكة المثلى. يستخدم اختبار بنية Feedforward Neural Network ناتج طريقة الشبكة 10 اختبارات تتضمن طبقات مخفية مختلفة ووظائف التنشيط والتحسينات. ثم يتم اختبار النتائج المعيارية المثلى باستخدام سيناريوهات 80:20 و 70:30. النتائج التي تم الحصول عليها هي سيناريو اختبار أمثل 70:30 بدقة 90٪ ، دقة 85٪ ، تذكر. توجد قيم وزن وتحيز من نتائج سيناريو الاختبار 70:30 والتي يتم تنفيذها بعد ذلك في الألعاب التعليمية. fl. /درجة 79 ، 80٪

BAB I

PENDAHULUAN

1.1 Latar Belakang

Teknologi sangat membantu dalam hal mempermudah pekerjaan dan kebutuhan serta dapat memberikan hiburan beragam bagi penggunanya. Pesatnya kemajuan teknologi telah banyak memberikan pengaruh besar salah satunya dalam bidang pendidikan (Huda, 2020). Peran teknologi pada bidang pendidikan sebagai infrastruktur pembelajaran dengan materi ajar yang terdapat dalam format digital sehingga proses belajar menjadi lebih inovatif (Cholik, 2021). Kegiatan pembelajaran adalah suatu inti pada pendidikan yang dilaksanakan di sekolah-sekolah. Belajar merupakan suatu proses yang ada pada diri dan kegiatan peserta didik dalam menerima, menanggapi, dan menganalisa materi yang disampaikan oleh pengajar sehingga dapat dikuasai (Rahmayanti, 2016). Pelaksanaan pembelajaran pada zaman sekarang dengan bantuan sarana teknologi sehingga dapat meningkatkan mutu pembelajaran (Aspi, 2022).

Peningkatan mutu pembelajaran terutama pada mata pelajaran pendidikan islam yang terintegrasi dengan pendidikan lingkungan hidup memberikan kesadaran agar dapat lebih memperhatikan lingkungan, etika hidup, moralitas spiritual pada diri manusia. Islam sebagai agama yang dapat mengatur hubungan manusia dengan Khaliknya, dan juga mengatur hubungan manusia dengan makhluknya berlandaskan pada aturan secara implisit dan eksplisit terhadap kelestarian lingkungan hidupnya (Menengah et al., 2008). Allah SWT berfirman dalam surah

Al-A'raf ayat 56 mengenai tugas manusia untuk menjaga lingkungan hidup, yang berbunyi :

وَلَا تُفْسِدُوا فِي الْأَرْضِ بَعْدَ إِصْلَاحِهَا وَادْعُوهُ خَوْفًا وَطَمَعًا إِنَّ رَحْمَتَ اللَّهِ قَرِيبٌ مِّنَ الْمُحْسِنِينَ

“Dan janganlah kamu berbuat kerusakan di bumi setelah (diciptakan) dengan baik. Berdoalah kepada-Nya dengan rasa takut dan penuh harapan. Sesungguhnya rahmat Allah sangat dekat kepada orang yang berbuat kebaikan.”(QS. Al-A'raf : 56)

Melestarikan lingkungan hidup menjadi hal yang harus disadari bagi setiap manusia agar tidak muncul bencana yang nantinya dapat mengganggu ketentraman hidup umat manusia. Ikut serta dalam melestarikan lingkungan hidup menjadi hal yang wajib dilakukan sehingga para generasi muda terutama anak-anak dapat mencontoh perilaku *positif* dalam menjaga lingkungan. Namun, kurangnya minat anak-anak dalam mempelajari buku pendidikan islam dan lingkungan hidup sehingga kesadaran mereka sangat kurang akan pentingnya mencintai lingkungan. Agar dapat mengatasi hal tersebut, maka dibangunlah suatu inovasi berupa permainan *digital* atau *game* edukasi. *Game* edukasi tersebut akan memberikan pembelajaran mengenai pendidikan islam dan lingkungan hidup bagi anak-anak sehingga belajar sambil bermain jadi lebih menyenangkan.

Game edukasi merupakan suatu teknologi multimedia interaktif yang dirancang untuk mendukung pembelajaran (Widiastuti, 2012). Merancang game edukasi sebagai media pembelajaran yang dapat meningkatkan minat belajar dalam hal melestarikan lingkungan hidup. *Game* edukasi menawarkan kemajuan dengan cara bermotivasi dan bermanfaat, menawarkan kesempatan lebih besar kepada para

anak-anak dalam hasil belajar yang diharapkan (Streicher & Smeddinck, 2016). Sehingga dengan *game* edukasi anak-anak akan memperoleh pengalaman baru dalam belajar.

Pembelajaran melalui perantara *game* edukasi memiliki permasalahan yang berkaitan dengan kontrol perilaku untuk memberikan suatu pengalaman berdasarkan tingkatan skenario dari pemain. Kontrol perilaku memungkinkan adanya elemen dari komputer untuk membuat keputusan sendiri ketika adanya pilihan dalam situasi tertentu (Xia & Anand, 2016). Dalam kontrol perilaku terdapat pandangan mengenai tingkatan kesulitan *game* yang mana apabila tidak seimbang kemampuan seorang pemain dengan kesulitan *game* maka menimbulkan zona frustrasi yang mengakibatkan pemain berhenti bermain. Sebaliknya, jika tingkatan kesulitan cenderung rendah dan kemampuan pemain terlalu tinggi maka terjadi zona kebosanan dalam memainkan *game* tersebut. Penentuan penyeimbangan kesulitan tentunya berbeda-beda tiap pemain hal tersebut dipengaruhi oleh profil pemain. Profile pemain menjadi landasan untuk menentukan tingkatan awal dalam memulai *game* yang tingkatan kesulitannya dipengaruhi oleh profile pemain itu sendiri.

Profil pemain merupakan karakter yang terdapat dan melekat pada diri pemain itu sendiri. Karakter tersebut dapat berupa tingkatan kemampuan ketika pemain memainkan *game*. Penyeimbangan kesulitan pada *game* sangat berpengaruh pada profile pemain dikarenakan hal tersebut menjadi dasar seorang dikatakan mampu untuk memecahkan masalah pada permainan. Penerapan penyeimbangan kesulitan dalam konteks *game* edukasi dapat mempelajari

kemampuan linguistik dan menjaga peserta didik tenggelam dalam permainan yang secara dinamis tingkatan kesulitan disesuaikan dan dapat mencapai target dari pembelajaran (Streicher & Smeddinck, 2016).

Metode tingkatan kesulitan pada *game* memerlukan suatu *algoritma* pembelajaran mesin sehingga nantinya sistem dapat menentukan keputusan secara otomatis berdasarkan dari masukan dari pemain. Salah satu algoritma pembelajaran mesin yaitu *Artificial Neural Network* atau jaringan syaraf tiruan. *Artificial Neural Network* adalah sistem cerdas yang berguna untuk mengolah informasi yang perkembangannya diambil dari generasi model matematika dan terinspirasi dari prinsip kerja jaringan syaraf manusia (Kukiela, n.d.). Terdapat jenis-jenis *Artificial Neural Network* salah satunya adalah *Feedforward Neural Network* merupakan model *Neural Network* yang banyak digunakan untuk berbagai bidang. Pada penelitian sebelumnya, *Feedforward Neural Network* digunakan untuk rekomendasi wisata disuatu kota. Pada penelitian tersebut, masukan data berupa preferensi pengguna yang berjumlah 10 preferensi yang bertujuan untuk mendapatkan rekomendasi 5 destinasi wisata (Wardani, 2022).

Berdasarkan uraian diatas, maka penelitian ini mengajukan penerapan *Feedforward Neural Network* untuk penyeimbangan skenario dalam *game* edukasi yang kontennya tentang Pendidikan Islam dan Lingkungan Hidup. Metode *Feedforward Neural Network* akan memberikan suatu keputusan berupa tingkatan *level* awal yang akan dimainkan oleh pemain, Sehingga penulis mengangkat judul **“Personalisasi Penyeimbangan Skenario Berbasis Profil Pemain Pada *Game* Pendidikan Islam Dan Lingkungan Hidup Menggunakan *Feedforward***

Neural Network (FFNN)”.

1.2 Identifikasi Masalah

Seberapa besar tingkat akurasi, *precision*, *recall*, *f1-score* metode *Feedforward Neural Network* dalam mengatur keseimbangan skenario berdasarkan profil pemain pada *game* Pendidikan Islam dan Lingkungan Hidup.

1.3 Tujuan Penelitian

Penelitian ini bertujuan untuk menguji tingkat akurasi, *precision*, *recall*, *f1-score* metode *Feedforward Neural Network* dalam mengatur keseimbangan skenario berdasarkan profil pemain pada *game* Pendidikan Islam dan Lingkungan Hidup.

1.4 Batasan Masalah

1. *Game* edukasi Pendidikan Islam dan Lingkungan Hidup menggunakan materi edukasi yang diperoleh dari buku Pendidikan Islam dan Lingkungan Hidup (Menengah et al., 2008)
2. *Game* menggunakan konten edukasi berdasarkan konsep Taksonomi Bloom
3. Penelitian difokuskan pada pembuatan *game* yang ditujukan kepada siswa sekolah dasar dan sederajat kelas 4.
4. *Game* bersifat *single player*.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini diharapkan dapat digunakan untuk alternatif dalam bermain *game* sehingga menambah pengetahuan tentang pendidikan islam dan kesadaran akan pentingnya melestarikan lingkungan hidup.

BAB II

STUDI PUSTAKA

2.1 Penelitian Terkait

Penelitian terkait bertujuan untuk acuan dan pembandingan dari penelitian ini. Beberapa peneliti telah mencoba membantu menyelesaikan permasalahan dengan metode dan data yang digunakan.

Pada penelitian yang dilakukan oleh Awwalia yang berjudul “*Game Pengenalan Tata Surya Menggunakan Metode Neural Network Backpropagation Dalam Menentukan Level*” yang mana peneliti menggunakan metode *Neural Network Backpropagation* untuk mengetahui nilai akurasi dari algoritma yang terdapat pada *game* dan hasil yang didapatkan ialah 76.47% algoritma berjalan sesuai dengan skor dan waktu yang telah ditentukan dalam hal kenaikan level (Hayati, 2018).

Pada penelitian yang dilakukan oleh M. Arif yang berjudul “*Game Sharraf Solitaire Menggunakan Neural Network Backpropagation Untuk Penentuan Level Pada Game Pembelajaran Tashrif Istilah Dalam Ilmu Sharraf*” yang mana dalam menjalankan proses perhitungan algoritma *Neural Network Backpropagation* membutuhkan waktu sekitar 3-5 detik untuk mendapatkan sebuah keputusan. Serta algoritma ini dapat melakukan *learning* terhadap suatu *inputan* dan menghasilkan *output* sebuah keputusan yang cocok diterapkan dalam *game* untuk meningkatkan level pemahaman agar *game* tidak bosan untuk dimainkan (Mukhlisin, 2014).

Pada penelitian yang dilakukan oleh Samuel, yang berjudul “*Pembuatan Survival Action Game Dengan Non-Player Character Berbasis Neural Network*”

dalam sistem *backpropagation* menggunakan metode *non-real time training* tidak akan menampilkan *output* yang tidak sesuai dengan *data training* karena proses *training* yang dilakukan berjalan dengan baik sehingga *uni-unit* tersebut dapat mengirimkan *signal input* dan menerima *signal output*. Pada penelitian tersebut terdapat 1 *hidden layer* dan 15 *neuron* dalam *hidden layer* (Yootje et al., 2016) .

Pada penelitian yang dilakukan oleh M.Syukron yang berjudul “*Penerapan Algoritma Backpropagation Untuk Menentukan Level Bonus Dan Score Pada Game Edukasi Nahwu Menggunakan Kartu Berbasis Android*” dengan metode *backpropagation* berhasil diimplementasikan pada *game*, yang mana pengujian jaringan syaraf tiruan terhadap 3 model arsitektur bahwa 3 lapisan tersembunyi dengan tingkat akurasi mencapai 91.71% (Syarif, 2016) .

Pada penelitian ini menggunakan metode *neural network* yang tahapannya mengimplementasikan model *feedforward neural network* dengan fungsi aktivasi sehingga dapat menentukan perubahan *level game* sebagai keputusan bagi pemain. Model *feedforward neural network* ini menggunakan beberapa arsitektur percobaan untuk diuji sampai menemukan arsitektur yang cocok untuk dilakukan pengimplementasian pada sistem. Kemudian dari beberapa arsitektur maka dapat melihat perbedaan untuk setiap arsitektur. Perubahan tersebut berupa tingkatan *game* yang menyesuaikan dengan pemain. Perubahan *level game* akan terus bertambah hingga waktu permainan berhenti.

2.2 Personalisasi

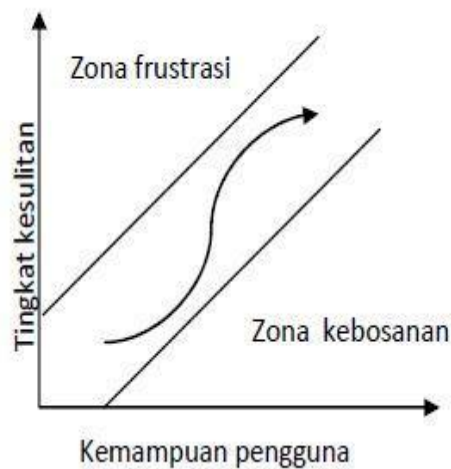
Personalisasi menurut KBBI yaitu proses, cara atau perbuatan yang mengubah atau memodifikasi sesuatu agar menunjukkan ciri personal tertentu.

Dalam *game* personalisasi memberikan pandangan kepada pemain bahwa *game* yang dimainkan memiliki keunikan yang berbeda-beda berupa segi audio, tampilan, tingkat kesulitan dan jenis *game*. Personalisasi pada *game* sebagai suatu proses untuk dapat melakukan penyeimbangan kesulitan pada *level game* tujuan dibentuknya untuk membandingkan beberapa tingkatan kesulitan pada *game* sehingga nantinya dalam *level game* banyak variasi yang harus dilalui oleh pemain untuk dapat menyelesaikan *game*. Personalisasi *game* sangat penting dalam proses merancang sebuah *game* karena dapat memodifikasi unsur-unsur yang terdapat dalam *game*.

2.3 Penyeimbangan Skenario

Penyeimbangan skenario sangat bervariasi dalam dunia *game* misalnya menyesuaikan variabel dalam permainan untuk memberikan efek yang berbeda-beda pada musuh yang akan dihadapi. Dalam kasus konten aset juga dapat digunakan seperti melewati platform dengan ukuran tertentu serta jarak, kecepatan, dan tinggi lompatan yang dapat mempengaruhi penyeimbangan kesulitan yang terdapat pada *game*.

Penyeimbangan skenario merupakan suatu bentuk keseimbangan kemampuan pemain dengan kesulitan *game* sehingga tidak menimbulkan frustrasi pada pemain. Penyeimbangan skenario dapat memberikan tantangan kepada pemain sesuai dengan tingkat dari pemahaman pemain. Jika *game* tidak memiliki tantangan maka *game* terasa bosan saat dimainkan karena pemain mudah menebak alur *game* dan hasilnya *game* tersebut jarang dimainkan oleh pemain.



Gambar 2.1 Penyeimbangan Skenario

Pada gambar menjelaskan tentang tingkat kemampuan pemain yang terlalu tinggi namun tantangan kesulitan dalam *game* rendah maka pemain akan sangat bosan sehingga *game* tersebut tidak asik lagi bagi mereka sedangkan jika kemampuan pemain rendah namun tantangan yang diberikan sangat tinggi sehingga mengakibatkan pemain akan stress dalam memainkan *game* tersebut. Maka dengan adanya penyeimbangan skenario *game* akan dikemas dengan tingkat kesulitan yang bergantung pada kemampuan dari pemain dan nantinya antara kemampuan pemain dan tingkat kesulitan *game* dapat *balance*.

Pada *game* suatu pengalaman memegang hal penting yang dapat memberikan pemahaman terhadap materi yang akan disampaikan. Untuk membentuk pemahaman tersebut diperlukan skenario *game* yang dapat menyajikan pengalaman berupa suatu *rewards* yang berfungsi menjaga motivasi dan pengalaman pemain ketika bermain.

Skenario adalah suatu alur cerita *game* tiap level dan menuju ke level selanjutnya. Setiap skenario dirancang berdasarkan karakteristik dari konten *game*

sehingga jalan cerita yang digambarkan memiliki perjalanan yang berbeda-beda dan adanya skenario berpotensi untuk meningkatkan kompleksitas dalam menyelesaikan *game* (Arif et al., 2021). Adanya skenario memudahkan supaya *game* dapat diarahkan agar tak melebar. Skenario sebagai alur penyampaian materi yang ada pada *game* kepada pemain agar mudah dimengerti dan pahami. Skenario *game* merupakan langkah pertama dalam membangun *game*. *Game* dapat dirancang secara struktur akibat adanya skenario tersebut. Jalan cerita yang terdapat pada *game* memberikan pembelajaran yang beragam bagi pemain sehingga apa yang dikisahkan atau hal yang disampaikan dapat tercapai dan dimengerti. Pada *game* terdapat beberapa skenario yang bergantung pada jenis permainannya salah satunya yaitu *single game* petualangan yang lebih menitikberatkan pada skenario yang menarik dan panjang sehingga pemain akan merasa penasaran bagaimana memainkan permainan tersebut hingga selesai (Caesarini, 2014).

1. Penyeimbangan Kesulitan

Pada *game* terdapat beberapa elemen seperti aksi dan tantangan dalam menyelesaikan aktivitas *game*. Tantangan dalam *game* dirancang dengan beberapa tingkat kesulitan seperti mudah, sedang, dan sulit (Perez et al., 2016). Namun, para pemain memiliki tingkat keterampilan yang berbeda antara satu sama lain sehingga untuk menyesuaikan mekanisme permainan maka digunakan penyeimbangan kesulitan untuk memberikan tingkat kesulitan dari pemain. Tingkat kesulitan dari permainan dapat memenuhi tingkat kemampuan dari pemain yang mengambil resiko atau menghindarinya sehingga corak permainan tiap pemain berbeda-beda.

Perancangan menggunakan profil pemain dalam merancang tingkat kesulitan

serta emosional yang tepat dalam permainan. Adanya penyeimbangan kesulitan dapat mengimplementasikan cara memantau tingkat dari kesuksesan pemain dan memberikan pengalaman menantang dalam mempertahankan minat dari pemain. Parameter yang terdapat pada penyeimbangan kesulitan meliputi waktu yang dibutuhkan pemain dalam menyelesaikan game, jumlah nyawa karakter, poin yang diperoleh, dan tipe tantangan yang harus diselesaikan.

Tabel 2.1 Nilai max dan min pada atribut

	Variabel	Nilai
Maximum	Score	100
	Time	300
	Health	100
	Jumlah benda	20
	Nilai benda	10
	Tipe musuh	5
	Jumlah Musuh	10
Minimum	Score	0
	Time	10
	Health	10
	Jumlah benda	5
	Nilai benda	5
	Tipe musuh	1
	Jumlah Musuh	2

Menghitung nilai keterampilan dari atribut yang mana menghitung *skill* pemain dan musuh berdasarkan nilai dari atribut dalam permainan ($fval$), nilai maksimum ($Fmax$) dan minimum ($Fmin$) dalam setiap permainan menggunakan rumus (1). (Avi Shena et al., 2019)'

$$efF_i = \left\{ \frac{Fval_i - Fmin_i}{Fmax_i - Fmin_i} \frac{Fmax_i - Fval_i}{Fmax_i - Fmin_i} \right\} \quad (2.1)$$

Menghitung nilai total *skill* pemain dan musuh dengan hasil berupa nilai

keterampilan atribut serta bobot dari tiap atribut yang nantinya dapat mendefinisikan keterampilan dari pemain. Menggunakan rumus (2).

$$ef = \frac{\sum_{i=1}^n (efF_i * weight_i)}{\sum_{i=1}^n weight_i} \quad (2.2)$$

Membandingkan keterampilan musuh dan pemain untuk dapat memutuskan penyesuaian tantangan level menggunakan rumus (3). efp adalah nilai dari *skill* pemain dan efo adalah nilai *skill* musuh nantinya dihitung dari rumus yang sebelumnya yaitu rumus ke 2.

$$diffef = |efp - efo| \quad (2.3)$$

Menentukan kebutuhan penyesuaian kesulitan menggunakan rumus (4) dengan $diffef$ yaitu perbedaan nilai pemain dan musuh. Terdapat $plim$ atau *skill* musuh apabila pemain dapat nilai lebih tinggi dari *skill* musuh maka tingkat kesulitan pada *game* akan disesuaikan.

$$diffef > plim * efp \quad (2.4)$$

2.4 Profil Pemain

Profil pemain merupakan karakter yang terdapat dan melekat pada diri pemain itu sendiri yang tentunya berbeda-beda setiap pemainnya. Profil tersendiri pengacu pada pandangan yang dilihat oleh pemain bentuknya dapat berupa keahlian yang terdapat pada pemain itu sendiri. Profile pemain menceritakan gambaran dari tiap pemain yang mana nantinya ketika bermain *game*, tiap pemain akan memulai level permainan yang berbeda hal tersebut, bergantung pada dari profil mereka sendiri.

Pada profil pemain ini mendeskripsikan berupa kemampuan dari pemain

yang mencakup *score*, *time*, *healty*, nilai benda, jumlah benda, tipe musuh, jumlah musuh. Kemampuan tersebut menjadi masukan yang akan digunakan untuk menghitung tingkat dari penyeimbangan skenario yang akan dimainkan oleh pemain.

2.5 *Game* Edukasi

Game merupakan suatu aktivitas bermain yang dilakukan dengan aturan yang sudah ditetapkan sehingga pemain memiliki tujuan untuk mendapatkan kemenangan, biasanya *game* dimainkan untuk *refreshing* atau mengisi waktu luang. Dalam sebuah *game* terdapat suatu konflik buatan yang diatur berdasarkan rule tertentu sehingga nantinya terdapat *goal* yang harus dicapai oleh pemain. *Game* menjadi salah satu cara untuk menghilangkan tekanan dengan melakukan kegiatan interaksi dengan *game* menggunakan kecerdasan berpikir melewati konflik yang direkayasa untuk menimbulkan keseruan dalam bermain *game*.

Edukasi merupakan suatu proses kegiatan belajar mengajar untuk meningkatkan kecerdasan pola pikir serta mengembangkan potensi yang terdapat pada peserta didik melalui berbagai macam cara sehingga proses pembelajaran tidak monoton (Afrianto, 2019). Menurut Undang-undang Nomor 20 Tahun 2003, mengenai pendidikan merupakan usaha sadar dan terencana untuk mewujudkan suasana belajar dan proses pembelajaran agar peserta didik secara aktif mengembangkan potensi dirinya untuk memiliki kekuatan spiritual keagamaan, pengendalian diri, kepribadian, kecerdasan, akhlak mulia, serta keterampilan yang diperlukan dirinya, masyarakat, bangsa dan negara.

Game edukasi merupakan kata yang berasal dari bahasa inggris, yaitu *game*

yang berarti permainan dan *education* yang berarti pendidikan. Maka penggabungan dari kata ini bertujuan untuk memberikan pengetahuan melalui permainan yang bersifat interaktif. Pengemasan nilai edukasi yang terdapat dalam *game* menjadikan *game* bernilai lebih menarik dan juga meningkatkan minat peserta didik dikarenakan dapat belajar sambil bermain. menurut Mubarak (2018) *game* edukasi merupakan suatu permainan yang dijadikan sebagai media pembelajaran untuk menunjang proses pembelajaran, yang didalamnya terdapat unsur mendidik atau nilai-nilai pendidikan. Menurut pendapat lain yaitu Chagas bahwa *game* edukasi, *game* pembelajaran, *game* serius dirancang untuk meningkatkan keterampilan kognitif dan intelektual orang yang bermain sehingga dapat memperoleh nilai edukasi *game* (Amalia et al., 2021).

Game sebagai media pembelajaran bermanfaat dalam mengembangkan diri peserta didik. *Game* edukasi ini dibangun untuk belajar sambil bermain dengan adanya konten edukasi, prinsip pembelajaran, dan *game* komputer. *Game* jenis ini digunakan untuk memadukan dua sisi yaitu belajar dan bermain, serta menarik perhatian peserta didik supaya lebih semangat, giat, dan rajin dalam belajar.

2.6 Pendidikan Islam dan Lingkungan Hidup

Hubungan agama dan lingkungan hidup sudah lama menjadi bincangan para ilmuwan. Adanya penafsiran mengenai lingkungan dalam AL-Qur'an sebagai aturan untuk manusia agar menjaga lingkungan dengan baik. Pendidikan islam menjadi membentuk kepribadian manusia agar menjadi insan yang baik dan bijaksana. Pendidikan islam merupakan bentuk usaha yang dilakukan secara sadar, dengan membimbing, mengasuh peserta didik agar dapat meyakini, menghayati dan

mengamalkan ajaran islam sehingga dapat mengubah sikap serta tingkah laku dalam usaha mendewasakan melalui upaya pengajaran.

Pengetahuan akan pendidikan islam memuat nilai islam yang diturunkan oleh Allah swt. Segenap upaya dari pengajar untuk mendidik manusia agar terbentuknya manusia yang sempurna secara jasmaniyah dan rohaniyah yang berlandaskan ajaran islam. Pendidikan islam banyak mencakup bidang salah satunya yaitu, bidang lingkungan hidup. Lingkungan hidup merupakan semua benda, daya dan suatu kondisi yang terdapat dalam satu ruang, tempat dimana kita berada serta mempengaruhi hidup kita (Dan et al., 1982). Lingkungan jangan sampai dirusak serta manusia harus bertanggung jawab atas lingkungan yang ditempati. Sebagai bagian dari lingkungan manusia sangat bergantung kepada lingkungan, manusia dan lingkungan tidak dapat dipisahkan. Oleh karena itu, manusia wajib memelihara lingkungan yang ada untuk kelanjutan hidupnya dan kehidupan generasi sesudahnya.

2.7 Edukasi

Pada *game* yang akan dibuat terdapat konten edukasi yang akan digunakan yang mengarah pada Taksonomi Bloom. Taksonomi bloom terdiri dari pengetahuan, pemahaman, aplikasi, analisis, sintesis, dan evaluasi. Terdapat 3 aspek pembelajaran yang terdapat dalam Taksonomi bloom yaitu aspek kognitif, aspek afektif, dan aspek psikomotorik pemain.

a. Kognitif

Aspek kognitif adalah aktivitas mental yang mengarah pada proses pembelajaran berupa kemampuan melihat, memperhatikan, menilai, dan

mempertimbangkan suatu kejadian sehingga nantinya mendapatkan suatu pengetahuan.

Tabel 2.2 Pertanyaan Kognitif

<i>Level</i>	<i>Output pertanyaan</i>	Kognitif
1	- Apa yang kamu ketahui tentang melestarikan alam?	K1-1
	- Apa yang kamu ketahui tentang kebersihan lingkungan?	
	- Apa yang kamu ketahui tentang reboisasi?	
	- Apa yang kamu ketahui tentang pencemaran lingkungan?	
	- Apa yang kamu ketahui tentang kekeringan?	
2	- Sebutkan 3 cara melestarikan alam?	K1-2
	- Sebutkan 3 cara menjaga kebersihan lingkungan?	
	- Sebutkan 3 cara melakukan reboisasi?	
	- Sebutkan 3 cara menghindari pencemaran lingkungan?	
	- Sebutkan 3 cara menghindari kekeringan?	
3	- Sebutkan 3 dampak yang terjadi apabila alam rusak?	K1-3
	- Sebutkan 3 dampak yang terjadi lingkungan rusak?	
	- Sebutkan 3 dampak yang terjadi apabila hutan gundul?	
	- Sebutkan 3 dampak yang terjadi apabila terjadi pencemaran lingkungan?	
	- Sebutkan 3 dampak yang terjadi apabila terjadi kekeringan?	
4	- Sebutkan penyebab rusaknya alam?	K1-4
	- Sebutkan penyebab rusaknya lingkungan?	
	- Sebutkan penyebab hutan gundul?	
	- Sebutkan penyebab pencemaran lingkungan?	
	- Sebutkan penyebab kekeringan?	
5	- Apa tujuan dari melestarikan alam?	K2-1
	- Apa tujuan dari menjaga lingkungan?	
	- Apa tujuan dari melakukan reboisasi?	
	- Apa tujuan dari membuang sampah pada tempatnya?	
	- Apa tujuan dari merawat hutan?	
6	- Apa yang kamu ketahui tentang ciri-ciri alam rusak?	K2-2
	- Apa yang kamu ketahui tentang ciri-ciri lingkungan sehat?	

	<ul style="list-style-type: none"> - Apa yang kamu ketahui tentang ciri-ciri dari reboisasi? 	
	<ul style="list-style-type: none"> - Apa yang kamu ketahui tentang ciri-ciri pencemaran lingkungan? 	
	<ul style="list-style-type: none"> - Apa yang kamu ketahui tentang ciri-ciri kekeringan? 	
7	<ul style="list-style-type: none"> - Sebutkan cara penanganan alam yang rusak? - Sebutkan cara penanganan lingkungan yang rusak? - Sebutkan cara penanganan hutan yang gundul? - Sebutkan cara penanganan lingkungan yang tercemar? - Sebutkan cara penanganan kekeringan? 	K2-3
8	<ul style="list-style-type: none"> - Apa yang perlu ditingkatkan dari penanganan alam yang rusak? - Apa yang perlu ditingkatkan dari penanganan lingkungan yang rusak? - Apa yang perlu ditingkatkan dari penanganan hutan gundul? - Apa yang perlu ditingkatkan dari penanganan lingkungan yang tercemar? - Apa yang perlu ditingkatkan dari penanganan kekeringan? 	K3-1
9	<ul style="list-style-type: none"> - Sebutkan Lembaga yang menangani alam rusak? - Sebutkan Lembaga yang menangani lingkungan rusak? - Sebutkan Lembaga yang menangani hutan gundul? - Sebutkan Lembaga yang menangani pencemaran lingkungan? - Sebutkan Lembaga yang menangani kekeringan? 	K3-2
10	<ul style="list-style-type: none"> - Sebutkan akibat dari terlambatnya penanganan alam rusak? - Sebutkan akibat dari terlambatnya penanganan lingkungan yang rusak? - Sebutkan akibat dari terlambatnya penanganan hutan gundul? - Sebutkan akibat dari terlambatnya penanganan pencemaran lingkungan? - Sebutkan akibat dari terlambatnya penanganan kekeringan? 	K3-3

b. Afektif

Aspek Afektif adalah aspek yang berkaitan dengan sikap mental berupa watak, karakter, emosi serta perilaku. Afektif mencakup perasaan serta emosional yang dapat berupa adanya perasaan suka terhadap sesuatu.

Tabel 2.3 Pertanyaan Afektif

<i>Level</i>	<i>Output</i> Pertanyaan	Afektif
1	- Bagaimana sikapmu jika melihat alam yang rusak?	A1-1
	- Bagaimana sikapmu jika melihat lingkungan yang rusak?	
	- Bagaimana sikapmu jika melihat reboisasi atau penghijauan?	
	- Bagaimana sikapmu jika melihat pencemaran lingkungan?	
	- Bagaimana sikapmu jika kekeringan?	
2	- Bagaimana sikapmu jika di lingkunganmu terjadi alam yang rusak?	A1-2
	- Bagaimana sikapmu jika tinggal di lingkungan yang rusak?	
	- Bagaimana sikapmu jika tinggal di wilayah hutan gundul?	
	- Bagaimana sikapmu jika tinggal di lingkungan yang tercemar?	
	- Bagaimana sikapmu jika tinggal di wilayah yang kekeringan?	
3	- Bagaimana sikapmu ikut serta dalam melestarikan alam?	A1-3
	- Bagaimana sikapmu ikut serta dalam menjaga kebersihan lingkungan?	
	- Bagaimana sikapmu ikut serta dalam melakukan reboisasi?	
	- Bagaimana sikapmu ikut serta dalam menjaga lingkungan?	
	- Bagaimana sikapmu ikut serta dalam mencegah terjadinya kekeringan?	
4	- Bagaimana sikapmu jika melihat masyarakat melakukan pelestarian alam?	A1-4
	- Bagaimana sikapmu jika melihat masyarakat menjaga kebersihan lingkungan?	
	- Bagaimana sikapmu jika melihat masyarakat melakukan reboisasi?	
	- Bagaimana sikapmu jika melihat masyarakat menjaga adanya pencemaran lingkungan?	
	- Bagaimana sikapmu jika melihat masyarakat mengatasi terjadinya kekeringan?	
5	- Bagaimana pendapatmu jika mendengar informasi mengenai pelestarian alam?	A2-1
	- Bagaimana pendapatmu jika mendengar informasi mengenai kebersihan lingkungan?	
	- Bagaimana pendapatmu jika mendengar informasi mengenai reboisasi?	
	- Bagaimana pendapatmu jika mendengar informasi mengenai pencemaran lingkungan?	
	- Bagaimana pendapatmu jika mendengar informasi mengenai kekeringan?	
6	- Bagaimana pendapatmu jika terdapat orang yang mengabaikan informasi mengenai pelestarian alam?	A2-2

	<ul style="list-style-type: none"> - Bagaimana pendapatmu jika terdapat orang yang mengabaikan informasi mengenai kebersihan lingkungan 	
	<ul style="list-style-type: none"> - Bagaimana pendapatmu jika terdapat orang yang mengabaikan informasi mengenai reboisasi? 	
	<ul style="list-style-type: none"> - Bagaimana pendapatmu jika terdapat orang yang mengabaikan informasi mengenai pencemaran lingkungan? 	
	<ul style="list-style-type: none"> - Bagaimana pendapatmu jika terdapat orang yang mengabaikan informasi mengenai kekeringan? 	
7	<ul style="list-style-type: none"> - Bagaimana pendapatmu mengenai aturan pelestarian alam? 	A2-3
	<ul style="list-style-type: none"> - Bagaimana pendapatmu mengenai aturan kebersihan lingkungan? 	
	<ul style="list-style-type: none"> - Bagaimana pendapatmu mengenai aturan reboisasi? 	
	<ul style="list-style-type: none"> - Bagaimana pendapatmu mengenai aturan pelestarian lingkungan 	
	<ul style="list-style-type: none"> - Bagaimana pendapatmu mengenai aturan pencegahan kekeringan? 	
8	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap pelestarian alam? 	A3-1
	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap kebersihan lingkungan? 	
	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap reboisasi? 	
	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap pencemaran lingkungan? 	
	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap kekeringan 	
9	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap pencegahan kerusakan alam? 	A3-2
	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap pencegahan pencemaran lingkungan? 	
	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap pencegahan hutan gundul? 	
	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap pencegahan banjir? 	
	<ul style="list-style-type: none"> - Bagaimana saranmu terhadap pencegahan kekeringan? 	
10	<ul style="list-style-type: none"> - Apa yang kamu rasakan jika melihat orang tinggal di alam yang rusak? 	A3-3
	<ul style="list-style-type: none"> - Apa yang kamu rasakan jika melihat orang tinggal di lingkungan yang tercemar? 	
	<ul style="list-style-type: none"> - Apa yang kamu rasakan jika melihat orang tinggal di wilayah hutan yang gundul? 	
	<ul style="list-style-type: none"> - Apa yang kamu rasakan jika melihat orang tinggal di wilayah yang terdampak banjir? 	
	<ul style="list-style-type: none"> - Apa yang kamu rasakan jika melihat orang tinggal di wilayah yang kekeringan? 	

c. Psikomotorik

Aspek psikomotorik adalah aspek yang berkaitan dengan keterampilan terdiri dari kesiapan, peniruan, membiasakan, menyesuaikan, serta menciptakan.

Dengan adanya pemahaman maka dapat langsung diterapkan dalam kehidupan sehari-hari.

Tabel 2.4 Pertanyaan Psikomotorik

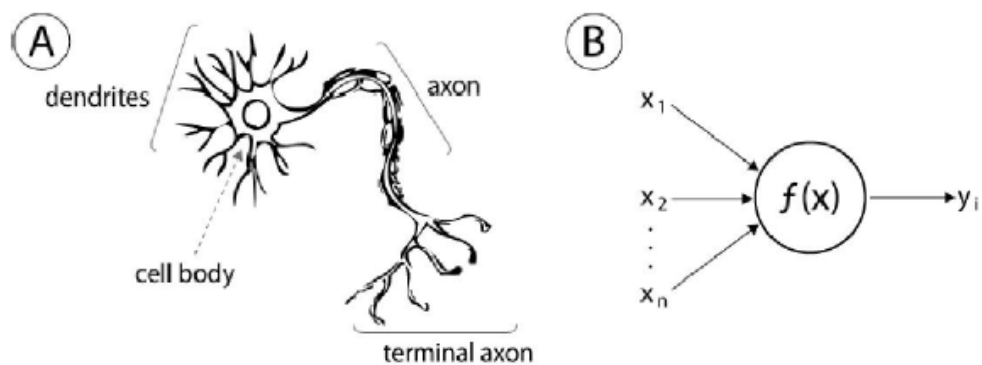
<i>Level</i>	<i>Output</i> Pertanyaan	Psikomotorik
1	- Apa yang kamu lakukan jika mendengar kabar akan terjadi bencana alam?	P1-1
	- Apa yang kamu lakukan jika mendengar kabar akan terjadi pencemaran lingkungan?	
	- Apa yang kamu lakukan jika mendengar kabar akan terjadi bencana tanah longsor?	
	- Apa yang kamu lakukan jika mendengar kabar akan bencana terjadi banjir?	
	- Apa yang kamu lakukan jika mendengar kabar akan terjadi kekeringan?	
2	- Apa yang kamu lakukan jika terjadi bencana alam di daerahmu?	P1-2
	- Apa yang kamu lakukan jika terjadi pencemaran lingkungan di daerahmu?	
	- Apa yang kamu lakukan jika terjadi tanah longsor di daerahmu?	
	- Apa yang kamu lakukan jika terjadi banjir di daerahmu?	
	- Apa yang kamu lakukan jika terjadi kekeringan di daerahmu?	
3	- Apa yang kamu lakukan setelah terjadi bencana alam?	P1-3
	- Apa yang kamu lakukan setelah terjadi bencana pencemaran lingkungan?	
	- Apa yang kamu lakukan setelah terjadi bencana tanah longsor?	
	- Apa yang kamu lakukan setelah terjadi bencana banjir?	
	- Apa yang kamu lakukan setelah terjadi bencana kekeringan?	
4	- Jenis bantuan yang akan kamu berikan kepada korban bencana alam?	P1-4
	- Jenis bantuan yang akan kamu berikan kepada korban gunung meletus?	
	- Jenis bantuan yang akan kamu berikan kepada korban tanah longsor?	
	- Jenis bantuan yang akan kamu berikan kepada korban banjir?	
	- Jenis bantuan yang akan kamu berikan kepada korban bencana kekeringan?	
5	- Apa usaha yang kamu lakukan untuk mencegah bencana alam?	P2-1

	<ul style="list-style-type: none"> - Apa usaha yang kamu lakukan untuk mencegah pencemaran lingkungan? 	
	<ul style="list-style-type: none"> - Apa usaha yang kamu lakukan untuk mencegah tanah longsor? 	
	<ul style="list-style-type: none"> - Apa usaha yang kamu lakukan untuk mencegah banjir? 	
	<ul style="list-style-type: none"> - Apa usaha yang kamu lakukan untuk mencegah kekeringan? 	
6	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika temanmu mengajakmu membantu korban bencana alam? 	P2-2
	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika temanmu mengajakmu membantu korban pencemaran lingkungan? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika temanmu mengajakmu membantu korban tanah longsor? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika temanmu mengajakmu membantu korban banjir? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika temanmu mengajakmu membantu korban kekeringan? 	
7	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat kotak sumbangan bencana alam? 	P2-3
	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat kotak sumbangan pencemaran lingkungan? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat kotak sumbangan bencana tanah longsor? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat kotak sumbangan bencana banjir? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat kotak sumbangan bencana kekeringan? 	
8	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat korban bencana alam yang sakit? 	P3-1
	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat korban pencemaran lingkungan yang sakit? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat korban tanah longsor yang sakit? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat korban banjir yang sakit? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan ketika melihat korban kekeringan yang sakit? 	
9	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika melihat korban bencana alam yang meninggal? 	P3-2
	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika melihat pencemaran lingkungan yang meninggal? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika melihat korban bencana tanah longsor yang kelaparan? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika melihat korban bencana banjir yang kedinginan? 	
	<ul style="list-style-type: none"> - Apa yang kamu lakukan jika melihat korban bencana kekeringan yang kepanasan? 	

10	- Apa yang kamu rasakan jika keluargamu terdampak bencana alam?	P3-3
	- Apa yang kamu rasakan jika keluargamu terdampak lingkungan yang rusak?	
	- Apa yang kamu rasakan jika keluargamu terdampak tanah longsor?	
	- Apa yang kamu rasakan jika keluargamu terdampak banjir?	
	- Apa yang kamu rasakan jika keluargamu terdampak kekeringan?	

2.8 Artificial Neural Network

Artificial Neural Network (ANN) merupakan sistem cerdas yang berguna untuk mengolah informasi yang perkembangannya di ambil dari generalisasi model matematika. Prinsip kerja dari *Artificial Neural Network* terinspirasi dari prinsip kerja sistem jaringan saraf manusia. Ilustrasi kemiripan arsitektur ANN dengan sistem jaringan syaraf manusia.



Gambar 2.2 Jaringan syaraf manusia VS ANN

Pada gambar A adalah struktur susunan sel *neuron*, terdiri dari dendrit berfungsi untuk menerima rangsangan, badan sel berfungsi mengakumulasi rangsangan serta memproses informasi dari dendrit menuju akson, akson kemudian meneruskan rangsangan yang telah diproses badan sel menuju *neuron* lain. Sedangkan pada gambar b adalah struktur ANN terdiri dari 3 bagian yaitu, $input(x)$

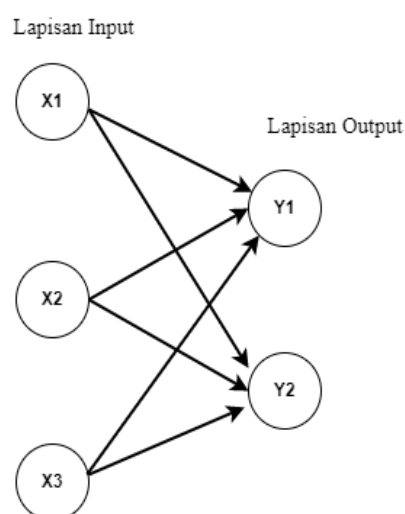
bertugas untuk menerima informasi menggunakan bobot yang telah ditentukan, $hidden(f(x))$ bertugas untuk mengumpulkan dan mengakulasikan bobot, serta hasil penjumlahan dibandingkan dengan $threshold$ yang ditentukan sebagai nilai aktifasi jika memenuhi syarat maka akan menuju $output(y)$ (Ryandhi, 2017).

2.8.1 Arsitektur Jaringan

Arsitektur ANN merupakan suatu gambaran susunan komponen layer berupa $input$, $hidden$, dan $output$ yang terhubung dengan bobot, fungsi aktivasi dan $learning function$. Adanya arsitektur jaringan dapat menjelaskan arah dari perjalanan data dalam jaringan. Terdapat 2 struktur ANN berdasarkan dari jumlah $hidden layer$ yang digunakan :

1. *Single Layer Network*

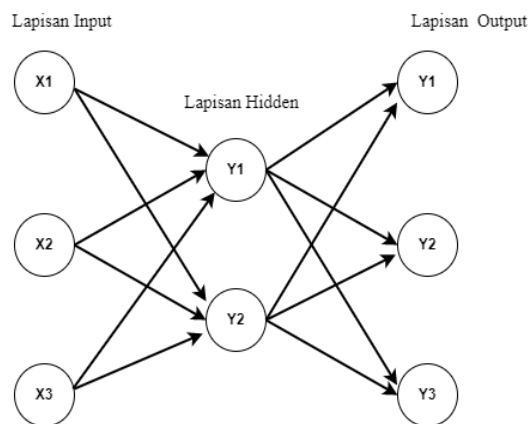
Terdiri dari 1 $layer input$ dan 1 $layer output$ dari bobot yang terhubung. Pada jaringan $single$ ini menerima $input$ kemudian langsung memprosesnya menjadi sebuah $output$ tanpa adanya $hidden layer$.



Gambar 2.3 *Single Layer Network*

2. Multilayer Network

Memiliki beberapa *layer* namun, secara umum terdapat 3 jenis *layer* yaitu *input*, *output*, dan *hidden*. Pada jaringan ini permasalahan yang diselesaikan lebih kompleks daripada jaringan *single* dan proses pengolahan membutuhkan waktu yang lebih lama. Akan tetapi, jaringan jenis ini lebih banyak menyelesaikan masalah dalam beberapa kasus pembelajaran.

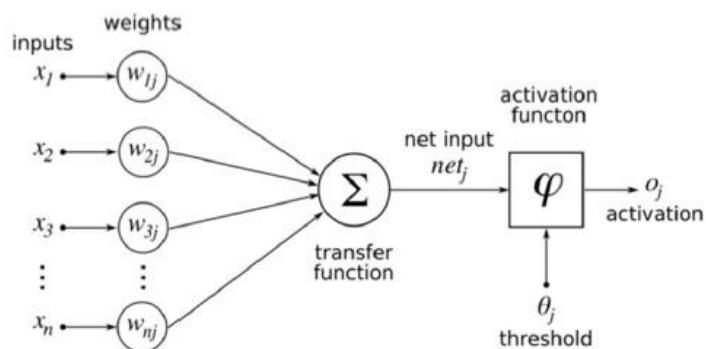


Gambar 2.4 Single Multi Network

2.8.2 Fungsi Aktivasi

Fungsi aktivasi digunakan untuk mengolah suatu inputan informasi.

Ilustrasi fungsi aktivasi dapat dilihat pada gambar berikut :



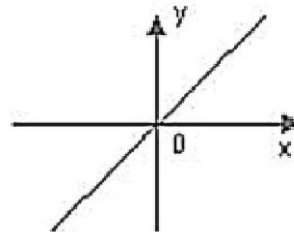
Gambar 2.5 Struktur ANN

Terdapat fungsi transfer yang mengubah sejumlah input ($X_1, X_2, X_3 \dots X_n$) yang memiliki bobot ($W_{1j}, W_{2j} \dots W_{nj}$) serta bias menjadi nilai *input* (*netj*) bagi fungsi aktivasi. Kemudian fungsi aktivasi memproses nilai *input* untuk dapat dibandingkan dengan *threshold* yang mengaktivasi nilai menjadi sebuah *output*. Beberapa jenis dari fungsi aktivasi :

1. *Fungsi linear*

Fungsi *linear* digunakan untuk memprediksi atau aproksimasi linier. Pada fungsi linier output bertipe diskrit hasilnya sama dengan input. Sedangkan pada matlab fungsi aktivasi *linear* dikenal dengan nama **purelin**, dengan syntax : $Y = \text{purelin}(a)$.

Dirumuskan : $y = x$ (2.5)



Gambar 2.6 Fungsi Aktivasi *Linear*

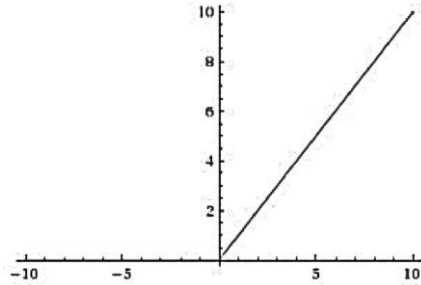
2. *Fungsi Rectified Non-Linear Unit (ReLU)*

Pada aktivasi ReLU membuat pembatas pada bilangan nol, yang mana nilai *input negatif* akan diubah menjadi nol, sedangkan nilai *input positif* akan diubah menjadi *output* dari *neuron* itu sendiri.

Dirumuskan : $f(x) = \max(0, x)$ (2.6)

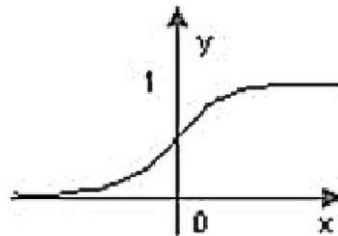
$$f(x) = 0, \text{ jika } x < 0$$

$$f(x) = x, \text{ jika } x \geq 0$$

Gambar 2.7 Fungsi Aktivasi *ReLU*

3. Fungsi sigmoid biner (*logsig*)

Fungsi ini memiliki nilai *range* 0 hingga 1 biasanya digunakan pada jaringan syaraf yang membutuhkan output bernilai kisaran interval 0 hingga 1, akan tetapi dapat juga dipakai pada jaringan syaraf bernilai outputnya 0 atau 1. Pada matlab syntax yang digunakan adalah $Y = \text{logsig}(a)$

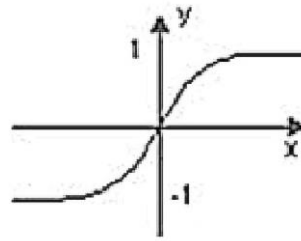
Gambar 2.8 Fungsi Aktivasi *Sigmoid Biner*

4. Fungsi sigmoid bipolar (*tansig*)

Fungsi sigmoid bipolar memiliki nilai *output* antara *range* 1 hingga -1.

Pada matlab syntax yang digunakan adalah $Y = \text{tansig}(a)$

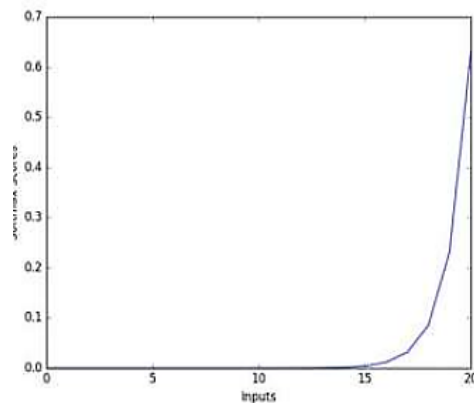
$$\text{Dirumuskan : } f(x) = \frac{2}{1+e^{-x}} - 1 \quad (2.7)$$

Gambar 2.9 Fungsi Aktivasi *Sigmoid Bipolar*

5. Fungsi Softmax

Fungsi ini merupakan distribusi probabilitas yang mana tidak hanya memetakan output pada kisaran $[0,1]$ tetapi dapat memetakan hingga mencapai 1. Fungsi *softmax* biasanya digunakan pada lapisan terakhir.

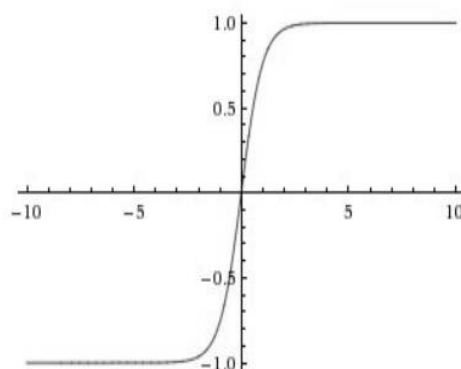
$$\text{Dirumuskan} = F(x_i) = \frac{\text{Exp}(x_i)}{\sum_{j=0}^k \text{Exp}(x_i)} \quad (2.8)$$

Gambar 2.10 Fungsi Aktivasi *Softmax*

6. Fungsi Tanh

Mengubah nilai *input* x menjadi *range* dari -1 sampai 1. Tanh merupakan perkembangan dari *sigmoid*.

$$\text{Dirumuskan} = \tanh \tanh (x) = 2\sigma(2x) - 1 \quad (2.9)$$

Gambar 2.11 Fungsi Aktivasi *Tanh*

2.8.3 Multi Layer Perceptron

Turunan dari *single perceptron* yang arsitekturnya lebih kompleks yaitu terdapat *input layer*, beberapa *hidden layer* dan *output layer*. Pada *input layer* terdapat nilai *input* yang diteruskan menuju *hidden layer* tanpa mengaktifkan *neuron* dengan fungsi aktivasi. Proses *hidden layer*, nilai dari *input* diterima setiap *neuron hidden layer*. Sehingga menghasilkan nilai *output hidden layer* kemudian dilakukan perhitungan nilai *input* dan bobot setelah menggunakan fungsi aktivasi pada setiap *neuron*. Jika memiliki lebih dari satu *hidden layer* maka dilakukan proses yang sama hingga terdapat hasil *output layer*. Proses tersebut dinamakan *feedforward neuron network*.

2.8.4 FeedForward Neural Network

Jaringan *neural network* proses bekerjanya menyerupai otak manusia dalam memproses data *input* sensorik, dan diterima sebagai *neuron input*. Kemudian *neuron* saling terhubung dengan node, dan sinyal yang ada pada *neuron* bekerja secara *paralel* digabungkan agar dapat menghasilkan

informasi dan reaksi. *Feedforward neural network* (FFNN) merupakan salah satu model *neural network* yang banyak digunakan untuk berbagai aplikasi (Kurniawan, 2017). Model FFNN terdiri dari satu lapis *input*, satu atau lebih lapis tersembunyi, dan satu lapis *output*. Adanya pengaruh dalam jumlah *neuron* yang terdapat pada lapisan tersembunyi yang digunakan untuk mempengaruhi *input* model dalam meminimalisir *error*. Dalam penerapannya, FFNN mengandung sejumlah parameter yang terbatas. Model FFNN melakukan perhitungan respon atau *output* dilakukan dengan memproses *input* menuju dari lapis ke lapis berikutnya secara tersembunyinya dan jumlah *neuron* yang terdapat pada masing-masing lapis. Jaringan FFNN yang cukup sederhana memiliki satu *hidden layer* yang dapat digunakan untuk memprediksi data *time series*.

Langkah-langkah dari perhitungan *feedforward neural network* :

1. Input yang terdiri dari $(x_i, i = 1, 2, 3, \dots, n)$ menerima sinyal dan meneruskannya ke *hidden layer*.
2. Pada *hidden layer* $(z_{net\ ij} = 1, 2, 3, \dots, n)$ penambahan bobot pada setiap *input*.

Menghitung keluaran dari *hidden layer*

$$z_{net\ ij} = \left(\sum_{ij=1}^n w_{ij} * x_{ij} \right) + b_{ij} \quad (2.10)$$

Menghitung fungsi menggunakan aktivasi

$$z_{ij} = f(z_{net\ ij}) = \frac{1}{1 + e^{-z_{net\ ij}}} \quad (2.11)$$

3. Menghitung unit o_{jk} ($jk = 1, 2, 3 \dots n$)

$$O_{net\ jk} = \left(\sum_{jk=1}^n w_{jk} * z_{jk} \right) + b_{jk} \quad (2.12)$$

Menghitung keluaran menggunakan aktivasi.

$$O_{net\ jk} = f(O_{net\ jk}) = \frac{1}{1+e^{-O_{net\ jk}}} \quad (2.13)$$

BAB III

DESAIN DAN RANCANGAN SISTEM

3.1 Fokus Penelitian

Peneliti ini berfokus pada pembuatan *game* dengan mengatur penyeimbangan kesulitan pada level menggunakan *artificial neural network* sebagai algoritmanya. Adanya tingkat penyeimbangan kesulitan berfungsi memberikan tantangan bagi pemain sehingga *game* menjadi lebih seru dengan adanya tingkat kesulitan yang bervariasi dan beragam. Dalam *game* konsep penyeimbangan kesulitan mengacu pada pertanyaan afektif, kognitif dan psikomotorik. Terdapat elemen kabut tiruan yang diperoleh dari penelitian Cahyani (Cahyani, 2020) dan Jeylani (Jeylani, 2020). Pada penelitian ini menggunakan *Game Engine Unity 3D*.



Gambar 3.1 Alur Penelitian

Alur penelitian dan rancangan ini memiliki langkah-langkah yang bertujuan untuk mengembangkan *game* edukasi berkonten Pendidikan Islam dan Lingkungan Hidup menggunakan enam langkah alur penelitian sebagai berikut :

1. Menyelidiki permasalahan yang ada yaitu kurangnya minat anak-anak dalam mempelajari buku mengenai pendidikan islam dan lingkungan hidup sehingga dibangun suatu inovasi berupa *game* edukasi. Agar *game* edukasi tidak bersifat monoton maka dibuat suatu penyeimbangan skenario pada *game* yang mengarah pada profil pemain yang berbeda-beda untuk menstabilkan minat dari pemain.
2. Melakukan pengumpulan data dengan cara melakukan kuisioner usability yang hasilnya akan digunakan sebagai klasifikasi dari nilai inputan untuk menghitung nilai dari proses *training* dan *testing* yang akan dipakai sebagai dataset.
3. Rancangan dari *game* berupa *flowchart* dan *menu flow* yang akan menjelaskan alur bagaimana *game* berjalan. Terdapat *storyboard* yang menjelaskan tampilan dari sistem *game* yang akan di bangun.
4. Sistem akan menggunakan metode FFNN sebagai pengaturan penyeimbangan skenario dengan menggunakan 7 *inputan* dan 4 *outputan*.
5. Pelaksanaan pengujian dengan melibatkan beberapa arsitektur percobaan dengan mengganti *hidden layer* yang berbeda-beda. Nantinya arsitektur paling ideal yang akan digunakan.
6. Hasil dari pengujian dengan metode FFNN akan diimplementasikan berdasarkan profile pemain yang akan digunakan. Kemudian menghasilkan

output berupa penyeimbangan skenario yang akan dimainkan oleh pemain yaitu level awal *game* mulai dijalankan.

3.2 Desain *Game*

3.2.1 Storyboard

Game ini menceritakan seorang anak berusaha menjelajahi alam untuk dapat mengumpulkan benda dan harta karun. Di alam yang luas anak tersebut akan bertemu dengan hewan-hewan berbahaya. Jika anak tersebut bersentuhan dengan hewan maka ia akan kehilangan *healty*. Namun, jika dapat mengumpulkan barang akan menjadi *score* dan apabila mendapatkan harta karun maka anak tersebut harus menjawab isi harta karun berupa kuis seputar pendidikan islam dan lingkungan hidup.

Contoh Storyboard pada *game*:

1. *Splash Screen*

Berikut ini perancangan antar muka *splash screen* dapat dilihat pada gambar dibawah ini:



Gambar 3.2 *Splash Screen*

2. Main Menu

Berikut ini perancangan antar muka main menu dapat dilihat pada gambar dibawah ini:



Gambar 3.3 Main Menu

3. Tampilan Kuis

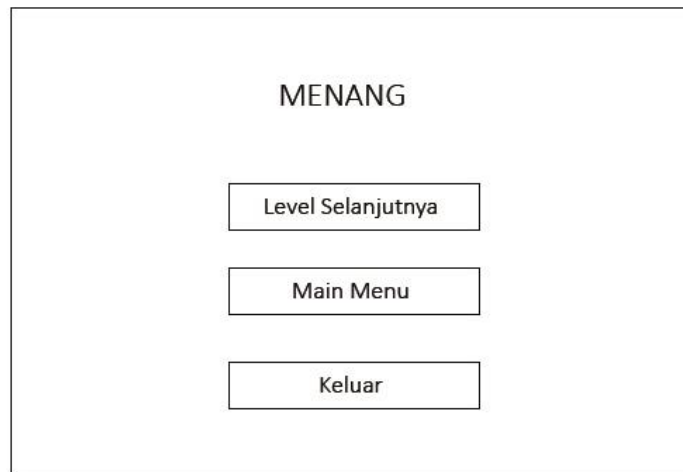
Berikut ini perancangan antar muka kuis dapat dilihat pada gambar dibawah ini



Gambar 3.4 Tampilan Kuis

4. Menang

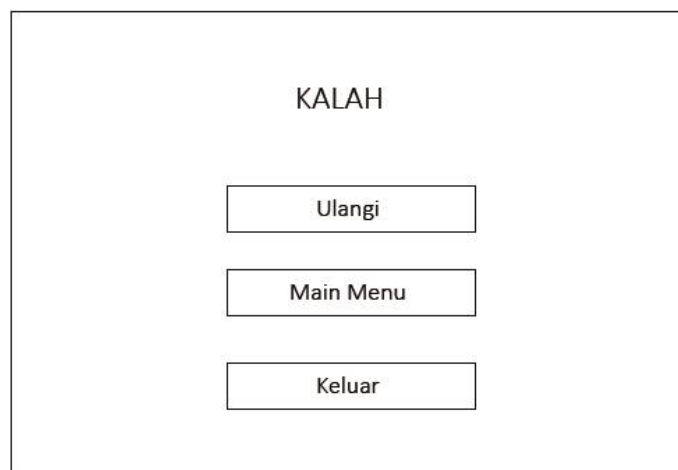
Berikut ini perancangan antar muka menang dapat dilihat pada gambar dibawah ini :



Gambar 3.5 Menang

5. Kalah

Berikut ini perancangan antar muka kalah dapat dilihat pada gambar dibawah ini :

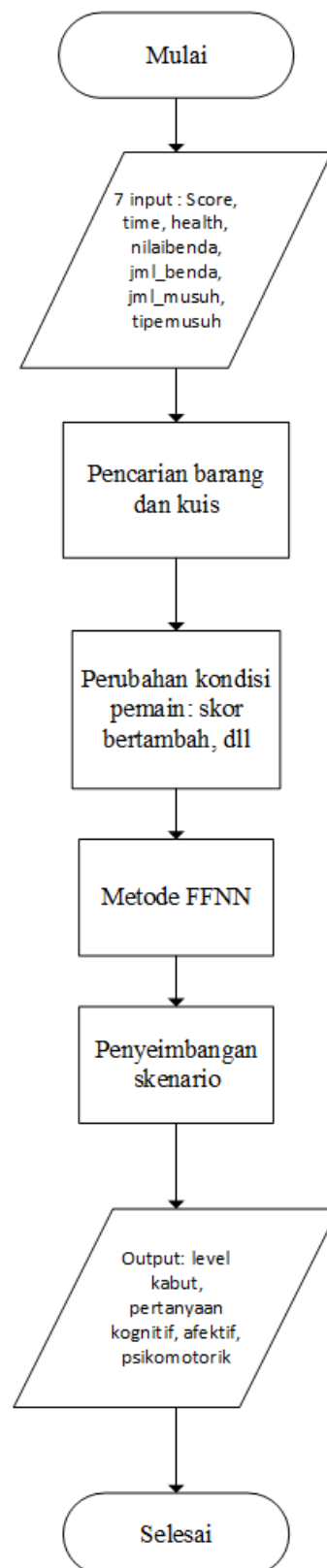


Gambar 3.6 Kalah

3.2.2 Desain Penyeimbangan Skenario

Penyeimbangan skenario berjalan dengan menyesuaikan aturan perilaku untuk objek *game*. Agar meningkatkan pengalaman pemain maka tingkat skenario dalam permainan menjadi sangat penting. Alur dari proses penyeimbangan skenario yang melibatkan metode FFNN yaitu :

1. Pemain mengumpulkan benda yang berfungsi sebagai inputan penyeimbangan kesulitan yaitu variabel *score* serta *time* dan *health*, benda, nilai benda, tipe musuh, dan jumlah musuh.
2. Pencarian benda dan kuis untuk mendapatkan *score* tambahan sehingga terdapat perubahan kondisi pada pemain.
3. Kemudian dilakukan proses perhitungan metode FFNN untuk mendapatkan hasil penyeimbangan skenario berupa level kabut.
4. Hasil *output* berupa level, pertanyaan kognitif, afektif dan psikomotorik yang setiap tingkatan akan berbeda.



Gambar 3.7 Flowchart Game

3.2.3 Desain Alur Game

Pada perancangan *game* dibuat *gameplay* yang disimulasikan untuk menentukan level kabut serta terdapat pertanyaan kognitif, afektif, dan psikomotorik sebagai edukasi bagi pemain untuk dapat memunculkan keluaran dalam peningkatan menuju level selanjutnya. *Gameplay* pada *game* yaitu menjalankan sebuah misi dengan mengumpulkan benda-benda yang nantinya akan bernilai *score* dan menghindari musuh untuk dapat mempertahankan *health* agar dapat dengan terus memainkan *game*. Dengan menerapkan metode *Feedforward neural network* untuk dapat melakukan penyeimbangan kesulitan yang terdapat pada *game* sehingga pemain tidak mudah jenuh. Alur *game* dijelaskan dalam bentuk desain sistem *game* berikut :

Desain sistem *game* yang dikembangkan sebagai berikut :

1. Terdapat 7 *inputan* yang digunakan dalam sistem meliputi : *score*, *time*, *health*, nilai benda, jumlah benda, nilai musuh dan tipe musuh. Nilai inputan tersebut didapatkan dari data sementara sebelum dilakukan update data.
2. Menentukan nilai *minimal* dan *maximal* pada semua *inputan* untuk menghasilkan data yang sudah ternormalisasi sehingga data tersebut lebih mudah untuk dimodifikasi dan rentang nilainya dari setiap data juga semakin pendek.
3. *Inputan* yang sudah ternormalisasi akan dihitung menggunakan metode FFNN dengan nilai *weight* dan *bias* diambil secara acak dengan nilai -1 sampai 1. Hasil dari perhitungan akan dilakukan fungsi aktivasi pada setiap *output*.

4. *Output* berupa nilai yang akan digunakan untuk menentukan penyeimbangan skenario, berupa level kabur, pertanyaan kognitif, afektif, dan psikomotorik.



Gambar 3.8 *Block Diagram FFNN*

3.3 Data Yang Digunakan

Data *input* yang digunakan dengan variabel *score*, *time*, *health*, benda, nilai benda, tipe musuh, dan jumlah musuh. Dengan setiap *variabel* akan mempengaruhi tingkat kesulitan *game*. Benda yang terkumpulkan dapat terakumulasi sebagai poin tambahan. Ketujuh *inputan* tersebut akan di proses oleh metode sehingga dapat menentukan *level* pemain.

Tabel 3.1 Penentuan nilai variabel setiap level

Masukan								Keluaran		
Score	Time	Healty	Nilai Benda	Jumlah Benda	Jumlah Musuh	Tipe Musuh	Level Kabut	Afektif	Kognitif	Psiko motorik
10	15	3	2	20	2	1	1	A1-1	K1-1	P1-1
10	12	3	2	18	4	1	2	A1-2	K1-2	P1-2
10	10	3	3	16	5	2	3	A1-3	K1-3	P1-3
9	10	3	3	14	6	2	4	A1-4	K1-4	P1-4
9	7	3	4	12	6	3	5	A2-1	K2-1	P2-1
9	7	2	4	10	7	3	6	A2-2	K2-2	P2-2
7	5	2	5	8	7	4	7	A2-3	K2-3	P2-3
7	5	2	5	6	8	4	8	A3-1	K3-1	P3-1
5	2	2	6	4	9	5	9	A3-2	K3-2	P3-2
5	2	2	6	2	10	5	10	A3-3	K3-3	P3-3

Pada tabel 3.1 diatas variabel masukan akan diubah sesuai dengan keseimbangan skenario. Semakin bertambah tingkatan keseimbangan skenario yang dimainkan, maka tantangan yang dilewati akan semakin sulit. Penyeimbangan skenario tersebut berupa level kabut yang awalnya sedikit atau cerah semakin lama akan gelap dan pekat sehingga pemain lebih susah untuk mendapatkan target *score*.

3.4 Preparasi Data

Sebelum dilakukan proses perhitungan *feedforward neural network* nilai *input* data dinormalisasikan. Sehingga setiap variabel memiliki rentang nilai yang sama yaitu antara 0-1 (Windarto et al., 2018). Pada normalisasi menggunakan persamaan berikut :

$$X_i = \frac{x - \min}{\max - \min} \quad (3.1)$$

Keterangan :

X_i = Hasil konversi data

x = Nilai yang akan dikonversi

min = Nilai *minimal* dari keseluruhan data

max = Nilai *maximal* dari keseluruhan data

Selanjutnya data dinormalisasikan dengan persamaan rumus yang diatas sehingga memperoleh nilai pada tabel berikut :

Tabel 3.2 Nilai normalisasi

Score	Time	Health	Nilai Benda	Jumlah Benda	Tipe Musuh	Jumlah Musuh
0,000	1,00	1	0	0	0	0
0,111	0,889	0,882	0,000	0,083	0,000	0,250
0,222	0,778	0,765	0,250	0,167	0,250	0,375
0,333	0,667	0,647	0,250	0,417	0,250	0,500

0,444	0,556	0,529	0,500	0,583	0,500	0,500
0,556	0,444	0,412	0,500	0,667	0,500	0,625
0,667	0,333	0,294	0,750	0,750	0,750	0,625
0,778	0,222	0,176	0,750	0,917	0,750	0,750
0,889	0,111	0,059	1	0,917	1	0,875
1	0	0	1	1	1	1

Pada tabel 3.2 data tersebut diperoleh dari data sementara untuk menjadi *inputan* pada perhitungan metode *feedforward neural network*. Sebelumnya data tersebut dinormalisasikan. Hasil normalisasi kemudian dimasukan kedalam rumus metode *feedforward neural network* sehingga mendapatkan nilai yang berfungsi untuk menentukan keseimbangan skenario berupa tingkatan level yang dimainkan oleh pemain.

3.5 Artificial Neural Network-Feedforward

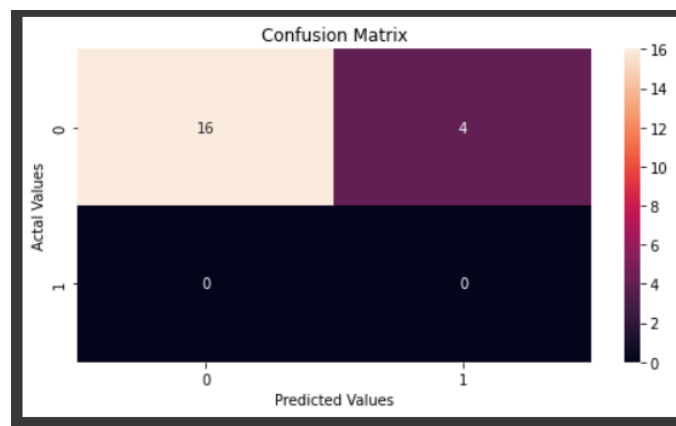
Pencarian arsitektur jaringan dengan melibatkan perbedaan antara *hidden layer* serta *neuron* yang terdapat dalam *hidden layer* sehingga dapat mengetahui hasil dari *accuracy*, *loss*, *precision*, *recall*, dan *fi-score*.

Tabel 3.3 Arsitektur Jaringan

Arsitektur Jaringan	Hidden Layer	Loss	Accuracy	Precision	Recall	F1 Score
7 – 5 – 4	1	0.00077246181 89968169	0.3000000119 2092896	0.847	0.827	0.818
7 – 12 – 4	1	0.00144552392 88508892	0.5666666626 930237	0.626	0.657	0.619
7 – 20 – 4	1	0.17499999701 976776	0.0016474617 877975106	1.0	0.17	0.30
7 – 4 – 3 – 4	2	0.09689617156 982422	0.4000000238 418579	1.0	0.60	0.75
7 – 8 – 6 – 4	2	0.00406895857 3043346	0.8000000119 20929	1.0	0.80	0.89
7 – 15 – 9 – 4	2	0.00227772747 1664548	0.0022777274 71664548	0.775	0.76	0.736
7 – 20 – 10 –	2	0.00196858076	0.3333333432	0,655	0,636	0,614

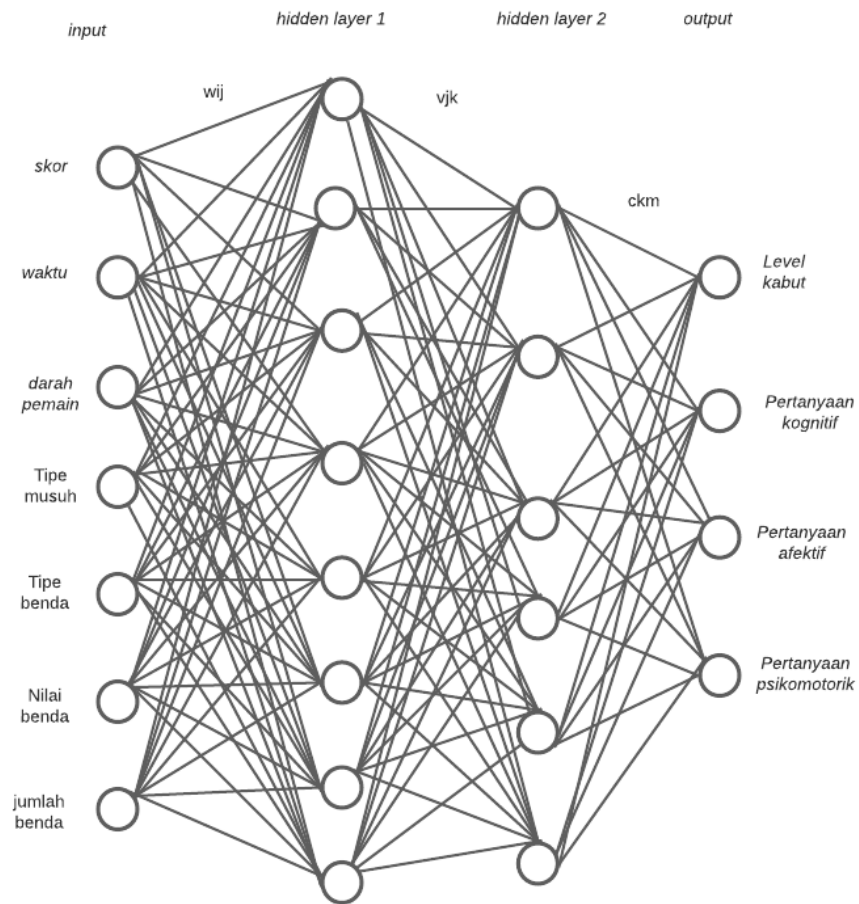
4		3787031	674408			
7 – 12 – 6 – 2 – 4	3	0.02500000037 252903	0.0975655466 3181305	1.0	0.30	0.50
7 – 16 – 12 – 8 – 4	3	0.00392700824 8865604	0.3666666746 1395264	0,817	0,762	0,746
7 – 18 – 13 – 9 – 4	3	0.00307055260 05476713	0.4250000119 2092896	1.0	0.42	0.60

Pada tabel diatas terdapat beberapa percobaan pencarian arsitektur. Terdapat arsitektur jaringan 7 – 8 – 6 – 4 yang nilai *accuracy* paling optimal diantara arsitektur yang lain sehingga peneliti melakukan pengujian dengan arsitektur tersebut.



Gambar 3.9 Ilustrasi confusion matrix

Pada tabel 3.2 terdapat data yang telah dinormalisasi kemudian dilakukan proses *feedforward*. Nilai *weight* dan *bias* diambil dari nilai random yaitu antara -1 sampai 1. Penulis menggunakan arsitektur jaringan 7 – 8 – 6 – 4 untuk menunjukkan perhitungan *feedforward neural network* yang menggunakan *multilayer perceptron*.



Gambar 3.10 Ilustrasi Arsitektur Jaringan 7 – 8 – 6 – 4

Tabel 3.4 Nilai bobot dan bias dari *input* menuju *hidden layer 1*

<i>Input</i>	<i>Neuron pada hidden layer 1</i>							
	<i>Neuron 1</i>	<i>Neuron 2</i>	<i>Neuron 3</i>	<i>Neuron 4</i>	<i>Neuron 5</i>	<i>Neuron 6</i>	<i>Neuron 7</i>	<i>Neuron 8</i>
Score	$W_{1.1}$	$W_{1.2}$	$W_{1.3}$	$W_{1.4}$	$W_{1.5}$	$W_{1.6}$	$W_{1.7}$	$W_{1.8}$
	0,75988	0,65763	-0,38763	0,95178	0,94070	0,11121	-0,70861	-0,07546
Time	$W_{2.1}$	$W_{2.2}$	$W_{2.3}$	$W_{2.4}$	$W_{2.5}$	$W_{2.6}$	$W_{2.7}$	$W_{2.8}$
	0,44616	0,95215	0,47694	-0,90491	0,19859	-0,28073	-0,15048	-0,41854
Healty	$W_{3.1}$	$W_{3.2}$	$W_{3.3}$	$W_{3.4}$	$W_{3.5}$	$W_{3.6}$	$W_{3.7}$	$W_{3.8}$
	0,06970	0,61491	0,85982	-0,36011	-0,93453	0,11847	0,85876	-0,04074
Benda	$W_{4.1}$	$W_{4.2}$	$W_{4.3}$	$W_{4.4}$	$W_{4.5}$	$W_{4.6}$	$W_{4.7}$	$W_{4.8}$
	-0,67172	-0,15541	-0,92738	0,19106	-0,57927	0,75848	0,28980	-0,48655
	$W_{5.1}$	$W_{5.2}$	$W_{5.3}$	$W_{5.4}$	$W_{5.5}$	$W_{5.6}$	$W_{5.7}$	$W_{5.8}$

Nilai Benda	-0,91082	-0,84963	-0,41149	-0,27339	0,64051	0,23945	0,78368	-0,66316
Musuh	$W_{6.1}$	$W_{6.2}$	$W_{6.3}$	$W_{6.4}$	$W_{6.5}$	$W_{6.6}$	$W_{6.7}$	$W_{6.8}$
	-0,66955	-0,92726	0,96024	0,93873	0,67350	0,25020	-0,28992	0,79918
Jumlah Musuh	$W_{7.1}$	$W_{7.2}$	$W_{7.3}$	$W_{7.4}$	$W_{7.5}$	$W_{7.6}$	$W_{7.7}$	$W_{7.8}$
	-0,96837	0,46743	0,32493	0,27616	-0,66706	0,42469	-0,05272	-0,16880
Bias	0,49550	0,01754	-0,91340	0,95392	-0,25838	0,11886	0,25701	0,67439

Tabel 3.3 merupakan nilai sebaran bobot serta bias yang terdapat pada *input layer* menuju *hidden layer* 1. Persamaan yang digunakan untuk menghitung *output feedforward* yang dilakukan pada arsitektur jaringan 7 – 8 – 6 – 4 dengan menggunakan nilai bobot dan bias yang diambil secara acak serta pada setiap *hidden layer* menggunakan fungsi aktivasi Relu dan *output layer* menggunakan fungsi aktivasi sigmoid. Menurut penelitian (Rahayu, 2021) pada *hidden layer* fungsi aktivasi yang digunakan secara umum yaitu ReLu sedangkan pada *output layer* yaitu sigmoid.

1. Menghitung pada *hidden layer* 1

$$Z_{in_j} = \sum_{i=1}^7 x_i w_{ij} + w_{0j} \quad (3.2)$$

$$Z_{in_1} = \left((0,1 \times 0,75988) + (0,9 \times 0,44616) + (0,9 \times 0,06970) + (0 \times (-0,67172)) + (0,1 \times (-0,91082)) + (0 \times (-0,66955)) + (0,3 \times (-0,96837)) \right) + 0,49550 = 0,720031$$

$$Z_{in_2} = 1,525595$$

$$Z_{in_3} = 0,27308$$

$$Z_{in_4} = -0,016185$$

$$Z_{in_5} = -0,915312$$

$$Z_{in_6} = 0,11233$$

$$Z_{in_7} = 0,854367$$

$$z_{in_8} = 0,16056$$

Kemudian menerapkan fungsi aktivasi ReLU

$$Z_j = f(Z_{in_j}) = f(0, Z_{in_j}) \quad (3.3)$$

$$z_1 = f(0, 0,720031) = 0,720031$$

$$z_2 = f(0, 1,525595) = 1,525595$$

$$z_3 = f(0, 0,27308) = 0,27308$$

$$z_4 = f(0, -0,016185) = 0$$

$$z_5 = f(0, -0,915312) = 0$$

$$z_6 = f(0, 0,11233) = 0,11233$$

$$z_7 = f(0, 0,854367) = 0,854367$$

$$z_8 = f(0, 0,16056) = 0,16056$$

2. Menghitung pada *hidden layer 2*

$$T_{in_k} = \sum_{i=1}^8 z_j v_{jk} + v_{ok} \quad (3.4)$$

$$t_{in_1} = \left((0,720031 \times 0,95909) + (1,525595 \times (-0,46388)) + (0,27308 \times 0,95961) + \right. \\ \left. (0 \times (-0,89421)) + (0 \times 0,08954) + (0,11233 \times 0,31902) + (0,854367 \times \right. \\ \left. (-0,85013) + (0,16056 \times (-0,87680)) \right) + 0,38928 = -0,197054$$

$$t_{in_2} = 0,354777$$

$$t_{in_3} = 0,38052$$

$$t_{in_4} = 0,652625$$

$$t_{in_5} = 0,591203$$

$$t_{in_6} = -0,23581$$

Kemudian menerapkan fungsi aktivasi ReLU

$$T_k = f(t_{in_k}) = f(0, t_{in_k}) \quad (3.5)$$

$$t_1 = f(0, -0,197054) = 0$$

$$t_2 = f(0, 0,354777) = 0,354777$$

$$t_3 = f(0, 0,38052) = 0,38052$$

$$t_4 = f(0, 0,652625) = 0,652625$$

$$t_5 = f(0, 0,591203) = 0,591203$$

$$t_6 = f(0, -0,23581) = 0$$

3. Menghitung nilai *output*

$$Y_{in_p} = \sum_{m=1}^6 t_m b_{mp} + b_{op} \quad (3.6)$$

$$y_{in_1} = \left((0 \times 0,67010) + (0,354777 \times 0,28121) + (0,38052 \times 0,30328) + \right. \\ \left. (0,652625 \times (-0,17110)) + (0,591203 \times 0,84985) + (0 \times 0,56726) \right) + \\ (-0,62364) = -0,017692$$

$$y_{in_2} = -0,275033$$

$$y_{in_3} = 0,475305$$

$$y_{in_4} = 0,000265$$

Kemudian menerapkan fungsi aktivasi sigmoid

$$Y_p = f(y_{in_p}) = \frac{1}{1+e^{-y_{in_p}}} \quad (3.7)$$

$$y_1 = \frac{1}{1+e^{-0,017692}} = 0,495577$$

$$y_2 = 0,431672$$

$$y_3 = 0,616639$$

$$y_4 = 0,500066$$

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

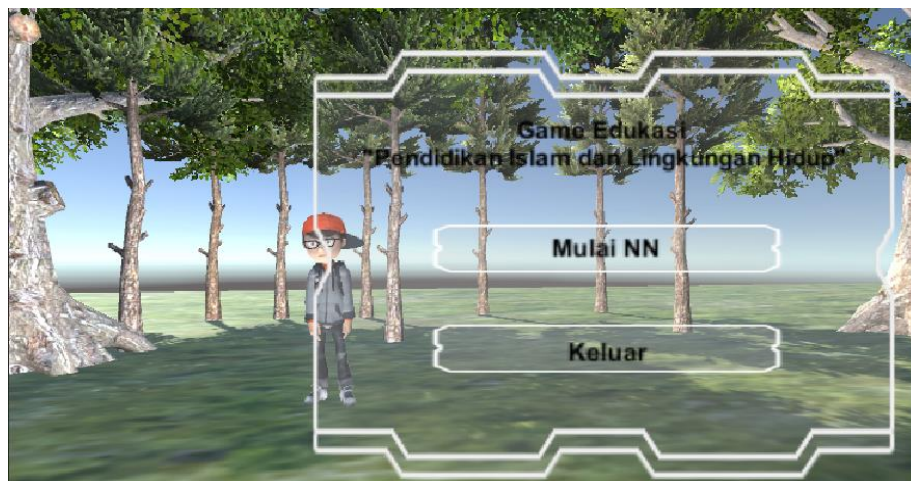
Pada Bab ini menjelaskan mengenai perancangan dan pengujian dari implementasi metode *feedforward neural network* terhadap *game* sehingga dapat mengetahui penyeimbangan skenario yang berupa level.

4.1.1 UI Game

Perancangan dari *storyboard* pada bab 3 yang terimplementasi pada *game* sebagai berikut.

1. Main Menu

Terdapat 2 opsi yaitu mulai *game* berfungsi untuk memainkan *game* dan keluar memberikan aksi keluar dari *game*.

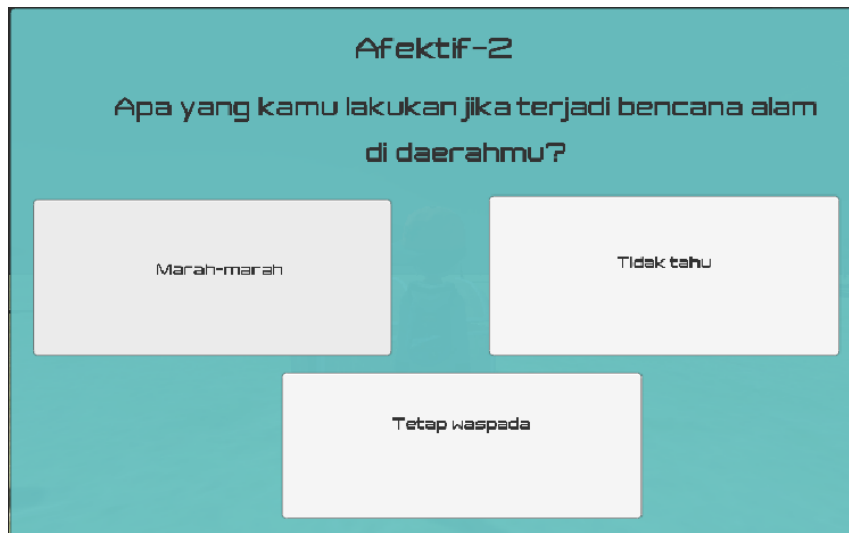


Gambar 4.1 *Main Menu*

2. Kuis

Tampilan Kuis yang berisi pertanyaan seputar pendidikan islam dan lingkungan hidup, terdapat 3 opsi yang ditawarkan namun hanya terdapat

satu jawaban yang benar. Apabila menjawab dengan benar maka akan mendapatkan poin tambahan.



Gambar 4.2 Kuis

4.1.2 Gameplay

1. *Game world*

Tampilan yang terdapat dalam *game*, karakter *game* berjalan maju mengumpulkan benda yang akan bernilai dan juga terdapat peti yang berisi pertanyaan berupa kuis. Terdapat *enemy* yang dapat mengurangi *healty* dari pemain apabila terkena. Level akan berubah bergantung pada perhitungan metode *Feedforward Neural Network*. Perubahan level biasanya beredabeda dari tiap pemain hal tersebut bergantung pada *skill* atau profil dari pemain.

Gambar 4.3 *Game world*

4.1.3 Perubahan kabut

Tampilan perubahan kabut berupa penyeimbangan skenario yang akan dilalui oleh pemain. Perubahan kabut berupa tingkatan kepekatan kabut yang semakin bertambah dengan *level* 1 sampai dengan 10. Tingkat kepekatan kabut membuat pemain semakin sulit untuk mencari benda dan mendapatkan penambahan *score*. Sehingga dapat menyulitkan pemain untuk mendapatkan poin. *Output level* kabut yang berubah-ubah dipengaruhi oleh personalisasi dari pemain itu sendiri. Terdapat 10 perubahan kabut yang mana tiap kabut akan memiliki nilai yang berbeda untuk kecepatan musuh, nilai benda, *health* pada setiap level.



Gambar 4.4 Kabut 1

Pada kabut level 1 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 0 pada level pertama yaitu 1, kecepatan musuh masih sangat rendah bahkan musuh tidak bergerak sehingga *healty* dari *player* masih dapat terjaga.

Tambahan Health = Tambahan *Health* yang bernilai 3 pada level 1, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 15 pada level 1, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 10 pada level 1, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 10 pada level 1.



Gambar 4.5 Kabut 2

Pada kabut level 2 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 3 pada level 2, kecepatan musuh masih sangat rendah dengan pergerakan lambat sehingga pemain masih mudah mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 3 pada level 2, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 12 pada level 2, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 10 pada level 2, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 10 pada level 2.



Gambar 4.6 Kabut 3

Pada kabut level 3 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 9 pada level 3, kecepatan musuh mulai meningkat dengan pergerakan cepat sehingga pemain perlu waspada dalam mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 3 pada level 3, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 10 pada level 3, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 10 pada level 3, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 10 pada level 3.



Gambar 4.7 Kabut 4

Pada kabut level 4 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 12 pada level 4, kecepatan musuh mulai meningkat dengan pergerakan cepat sehingga pemain perlu waspada dalam mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 3 pada level 4, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 10 pada level 4, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 9 pada level 4, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 9 pada level 4.



Gambar 4.8 Kabut 5

Pada kabut level 5 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 15 pada level 5, kecepatan musuh mulai meningkat dengan pergerakan cepat sehingga pemain perlu waspada dalam mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 3 pada level 5, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 7 pada level 5, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 9 pada level 5, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 9 pada level 5.



Gambar 4.9 Kabut 6

Pada kabut level 6 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 18 pada level 6, kecepatan musuh mulai meningkat dengan pergerakan cepat sehingga pemain perlu waspada dalam mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 2 pada level 6, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 7 pada level 6, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 9 pada level 6, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 9 pada level 6.



Gambar 4.10 Kabut 7

Pada kabut level 7 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 21 pada level 7, kecepatan musuh mulai meningkat dengan pergerakan sangat cepat sehingga pemain perlu waspada dalam mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 2 pada level 7, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 5 pada level 7, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 7 pada level 7, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 7 pada level 7.



Gambar 4.11 Kabut 8

Pada kabut level 8 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 24 pada level 8, kecepatan musuh mulai meningkat dengan pergerakan sangat cepat sehingga pemain perlu waspada dalam mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 2 pada level 8, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 5 pada level 8, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 7 pada level 8, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 7 pada level 8.



Gambar 4.12 Kabut 9

Pada kabut level 9 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 27 pada level 9, kecepatan musuh mulai meningkat dengan pergerakan cepat sehingga pemain perlu waspada dalam mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 2 pada level 9, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 2 pada level 9, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 5 pada level 9, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 5 pada level 9.



Gambar 4.13 Kabut 10

Pada kabut level 10 terdapat **KETERANGAN** yang memberikan informasi berupa :

Kecepatan Musuh = Kecepatan musuh yang bernilai 30 pada level 10, kecepatan musuh mulai meningkat dengan pergerakan musuh begitu cepat sehingga pemain perlu waspada dalam mempertahankan *health* nya.

Tambahan Health = Tambahan *Health* yang bernilai 2 pada level 10, *health* akan bertambah apabila berhasil mendapatkan benda.

Tambahan Waktu = Tambahan Waktu yang bernilai 2 pada level 10, waktu akan bertambah apabila berhasil mendapatkan benda yang berbentuk jam.

Tambahan Score = Tambahan *Score* yang bernilai 5 pada level 10, *Score* akan bertambah apabila berhasil mendapatkan benda-benda, masing-masing benda tersebut bernilai 5 pada level 10.

4.2 Pencarian Arsitektur Neural Network

4.2.1 Data

Data pengujian yang akan digunakan untuk mengolah arsitektur *Feedforward Neural Network*. Data tersebut dihitung dan nantinya akan diuji dengan beberapa arsitektur jaringan yang berbeda-beda sehingga mendapatkan hasil yang optimal. Pada tabel berikut berisi nilai dari *score*, *time*, *healty*, *nilai_benda*, *jumlah_benda*, *tipe musuh*, dan *jumlah_musuh*.

Tabel 4.1 Dataset

No	Score	Time	Healty	Nilai benda	Jumlah benda	Tipe musuh	Jumlah musuh
1	2	300	100	9	14	3	9
2	4	297	99	9	14	1	8
3	6	294	98	7	13	4	9
4	8	291	97	9	18	1	4
5	10	288	96	7	10	5	9
6	12	285	95	7	11	5	7
7	14	282	94	9	13	4	8
8	16	279	93	7	10	4	3
9	18	276	92	5	12	4	6
10	20	273	91	7	11	1	9
11	22	270	90	5	14	1	2
12	24	267	89	9	18	5	2
13	26	264	88	9	13	4	5
14	28	261	87	9	18	2	10
15	30	258	86	7	18	4	9
16	32	255	85	7	19	3	5
17	34	252	84	5	20	2	8
18	36	249	83	10	13	4	2
19	38	246	82	9	14	3	9
20	40	243	81	7	15	1	9
21	42	240	80	9	14	5	5
22	44	237	79	9	11	5	9
23	46	234	78	7	14	5	10
24	48	231	77	10	12	4	6
25	50	228	76	5	11	3	2
26	52	225	75	7	15	3	3

27	54	222	74	5	19	5	2
28	56	219	73	7	18	1	5
29	58	216	72	5	16	2	6
30	60	213	71	9	17	4	8
31	62	210	70	7	13	4	9
32	64	207	69	10	15	1	7
33	66	204	68	9	20	5	9
34	68	201	67	7	12	3	10
35	70	198	66	7	11	1	6
36	72	195	65	7	12	4	4
37	74	192	64	10	15	4	4
38	76	189	63	9	17	5	4
39	78	186	62	9	15	2	5
40	80	183	61	7	20	5	5
41	82	180	60	9	16	2	7
42	84	177	59	7	12	3	2
43	86	174	58	9	12	2	8
44	88	171	57	10	11	4	7
45	90	168	56	9	19	2	2
46	92	165	55	5	20	3	4
47	94	162	54	9	10	2	8
48	96	159	53	7	13	1	9
49	98	156	52	9	16	1	8
50	100	153	51	5	14	1	4
51	102	150	50	7	12	2	4
52	104	147	49	7	14	3	9
53	106	144	48	5	17	4	8
54	108	141	47	7	18	2	6
55	110	138	46	10	10	2	2
56	112	135	45	7	13	4	5
57	114	132	44	7	13	4	3
58	116	129	43	5	19	4	5
59	118	126	42	7	15	4	4
60	120	123	41	9	15	3	7
61	122	120	40	7	11	2	7
62	124	117	39	5	18	1	7
63	126	114	38	9	15	5	7
64	128	111	37	10	10	4	8
65	130	108	36	7	15	2	9
66	132	105	35	7	18	3	2
67	134	102	34	9	17	5	7

68	136	99	33	7	10	5	7
69	138	96	32	7	18	4	8
70	140	93	31	7	6	4	8
71	142	90	30	5	6	2	2
72	144	87	29	9	10	2	2
73	146	84	28	9	5	3	6
74	148	81	27	7	8	4	6
75	150	78	26	9	9	3	3
76	152	75	25	9	7	2	10
77	154	72	24	5	5	1	6
78	156	69	23	10	6	2	3
79	158	66	22	9	7	1	5
80	160	63	21	5	6	4	10
81	162	60	20	9	9	3	6
82	164	57	19	7	5	5	8
83	166	54	18	5	9	5	5
84	168	51	17	9	10	1	9
85	170	48	16	10	7	2	10
86	172	45	15	5	9	5	6
87	174	42	14	9	6	3	9
88	176	39	13	9	5	4	3
89	178	36	12	5	8	3	2
90	180	33	11	10	7	3	3
91	182	30	10	9	10	1	7
92	184	27	9	7	7	1	10
93	186	24	8	5	8	5	10
94	188	21	7	9	9	3	6
95	190	18	6	7	9	1	7
96	192	15	5	7	10	5	5
97	194	12	4	9	10	2	9
98	196	9	3	9	5	3	8
99	198	6	2	10	5	3	4
100	200	3	1	5	8	4	9

Pada tabel 4.1 terdapat data yang akan diuji untuk membantu dalam pencarian arsitektur yang optimal untuk digunakan pada sistem sehingga menghasilkan nilai akurasi yang baik. Data tersebut akan dibagi menjadi data *training* dan data *testing*.

4.2.2 Arsitektur *Neural Network*

Tahap ini menjelaskan eksperimen pencarian arsitektur jaringan yang optimal, Percobaan yang digunakan pada penelitian ini melibatkan 10 arsitektur jaringan dengan beberapa perbedaan diantaranya : *hidden layer*, fungsi aktivasi, serta *optimaizer* yang berbeda sehingga dapat lebih mudah untuk mengetahui bahwa arsitektur yang lebih optimal.

Hidden layer yang diuji yaitu : 1 *hidden layer*, 2 *hidden layer*, dan 3 *hidden layer* dengan jumlah *neuron* yang berbeda-beda. Untuk fungsi aktivasi menggunakan relu, sigmoid, dan tanh. Sedangkan *optimaizer* menggunakan adam dan SGD. Berikut adalah arsitektur jaringan yang akan diuji.

4.2.2.1 Arsitektur jaringan 7 – 5 – 4

Pengujian arsitektur jaringan 7 – 5 – 4 menggunakan 7 *input* data dan 1 *hidden layer* dengan jumlah *neuron* 5, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.2 dan tabel 4.3.

Tabel 4.2 Arsitektur 7 – 5 – 4

Arsitektur Jaringan	7-5-4	7-5-4	7-5-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.0007724618189 968169	0.00756642268 9706087	0.00233417260 46055555
Accuracy	0.3000000119209 2896	0.53333336114 8834	0.5

Tabel 4.3 Arsitektur 7 – 5 – 4

Arsitektur Jaringan	7-5-4	7-5-4	7-5-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01

Optimizer	SGD	SGD	SGD
Loss	0.0187147818505 764	0.11607934534 549713	0.01405750121 9213009
Accuracy	0.2333333343267 4408	0.23333333432 674408	0.20000000298 023224

Pada tabel 4.2 dan 4.3 Pencarian arsitektur 7 – 5 – 4 yang terdiri dari 1 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi yang menggunakan *logistic* dengan *optimaizer* Adam menghasilkan 0.533333361148834 nilai akurasi.

4.2.2.2 Arsitektur jaringan 7 – 12 – 4

Pengujian arsitektur jaringan 7 – 12 – 4 menggunakan 7 *input* data dan 1 *hidden layer* dengan jumlah *neuron* 12, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.4 dan tabel 4.5.

Tabel 4.4 Arsitektur 7 – 12 – 4

Arsitektur Jaringan	7-12-4	7-12-4	7-12-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.0014455239288 508892	0.00341125670 82047462	0.0022562795 784324408
Accuracy	0.5666666626930 23	0.20000000298 023224	0.1333333402 8720856

Tabel 4.5 Arsitektur 7 – 12 – 4

Arsitektur Jaringan	7-12-4	7-12-4	7-12-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.0049856635741 88948	0.12152289599 180222	0.0118250194 93699074

Accuracy	0.3000000119209 2896	0.40000000596 04645	0.2000000029 8023224
----------	-------------------------	------------------------	-------------------------

Pada tabel 4.4 dan 4.5 Pencarian arsitektur 7 – 12 – 4 yang terdiri dari 1 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi menggunakan Relu dengan *optimaizer* Adam menghasilkan 0.566666662693023 nilai akurasi.

4.2.2.3 Arsitektur jaringan 7 – 20 – 4

Pengujian arsitektur jaringan 7 – 12 – 4 menggunakan 7 *input* data dan 1 *hidden layer* dengan jumlah *neuron* 20, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.6 dan tabel 4.7.

Tabel 4.6 Arsitektur Jaringan 7 – 20 – 4

Arsitektur Jaringan	7-20-4	7-20-4	7-20-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.0013789139920 845628	0.00420976942 4051046	0.0022063811 775296926
Accuracy	0.5	0.46666666865 348816	0.2666666805 744171

Tabel 4.7 Arsitektur Jaringan 7 – 20 – 4

Arsitektur Jaringan	7-20-4	7-20-4	7-20-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.0044552027247 84613	0.11367524415 254593	0.0077618192 88134575
Accuracy	0.1666666716337 204	0.36666667461 395264	0.3333333432 674408

Pada tabel 4.6 dan tabel 4.7 Pencarian arsitektur 7 – 20 – 4 yang terdiri dari 1 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi yang menggunakan ReLU dengan *optimaizer* Adam menghasilkan 0.5 nilai akurasi.

4.2.2.4 Arsitektur jaringan 7 – 4 – 3 – 4

Pengujian arsitektur jaringan 7 – 4 – 3 – 4 menggunakan 7 *input* data dan 2 *hidden layer* dengan jumlah *neuron* 4 dan 3, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.8 dan tabel 4.9.

Tabel 4.8 Arsitektur Jaringan 7 – 4 – 3 – 4

Arsitektur Jaringan	7-4-3-4	7-4-3-4	7-4-3-4
Fungsi Aktivasi	ReLu	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.0009004363091 662526	0.00186898186 802864	0.0022903864 73760009
Accuracy	0.0666666701436 0428	0.10000000149 011612	0.3666666746 1395264

Tabel 4.9 Arsitektur Jaringan 7 – 4 – 3 – 4

Arsitektur Jaringan	7-4-3-4	7-4-3-4	7-4-3-4
Fungsi Aktivasi	ReLu	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.2003868371248 2452	0.12090918421 7453	0.0202392265 20061493
Accuracy	0.3333333432674 408	0.26666668057 44171	0.3666666746 1395264

Pada tabel 4.8 dan tabel 4.9 Pencarian arsitektur 7 – 4 – 3 – 4 yang terdiri dari 2 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi yang menggunakan Tanh dengan *optimaizer* Adam dan SGD yaitu nilai keduanya

0.36666667461395264 menghasilkan akurasi yang sama.

4.2.2.5 Arsitektur jaringan 7 – 8 – 6 – 4

Pengujian arsitektur jaringan 7 – 8 – 6 – 4 menggunakan 7 *input* data dan 2 *hidden layer* dengan jumlah *neuron* 8 dan 6, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.10 dan tabel 4.11.

Tabel 4.10 Arsitektur Jaringan 7 – 8 – 6 – 4

Arsitektur Jaringan	7-8-6-4	7-8-6-4	7-8-6-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.0930326804518 6996	0.00197778060 10097265	0.0018674627 64494121
Accuracy	0.6333333253860 474	0.33333334326 74408	0.2333333343 2674408

Tabel 4.11 Arsitektur Jaringan 7 – 8 – 6 – 4

Arsitektur Jaringan	7-8-6-4	7-8-6-4	7-8-6-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.0085443127900 36201	0.11902582645 41626	0.01577945984 8999977
Accuracy	0.4666666686534 8816	0.03333333507 180214	0.30000001192 092896

Pada tabel 4.10 dan tabel 4.11 Pencarian arsitektur 7 – 8 – 6 – 4 yang terdiri dari 2 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi yang menggunakan ReLU dengan *optimizer* Adam menghasilkan 0.6333333253860474 nilai akurasi.

4.2.2.6 Arsitektur jaringan 7 – 15 – 9 – 4

Pengujian arsitektur jaringan 7 – 15– 9 – 4 menggunakan 7 *input* data dan 2 *hidden layer* dengan jumlah *neuron* 15, dan 9, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.12 dan tabel 4.13.

Tabel 4.12 Arsitektur Jaringan 7 – 15– 9 – 4

Arsitektur Jaringan	7-15-9-4	7-15-9-4	7-15-9-4
Fungsi Aktivasi	ReLu	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.0022777274716 64548	0.00176432286 38917208	0.00234964955 5981159
Accuracy	0.0022777274716 64548	0.26666668057 44171	0.20000000298 023224]

Tabel 4.13 Arsitektur Jaringan 7 – 15– 9 – 4

Arsitektur Jaringan	7-15-9-4	7-15-9-4	7-15-9-4
Fungsi Aktivasi	ReLu	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.0954744964838 028	0.11957617104 053497	0.00794481299 8175621
Accuracy	0.4000000059604 645	0.56666666269 30237	0.16666667163 37204

Pada tabel 4.12 dan tabel 4.13 Pencarian arsitektur 7 – 15– 9 – 4 yang terdiri dari 2 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi yang menggunakan *Logistic* dengan *optimaizer* SGD menghasilkan 0.5666666626930237 nilai akurasi.

4.2.2.7 Arsitektur jaringan 7 – 20 – 10 – 4

Pengujian arsitektur jaringan 7 – 20– 10 – 4 menggunakan 7 *input* data dan

2 *hidden layer* dengan jumlah *neuron* 20, dan 10, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.14 dan tabel 4.15.

Tabel 4.14 Arsitektur Jaringan 7 – 20– 10 – 4

Arsitektur Jaringan	7-20-10-4	7-20-10-4	7-20-10-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.0019685807637 87031	0.0018255426548 421383	0.0018499729922 041297
Accuracy	0.3333333432674 408	0.3333333432674 408	0.2666666805744 171

Tabel 4.15 Arsitektur Jaringan 7 – 20– 10 – 4

Arsitektur Jaringan	7-20-10-4	7-20-10-4	7-20-10-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.0956054702401 1612	0.1187472566962 2421	0.0069961468689 14366
Accuracy	0.3666666746139 5264	0.0	0.2333333343267 4408

Pada tabel 4.13 dan tabel 4.15 Pencarian arsitektur 7 – 20– 10 – 4 yang terdiri dari 2 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi yang menggunakan ReLU dengan *optimizer* SGD menghasilkan 0.36666667461395264 nilai akurasi.

4.2.2.8 Arsitektur jaringan 7 – 12 – 6 – 2 – 4

Pengujian arsitektur jaringan 7 – 12– 6 – 2 – 4 menggunakan 7 *input* data dan 3 *hidden layer* dengan jumlah *neuron* 12, 6, dan 2, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.16 dan tabel 4.17.

Tabel 4.16 Arsitektur Jaringan 7 – 12– 6 – 2 – 4

Arsitektur Jaringan	7-12-6-2-4	7-12-6-2-4	7-12-6-2-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.1835662573575 9735	0.0017906358698 382974	0.0019540747161 95464
Accuracy	0.5333333611488 342	0.2000700298023 224	0.4666676868653 48816

Tabel 4.17 Arsitektur Jaringan 7 – 12– 6 – 2 – 4

Arsitektur Jaringan	7-12-6-2-4	7-12-6-2-4	7-12-6-2-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.0980424731969 8334	0.1215297877788 5437	0.0173198599368 3338
Accuracy	0.4666666686534 8816	0.0	0.5

Pada tabel 4.16 dan tabel 4.17 Pencarian arsitektur 7 – 12– 6 – 2 – 4 yang terdiri dari 3 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi ReLU dengan *optimaizer* Adam menghasilkan 0.5333333611488342 nilai akurasi.

4.2.2.9 Arsitektur jaringan 7 – 16 – 12 – 8 – 4

Pengujian arsitektur jaringan 7 – 16– 12 – 8 –4 menggunakan 7 *input* data dan 3 *hidden layer* dengan jumlah *neuron* 16, 12, dan 8, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.18 dan tabel 4.19.

Tabel 4.18 Arsitektur Jaringan 7 – 16– 12 – 8 –4

Arsitektur Jaringan	7-16-12-8-4	7-16-12-8-4	7-16-12-8-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam

Loss	0.0039270082488 65604	0.00155621813610 19611	0.00186444818 97354126
Accuracy	0.3666666746139 5264	0.40000000596046 45	0.30000001192 092896

Tabel 4.19 Arsitektur Jaringan 7 – 16 – 12 – 8 – 4

Arsitektur Jaringan	7-16-12-8-4	7-16-12-8-4	7-16-12-8-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.0050739930011 332035	0.11875145137310 028	0.00768853118 6431646
Accuracy	0.2000000029802 3224	0.13333334028720 856	0.36666667461 395264

Pada tabel 4.18 dan tabel 4.19 Pencarian arsitektur 7 – 16– 12 – 8 –4 yang terdiri dari 3 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi *Logistic* dengan *optimaizer* Adam menghasilkan 0.4000000059604645 nilai akurasi.

4.2.2.10 Arsitektur jaringan 7 – 18 – 13 – 9 – 4

Pengujian arsitektur jaringan 7 – 18 – 13 – 9 – 4 menggunakan 7 *input* data dan 3 *hidden layer* dengan jumlah *neuron* 18, 13, dan 9, dengan *learning rate* 0.01, dan epoch 1000. Hasil akurasi dan loss, ditunjukkan pada tabel 4.20 dan tabel 4.21.

Tabel 4.20 Arsitektur Jaringan 7 – 18– 13 – 9 –4

Arsitektur Jaringan	7-18-13-9-4	7-18-13-9-4	7-18-13-9-4
Fungsi Aktivasi	ReLU	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	Adam	Adam	Adam
Loss	0.0041538188233 9716	0.00156060163863 00325	0.00260804593 56307983
Accuracy	0.1000000014901 1612	0.46666666865348 816	0.30000001192 092896

Tabel 4.21 Arsitektur Jaringan 7 – 18– 13 – 9 – 4

Arsitektur Jaringan	7-18-13-9-4	7-18-13-9-4	7-18-13-9-4
Fungsi Aktivasi	ReLu	Logistic	Tanh
Epoch	1000	1000	1000
Learning Rate	0.01	0.01	0.01
Optimizer	SGD	SGD	SGD
Loss	0.0063969874754 54807	0.11955296993255 615	0.00646367343 1426287
Accuracy	0.3333333432674 408	0.0	0.5

Pada tabel 4.19 dan tabel 4.20 Pencarian arsitektur 7 – 18 – 13 – 9 – 4 yang terdiri dari 3 *hidden layer*, nilai akurasi yang tertinggi terdapat pada fungsi aktivasi yang menggunakan Tanh dengan *optimaizer* SGD menghasilkan 0.5 nilai akurasi.

4.3 Hasil Arsitektur Jaringan

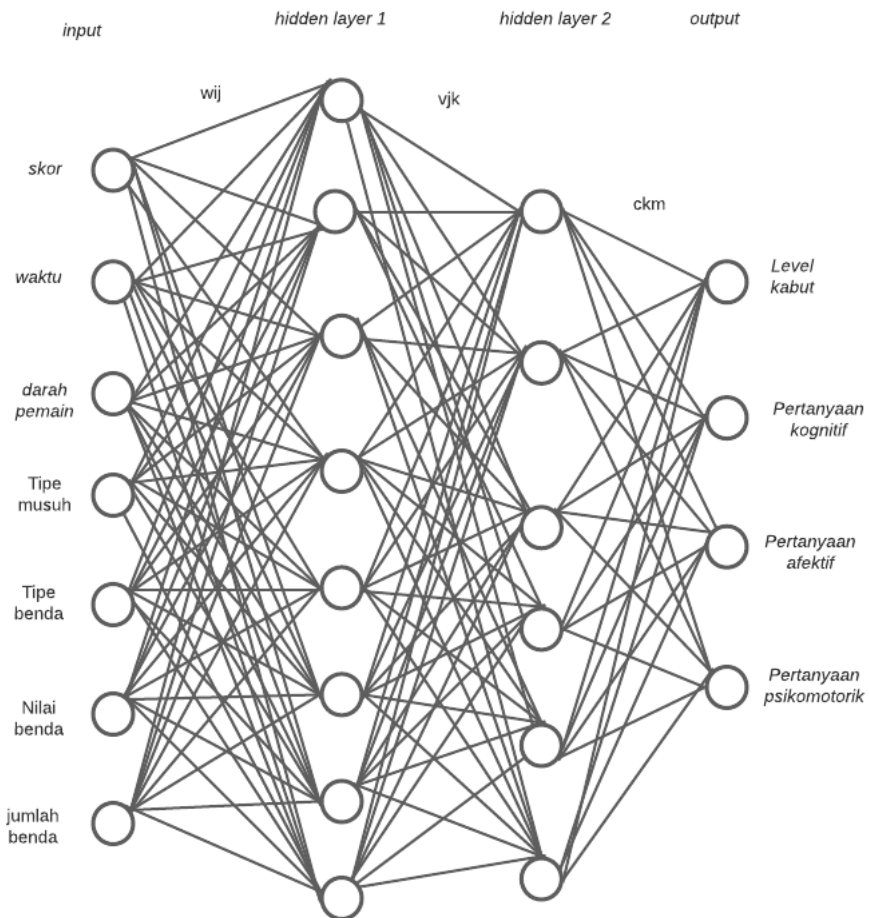
Hasil pengujian arsitektur jaringan dapat disimpulkan pada tabel berikut :

Tabel 4.22 Hasil Pengujian

No	Arsitektur Jaringan	Fungsi Aktivasi	<i>Optimaizer</i>	<i>Loss</i>	<i>Accuracy</i>
1	7 – 5 – 4	Logistic	Adam	0.0075664226 89706087	0.5333333611 488342
2	7 – 12 – 4	ReLU	Adam	0.0014455239 288508892	0.5666666626 930237
3	7 – 20 – 4	ReLU	Adam	0.0013789139 920845628	0.5
4	7 – 4 – 3 – 4	Tanh	SGD	0.0202392265 20061493	0.3666666746 1395264
5	7 – 8 – 6 – 4	ReLU	Adam	0.0930326804 5186996	0.6333333253 860474

6	7 – 15 – 9 – 4	Logistic	SGD	0.1195761710 4053497	0.5666666626 930237
7	7 – 20 – 10 – 4	ReLU	SGD	0.0956054702 4011612	0.3666666746 1395264
8	7 – 12 – 6 – 2 – 4	ReLU	Adam	0.1835662573 5759735	0.5333333611 488342
9	7 – 16 – 12 – 8 – 4	Logistic	Adam	0.0015562181 361019611	0.4000000059 604645
10	7 – 18 – 13 – 9 – 4	Tanh	SGD	0.0064636734 31426287	0.5

Hasil yang di peroleh dari tabel-tabel diatas dapat disimpulkan bahwa arsitektur yang optimal berada pada arsitektur 7 – 8 – 6 – 4 Hal tersebut disimpulkan dari nilai loss sebesar 0.09303268045186996 dan nilai akurasi sebesar 0.6333333253860474 dengan fungsi aktivasi ReLU dan *optimaizer* Adam. Arsitektur jaringan 7 – 8 – 6 – 4 menghasilkan nilai akurasi yang paling tinggi diantara seluruh percobaan yang telah diuji. Berikut adalah ilustrasi dari arsitektur jaringan yang dipilih :



Gambar 3.4 Arsitektur 7 – 8 – 6 – 4

Pada ilustrasi diatas, dilakukan perhitungan metode *feedforward neural network* dengan sebaran nilai bobot sebagai berikut :

Tabel 4.23 Nilai bobot dan bias dari *input* ke *hidden layer 1*

<i>Input</i>	<i>Neuron pada hidden layer 1</i>							
	<i>Neuron 1</i>	<i>Neuron 2</i>	<i>Neuron 3</i>	<i>Neuron 4</i>	<i>Neuron 5</i>	<i>Neuron 6</i>	<i>Neuron 7</i>	<i>Neuron 8</i>
Score	$W_{1.1}$	$W_{1.2}$	$W_{1.3}$	$W_{1.4}$	$W_{1.5}$	$W_{1.6}$	$W_{1.7}$	$W_{1.8}$
	- 0.4802 0765	1.0728 201	- 0.4877 614	0.2766 868	0.6922 3756	0.4893 7598	- 0.1776 4606	- 0.4849 666
Time	$W_{2.1}$	$W_{2.2}$	$W_{2.3}$	$W_{2.4}$	$W_{2.5}$	$W_{2.6}$	$W_{2.7}$	$W_{2.8}$

	0.0248 4357	0.1130 4035	0.2308 4572	0.4792 3568	- 0.6312 689	- 0.3852 9205	- 0.5107 792	0.1226 4923
Healty	$W_{3.1}$	$W_{3.2}$	$W_{3.3}$	$W_{3.4}$	$W_{3.5}$	$W_{3.6}$	$W_{3.7}$	$W_{3.8}$
	- 0.2964 3607	- 0.3305 812	0.7870 8476	- 0.1809 1552	0.3318 0013	0.1548 748	- 0.6067 041	0.3594 439
Benda	$W_{4.1}$	$W_{4.2}$	$W_{4.3}$	$W_{4.4}$	$W_{4.5}$	$W_{4.6}$	$W_{4.7}$	$W_{4.8}$
	- 0.2926 0907	0.0053 4408	- 0.1897 7393	- 0.1238 7379	0.0949 5375	- 0.2817 82	0.4135 0532	0.1949 0398
Nilai Benda	$W_{5.1}$	$W_{5.2}$	$W_{5.3}$	$W_{5.4}$	$W_{5.5}$	$W_{5.6}$	$W_{5.7}$	$W_{5.8}$
	- 0.0775 9339	0.3344 3928	0.5646 2234	- 0.6447 0077	0.0072 1175	- 0.5383 3693	0.1814 9693	0.3707 8393
Musuh	$W_{6.1}$	$W_{6.2}$	$W_{6.3}$	$W_{6.4}$	$W_{6.5}$	$W_{6.6}$	$W_{6.7}$	$W_{6.8}$
	0.1416 8447	0.2081 1361	0.1232 2763	- 0.3577 118	- 0.2576 0883	- 0.0512 108	0.1112 3174	0.3707 8393
Jumlah Musuh	$W_{7.1}$	$W_{7.2}$	$W_{7.3}$	$W_{7.4}$	$W_{7.5}$	$W_{7.6}$	$W_{7.7}$	$W_{7.8}$
	0.3051 9032	0.2507 6926	0.5071 811	- 0.4664 5862	0.0529 2074	- 0.1972 8762	- 0.5558 4383	0.3172 9648
Bias	0	0.3411 3997	- 0.1044 55	0.0237 2278	0.0575 1802	- 0.1333 4651	- 0.1942 585	0.2608 2236

4. Menghitung pada *hidden layer* 1

$$Z_{in_j} = \sum_{i=1}^7 x_i w_{ij} + w_{oj} \quad (4.1)$$

$$z_{in_1} = \left((0,111 \times (-0,48020765)) + (0,889 \times 0,02484357) + (0,882 \times (-0,29643607)) + (0 \times (-0,29260907)) + (0,083 \times (-0,07759339)) + (0 \times 0,14168447) + (0,25 \times 0,30519032) \right) + 0 = -0,22282$$

$$z_{in_2} = 0,359594$$

$$z_{in_3} = 0,914493$$

$$z_{in_4} = 0,150783$$

$$z_{in_5} = -0,12037$$

$$z_{in_6} = -0,37895$$

$$z_{in_7} = -1,32707$$

$$z_{in_8} = 0,743155$$

Kemudian menerapkan fungsi aktivasi ReLU

$$Z_j = f(Z_{in_j}) = f(0, Z_{in_j}) \quad (4.2)$$

$$z_1 = f(0, -0,22282) = 0$$

$$z_2 = f(0, 0,359594) = 0,359594$$

$$z_3 = f(0, 0,914493) = 0,914493$$

$$z_4 = f(0, 0,150783) = 0,150783$$

$$z_5 = f(0, -0,12037) = 0$$

$$z_6 = f(0, -0,37895) = 0$$

$$z_7 = f(0, -1,32707) = 0$$

$$z_8 = f(0, 0,743155) = 0,743155$$

Tabel 4.24 Nilai bobot dan bias dari *input* ke *hidden layer 2*

		Neuron pada hidden layer 3						
		Neuron 1	Neuron 2	Neuron 3	Neuron 4	Neuron 5	Neuron 6	
Neuron pada hidden layer ke 2	Neuron 1	$W_{1,1}$	$W_{1,2}$	$W_{1,3}$	$W_{1,4}$	$W_{1,5}$	$W_{1,6}$	
		0.455809 95	0.6026705 5	0.206522 35	0.136668 92	0.394784 06	0.036399 48	
	Neuron 2	$W_{2,1}$	$W_{2,2}$	$W_{2,3}$	$W_{2,4}$	$W_{2,5}$	$W_{2,6}$	
		0.872946 1	- 0.3239385 5	- 0.858570 93	- 0.620958 45	- 0.046765 93	- 0.219512 22	
	Neuron 3	$W_{3,1}$	$W_{3,2}$	$W_{3,3}$	$W_{3,4}$	$W_{3,5}$	$W_{3,6}$	
		- 0.313329 58	- -0.632327	- 0.144453 21	- 0.248187 21	- 0.280615 78	- 0.499494 76	
			$W_{4,1}$	$W_{4,2}$	$W_{4,3}$	$W_{4,4}$	$W_{4,5}$	$W_{4,6}$

Neuron 4	0.642631 05	0.5284875 6	- 0.496279 98	0.096225 38	0.381855 34	- 0.338432 88
Neuron 5	$W_{5.1}$	$W_{5.2}$	$W_{5.3}$	$W_{5.4}$	$W_{5.5}$	$W_{5.6}$
	0.368639 83	0.3823196 3	- 0.693760 7	0.260380 03	0.481172 35	0.365739 97
Neuron 6	$W_{6.1}$	$W_{6.2}$	$W_{6.3}$	$W_{6.4}$	$W_{6.5}$	$W_{6.6}$
	- 0.179320 51	0.049728	- 0.574095 2	- 0.484039 78	0.338987 6	- 0.193237 99
Neuron 7	$W_{7.1}$	$W_{7.2}$	$W_{7.3}$	$W_{7.4}$	$W_{7.5}$	$W_{7.6}$
	- 0.376077	- 0.4403934 5	0.493569	- 0.237955 3	0.020506 97	- 0.250884 98
Neuron 8	$W_{8.1}$	$W_{8.2}$	$W_{8.3}$	$W_{8.4}$	$W_{8.5}$	$W_{8.6}$
	- 0.290267 4	- 0.5588025 5	0.003814 58	- 0.039687 51	-1.1398	- 0.033463 7
Bias	0.372060 54	- 0.0964760 3	- 0.249093 1	0	- 0.049863 7	- 0.211503 48

5. Menghitung pada *hidden layer 2*

$$T_{in_k} = \sum_{i=1}^8 z_j v_{jk} + v_{ok} \quad (4.3)$$

$$t_{in_1} = \left((0 \times 0,45580995) + (0,359594 \times 0,8729461) + (0,914493 \times (-0,313329581)) + (0,150783 \times (0,64263105)) + (0 \times 0,36863983) + (0 \times (-0,17932051)) + (0 \times ((-0,376077)) + (0,743155 \times (-0,2902674)) \right) + 0,38928 = 0,496327$$

$$t_{in_2} = -1,12681084$$

$$t_{in_3} = -0,761927405$$

$$t_{in_4} = -0,465243217$$

$$t_{in_5} = -1,11277224$$

$$t_{in_6} = -0,823121949$$

Kemudian menerapkan fungsi aktivasi ReLU

$$T_k = f(t_{in_k}) = f(0, t_{in_k}) \quad (4.4)$$

$$t_1 = f(0, -0,197054) = 0,496327$$

$$t_2 = f(0, 0,354777) = 0$$

$$t_3 = f(0, 0,38052) = 0$$

$$t_4 = f(0, 0,652625) = 0$$

$$t_5 = f(0, 0,591203) = 0$$

$$t_6 = f(0, -0,23581) = 0$$

Tabel 4.25 Nilai bobot dan bias dari *input* ke *hidden layer output*

		Neuron pada output layer			
		Neuron 1	Neuron 2	Neuron 3	Neuron 4
Neuron pada hidden layer ke 3	Neuron 1	$W_{1,1}$	$W_{1,2}$	$W_{1,3}$	$W_{1,4}$
		0.5124001	0.51217335	-0.58877194	0.51223963
	Neuron 2	$W_{2,1}$	$W_{2,2}$	$W_{2,3}$	$W_{2,4}$
		-0.70304793	0.665556	-0.7324973	0.45347095
	Neuron 3	$W_{3,1}$	$W_{3,2}$	$W_{3,3}$	$W_{3,4}$
		-0.03596966	-0.31060687	-0.73984987	0.14893067
	Neuron 4	$W_{4,1}$	$W_{4,2}$	$W_{4,3}$	$W_{4,4}$
		0.728508	-0.63302696	0.01068622	0.01011705
	Neuron 5	$W_{5,1}$	$W_{5,2}$	$W_{5,3}$	$W_{5,4}$
		-0.24157047	0.6711587	-0.67860115	0.05491217
	Neuron 6	$W_{6,1}$	$W_{6,2}$	$W_{6,3}$	$W_{6,4}$
		0.13033858	0.01478669	-0.37619475	-0.6173892
Bias	0.00151891	0.00118403	0	0.00141743	

6. Menghitung nilai *output*

$$Y_{in_p} = \sum_{m=1}^6 t_m b_{mp} + b_{op} \quad (4.5)$$

$$y_{in_1} = \left((0,496327 \times 0,5124001) + (0 \times (-0,70304793)) + (0 \times (-0,03596966)) + \right.$$

$$(0 \times 0,728508) + (0 \times (-0,24157047)) + (0 \times 0,13033858) + (-0,62364) =$$

$$0,255836922$$

$$y_{in_2} = 0,2553895$$

$$y_{in_3} = -0,29222342$$

$$y_{in_4} = 0,255655797$$

Kemudian menerapkan fungsi aktivasi sigmoid

$$T_k = f(t_{in_k}) = f(0, t_{in_k}) \quad (4.6)$$

$$t_1 = f(0, 0,255836922) = 0,255836922$$

$$t_2 = f(0, 0,2553895) = 0,2553895$$

$$t_3 = f(0, -0,29222342) = 0$$

$$t_4 = f(0, 0,255655797) = 0,255655797$$

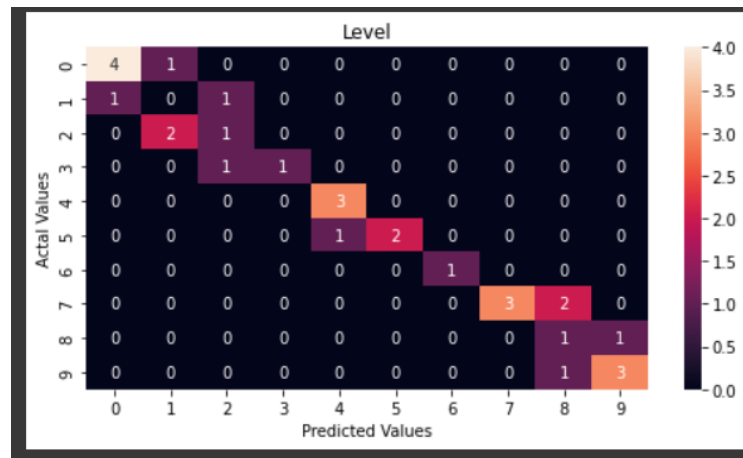
4.4 Skenario Uji

Arsitektur 7 – 8 – 6 – 4 menghasilkan nilai akurasi yang paling baik maka kemudian akan dilakukan pengujian dengan metode FFNN dibagi dalam 2 skenario uji, yaitu skenario 80:20 dan skenario 70:30. Pembagian skenario uji tersebut meliputi 80 data *training* dan 20 data *testing* dan 70 data *training* dan 30 data *testing*.

4.4.1 Pengujian 80:20

a. Confusion Matrix

Dari hasil pengujian menggunakan data *training* pada tabel 4.1 didapatkan confusion matrix sebagai berikut :



Gambar 4.14 Confusion matrix 80 : 20

b. Nilai *Precision*, *Recall*, *F1-Score*

Kemudian dari hasil *confusion matrix* dapat menentukan hasil dari *Precision*, *Recall*, *F1-Score*.

Tabel 4.25 Nilai *Precision*, *Recall*, *F1-Score*

Level	Precision	Recall	F1-Score
1	0.80	0.80	0.80
2	0.00	0.00	0.00
3	0.33	0.33	0.33
4	1.00	0.50	0.67
5	0.75	1.00	0.86
6	1.00	0.67	0.80
7	1.00	1.00	1.00
8	1.00	0.60	0.75
9	0.25	0.50	0.33
10	0.75	0.75	0.75

Hasil dari keseluruhan *Precision*, *Recall*, dan *F1-Score* dihitung maka mendapatkan nilai sebagai berikut :

Tabel 4.27 Rata-rata *Precision*, *Recall*, *F1-Score*

Precision	Recall	F1-Score
0.688	0.615	0.629

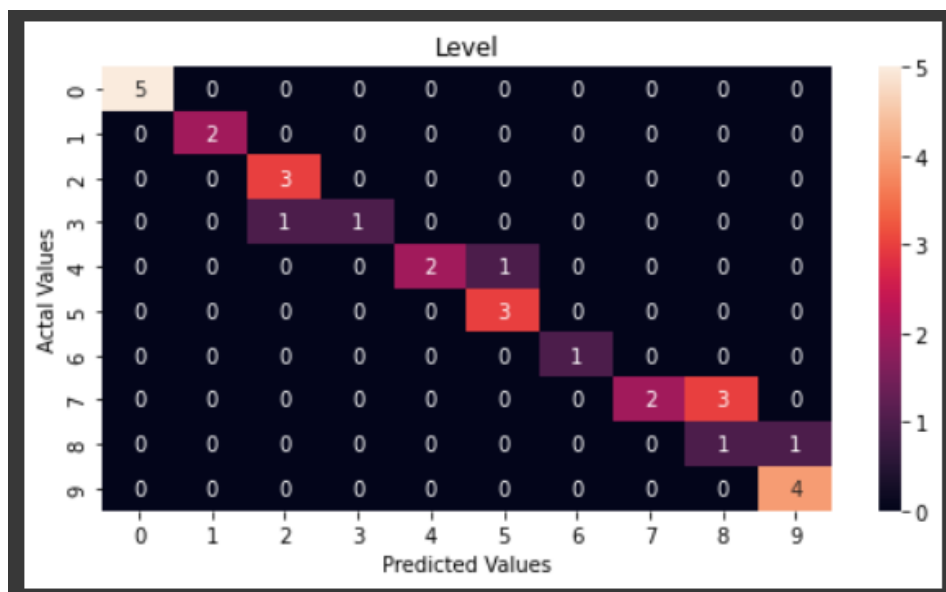
c. Nilai Akurasi

Pada pengujian 80:20 mendapatkan nilai akurasi sebesar 60%

4.4.2 Pengujian 70:30

a. Confusion Matrix

Dari hasil pengujian menggunakan data *training* pada tabel 4.1 didapatkan confusion matrix sebagai berikut :



Gambar 4.15 Confusion matrix 70 : 20

b. Nilai *Precision*, *Recall*, *F1-Score*

Kemudian dari hasil *confusion matrix* dapat menentukan hasil dari *Precision*, *Recall*, *F1-Score*.

Tabel 4.28 Nilai *Precision*, *Recall*, *F1-Score*

Level	Precision	Recall	F1-Score
1	1.00	1.00	1.00
2	1.00	1.00	1.00
3	0.75	1.00	0.86
4	1.00	0.50	0.67
5	1.00	0.67	0.80
6	0.75	1.00	0.86
7	1.00	1.00	1.00
8	1.00	0.40	0.57
9	0.25	0.50	0.33
10	0.80	1.00	0.89

Hasil dari keseluruhan *Precision*, *Recall*, dan *F1-Score* dihitung maka mendapatkan nilai sebagai berikut :

Tabel 4.29 Rata-rata *Precision*, *Recall*, *F1-Score*

Precision	Recall	F1-Score
0.855	0.807	0.798

c. Nilai Akurasi

Pada pengujian 70:30 mendapatkan nilai akurasi sebesar 90%

4.4.3 Hasil Skenario Uji

Pengujian yang dilakukan menggunakan 2 skenario uji kemudian menghasilkan nilai *precision*, *recall*, *f1 score*, dan akurasi yaitu

1. Pengujian 80:20, Hasilnya sebagai berikut :

- Rata-rata *precision* yang didapatkan adalah 0.688

- Rata-rata *recall* yang didapatkan adalah 0.615
- Rata-rata *f1-score* yang didapatkan adalah 0.629
- Akurasi yang didapatkan adalah 0.6 = 60%

2. Pengujian 70:30, Hasilnya sebagai berikut :

- Rata-rata *precision* yang didapatkan adalah 0.855
- Rata-rata *recall* yang didapatkan adalah 0.807
- Rata-rata *f1-score* yang didapatkan adalah 0.798
- Akurasi yang didapatkan adalah 0.9 = 90%

4.5 Integrasi Islam

Menuntut ilmu itu tidak hanya dengan membaca buku, mengikuti pelajaran namun, dapat melakukan beberapa inovasi belajar yang dapat memberikan pengetahuan lebih luas yaitu dengan cara bermain game edukasi yang mana game edukasi ini terdapat suatu kuis yang kontennya diambil dari buku pendidikan Islam dan lingkungan hidup. Konten berupa edukasi dengan dibalut oleh *game* dapat dengan mudah diingat oleh anak-anak sehingga tidak dihampiri oleh rasa bosan. Dengan adanya hal tersebut, maka dapat dengan mudah untuk peningkatan mutu pembelajaran baik disekolah maupun belajar mandiri. *Game* yang dirancang bertema pendidikan islam dan lingkungan hidup. Lingkungan yang baik berpengaruh dalam kualitas hidup manusia. Allah Subhanahu wa ta'ala berfirman dalam surah Ar-Rum ayat 41 :

لَعَلَّهُمْ عَمَلُوا الَّذِي بَعْضَ لِيُذِيقَهُمُ النَّاسِ آيَاتِي كَسَبَتْ بِمَا وَالْبَحْرِ الْبَرِّ فِي الْفَسَادِ ظَهَرَ

يَرْجِعُونَ

“Telah tampak kerusakan di darat dan di laut disebabkan karena perbuatan tangan manusia; Allah menghendaki agar mereka merasakan sebagian dari (akibat) perbuatan mereka, agar mereka kembali (ke jalan yang benar)”. (QS.Ar-Rum : 41)

Berdasarkan Tafsir Ibnu Katsir menjelaskan bahwa kerusakan di darat dan di laut disebabkan perbuatan tangan manusia, yaitu bermakna berkurangnya hasil tanaman dan buah-buahan karena banyak perbuatan maksiat yang dikerjakan oleh para penghuninya. Dalam riwayat hadist Bukhari-Muslim menjelaskan :

إِنَّ الْفَاجِرَ إِذَا مَاتَ تَسْتَرِيحُ مِنْهُ الْعِبَادُ وَالْبِلَادُ، وَالشَّجَرُ وَالذَّوَابُّ

“Apabila seorang pendurhaka mati, maka merasa gembiralah semua hamba, negeri, pepohonan, dan hewan-hewan dengan kematiannya itu.” (Bukhari-Muslim, Tafsir Ibnu Katsir).

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil dari implementasi *Feedforward Neural Network* pada *game* edukasi Pendidikan Islam dan Lingkungan Hidup, maka dapat diambil kesimpulan adalah *game* edukasi menggunakan 7 *input* yaitu *score*, *time*, *health*, nilai benda, jumlah benda, jumlah musuh, dan tipe musuh. *Inputan* tersebut berfungsi untuk menentukan hasil keseimbangan skenario. Nilai 7 variabel tersebut diproses menggunakan metode FFNN untuk mendapatkan keseimbangan skenario yang optimal.

Pengujian dilakukan dengan 10 arsitektur jaringan dengan melibatkan *hidden layer*, fungsi aktivasi dan *optimaizer* yang berbeda. Kemudian dihasilkan bahwa arsitektur jaringan optimal berada pada arsitektur jaringan 7 – 8 – 6 – 4 dengan skenario uji 70:30. yang mana hasil akurasi sebesar 90% dan nilai bobot serta bias akan diimplementasikan pada sistem .

5.2 Saran

Dalam penelitian ini penulis yakin implementasi keseimbangan skenario pada *game* pendidikan islam dan lingkungan hidup mempunyai banyak kekurangan sehingga perlu untuk dikembangkan secara terus-menerus agar lebih baik lagi.

Berikut saran yang diberikan oleh peneliti untuk penelitian selanjutnya :

1. Dalam penelitian ini, *game* masih bersifat *single player*. Maka penelitian selanjutnya dapat dikembangkan menjadi *multiplayer*, sehingga ada suatu intraksi dan kompetisi antar pemain.

2. Dalam penelitian selanjutnya dapat dikembangkan berupa penambahan *dataset* untuk menghasilkan jaringan yang lebih baik karena metode *artificial neural network* jika semakin banyak data maka pengenalan pola akan semakin baik.
3. Dalam penelitian ini, hanya terdapat objek penambahan *score*. Sehingga perlu dikembangkan penambahan objek yang berfungsi untuk mengurangi *score* pada saat bermain.

DAFTAR PUSTAKA

- Afrianto, F. (2019). *Rancang Bangun Game Edukasi Lingkungan, Bagi Anak Seusia Siswa Sd Berbasis Android*.
- Amalia, R., Akbar, Z., & Nurani, Y. (2021). Pengembangan Media Game Edukasi Adventure Cooking untuk Meningkatkan Perilaku Prososial Anak Usia Dini. *Jurnal Obsesi : Jurnal Pendidikan Anak Usia Dini*, 6(3), 1501–1513.
- Arif, Y. M., Harini, S., Nugroho, S. M. S., & Hariadi, M. (2021). An Automatic Scenario Control in Serious Game to Visualize Tourism Destinations Recommendation. *IEEE Access*, 9, 89941–89957.
- Aspi, M. (2022). Profesional Guru Dalam Menghadapi Tantangan Perkembangan Teknologi Pendidikan. *Indonesian Journal of Education (INJOE)*, 3(2), 291–300.
- Avi Shena, B. S., Sitohang, B., & Rukmono, S. A. (2019). Application of Dynamic Difficulty Adjustment on Evidence-centered Design Framework for Game Based Learning. *Proceedings of 2019 International Conference on Data and Software Engineering, ICoDSE 2019*.
- Caesarini, A. (2014). *Pada Game Bahasa Osing Anindita Caesarini Jurusan Teknik Informatika*.
- Cahyani, B. (2020). *Implementasi Perlin Noise Pada Simulasi Kabut Skripsi Oleh : Berlian Gita Cahyani*. Issue 1969.
- Cholik, C. A. (2021). *Teknologi Informasi, ICT*, 2(2), 39–46.
- Dan, P., Lingkungan, P., Dan, P., & Lingkungan, S. (1982).
- Haryanto, H. (2016). Reward Dinamis dalam Skenario Adaptif Menggunakan Metode Finite State Machine pada Game Edukasi. *Journal of Applied Intelligent System*, 1(2), 144–153.
- Hayati, A. N. (2018). *Game Pengenalan Tata Surya Menggunakan Metode Neural Network Backproagation Dalam Menentukan Level*. [Skripsi]. UIN Maulana Malik Ibrahim Malang.
- Huda, I. A. (2020). Irkham Abdaul Huda, Perkembangan Teknologi Informasi Dan Komunikasi (Tik) Terhadap Kualitas Pembelajaran Di Sekolah Dasar, *Jurnal Pendidikan dan Konseling*, Vol2No1, 2020:hal121. *Jurnal Pendidikan Dan Konseling (JPDK)*, 2(1), 121–125.
- Jeylani, U. (2020). Sistem Level Kabut pada Game Adventure Gunung Kelud Menggunakan Dynamic Difficulty Adjustment. In *Skripsi Teknik Informatika*

UIN Maulana MALik Ibrahim Malang.

- Kukiela, H. K. & D. (n.d.). *Copy of Neural Networks from Scratch in Python, Harrison Kinsle; Daniel Kukiela.pdf.*
- Kurniawan, M. A. (2017). Penerapan Metode Feed Forward Neural Network (FFNN) Backpropagation Untuk meramalkan harga saham. *Universitas Negeri Semarang (UNNES)*, 1–10.
- Menengah, S., Maxwell, K., Lumpur, K., Buka, J., Soalan, K., & Sehingga, I. N. I. (2008). *Pendidikan islam* (Vol. 5).
- Mukhlisin, M. A. (2014). *Game Sharraf Solitaire Menggunakan Neural Network Backpropagation Untuk Penentuan Level Pada Game Pembelajaran Tashrif Skripsi.* UIN Maulana Malik Ibrahim Malang.
- Perez, L. J. F., Calla, L. A. R., Valente, L., Montenegro, A. A., & Clua, E. W. G. (2016). Dynamic game difficulty balancing in real time using evolutionary fuzzy cognitive maps. *Brazilian Symposium on Games and Digital Entertainment, SBGAMES*, 24–32
- Rahayu, I. M. (2021). *Prediksi kesiapan sekolah menggunakan Machine Learning berbasis kombinasi Adam dan Nesterov Momentum.*
<http://digilib.uinsby.ac.id/id/eprint/49533>
- Rahmayanti, V. (2016). Pengaruh Minat Belajar Siswa dan Persepsi atas Upaya Guru dalam Memotivasi Belajar Siswa terhadap Prestasi Belajar Bahasa Indonesia Siswa SMP di Depok. *SAP (Susunan Artikel Pendidikan)*, 1(2), 206–216.
- Ryandhi, R. (2017). Penerapan Metode Artificial Neural Network (Ann) Untuk Peramalan Inflasi Di Indonesia (Ann). *Theses*, 231.
- Streicher, A., & Smeddinck, J. D. (2016). Personalized and adaptive serious games. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 9970 LNCS, 332–377.
- Syarif, M. S. (2016). *Penerapan Algoritma Backproagation Untuk Menentukan Level Bonus Dan Score Bonus Pada Game Edukasi Nahwu Menggunakan Kartu Berbasis Android.* 1–97.
- Wardani, D. (2022). Mengatasi Cold-Start Problem Menggunakan Artificial Neural Network Untuk Sistem Rekomendasi Pada Game Destinasi Wisata Kota Batu. *γ787*, 8.5.2017, 2003–2005. <https://www.who.int/news-room/fact-sheets/detail/autism-spectrum-disorders>
- Widiastuti, N. I. (2012). *Membangun Game Edukasi Sejarah Walisongo.*

Komputa : Jurnal Ilmiah Komputer Dan Informatika, 1(2), 41–48.
<https://doi.org/10.34010/komputa.v1i2.60>

Windarto, A. P., Lubis, M. R., & Solikhun. (2018). Model Arsitektur Neural Network Dengan Backpropogation Pada Prediksi Total Laba. *Kumpulan Jurnal Ilmu Komputer*, 05(02), 147–158.

Xia, W., & Anand, B. (2016). Game balancing with ecosystem mechanism. *Proceedings of 2016 International Conference on Data Mining and Advanced Computing, SAPIENCE 2016, Sapience*, 317–324.

Yootje, S., Tawas, D., Budhi, G. S., Purba, K. R., Studi, P., Informatika, T., Industri, F. T., Petra, U. K., & Siwalankerto, J. (2016). Pembuatan Survival Action Game Dengan Non - Player Character Berbasis Neural Network. *Jurnal Infra*, 2–6.