

BAB V

Hasil dan Pembahasan

Pada bab ini, akan dijelaskan hasil dan pembahasan dari implementasi yang telah dilakukan berdasarkan rancangan yang telah dibuat. Pembahasan akan dilakukan dengan bantuan sejumlah tangkapan layar antarmuka grafis.

5.1 Hasil

Dari hasil eksekusi model *Artificial Neural Network* (ANN) dengan dataset yang telah ditetapkan diperoleh hasil-hasil berikut ini:

A. Hasil untuk kedua lapisan dengan jumlah masing-masing *neuron* 100

Tabel 5. 1 Perbandingan Kedua Lapisan Tersembunyi dengan *neuron* 100

MODEL	Iterasi	Jumlah Neuron	Akurasi Pelatihan	Akurasi Pengujian	Durasi Pelatihan	Durasi Pengujian per Huruf
ANN 1 Lapisan Tersembunyi	100	100	97.86%	94.66%	19 detik	0 - 0.012 detik
ANN 2 Lapisan Tersembunyi	100	100, 100	83.83%	69.03%	20 detik	

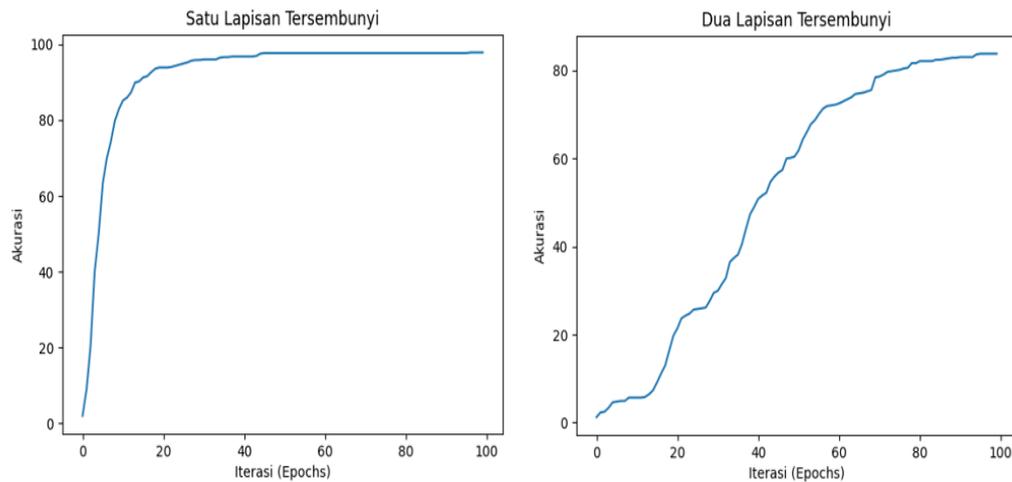
Dalam model *Artificial Neural Network* (ANN) satu lapisan tersembunyi, terlihat bahwa akurasi pelatihan (97.86%) dan akurasi pengujian (94.66%) cukup tinggi, menunjukkan bahwa model mampu mempelajari pola-pola dalam data dengan baik dan juga dapat menggeneralisasi dengan baik pada data baru. Namun, perlu diperhatikan bahwa ada sedikit perbedaan antara akurasi pelatihan dan pengujian, yang dapat menunjukkan adanya *overfitting* pada model.

Sementara itu, model *Artificial Neural Network* (ANN) dua lapisan tersembunyi menunjukkan akurasi pelatihan (83.81%) dan akurasi pengujian (69.03%) yang lebih rendah. Hal ini menandakan bahwa model ini mungkin mengalami masalah *underfitting*, di mana model tidak dapat mempelajari pola dengan baik dan tidak mampu menggeneralisasi pada data baru.

Perlu dicatat bahwa meskipun akurasi pelatihan tinggi dapat dianggap sebagai indikasi yang baik, penting untuk mempertimbangkan performa pada data

pengujian, karena akurasi pengujian yang lebih rendah dapat mengungkapkan kekurangan model dalam menggeneralisasi pada data yang belum pernah dilihat sebelumnya.

a. Perbandingan akurasi kedua lapisan tersembunyi dengan *neuron* 100



Gambar 5. 1 Tangkapan layar Akurasi Pelatihan Dengan Jumlah *Neuron* 100

1. Grafik Akurasi Pelatihan 1 Lapisan Tersembunyi

Kami melihat bahwa akurasi pelatihan meningkat pada awalnya kemudian meningkatkan akurasi secara perlahan. Jumlah iterasi model yang diperbanyak akan mencapai akurasi terbaik.

2. Grafik Akurasi Pelatihan Dua Lapisan Tersembunyi

Dalam grafik akurasi pelatihan tidak meningkat dengan lancar, ada beberapa kesalahan. Akan tetapi kita dapat melihat bahwa akurasi jauh meningkat dengan meningkatnya iterasi.

b. Pengujian huruf berdasarkan pelatihan dengan jumlah *neuron* 100



Gambar 5. 2 Tangkapan Layar Hasil Pengujian Acak Dengan Jumlah Neuron 100

Berdasarkan gambar di atas, saat dilakukan uji tes pada gambar huruf 'A' dan 'u' menggunakan ANN satu lapisan tersembunyi dan ANN dua lapisan tersembunyi, diperoleh hasil prediksi dan waktu prediksi sebagai berikut:

1. Gambar huruf 'A':

- ANN satu lapisan tersembunyi memprediksi: 'A'
- ANN dua lapisan tersembunyi memprediksi: 'A'
- Waktu prediksi kedua model: 0.006 detik

2. Gambar huruf 'u':

- ANN satu lapisan tersembunyi memprediksi: 'u'
- ANN dua lapisan tersembunyi memprediksi: 'B'

Penjelasan dan pembahasan terkait hasil prediksi dan waktu prediksi di atas adalah sebagai berikut:

1. Prediksi:

- ANN satu lapisan tersembunyi memprediksi 'A' untuk gambar huruf 'A' dan 'u' untuk gambar huruf 'u'. Hasil prediksi yang berbeda untuk kedua gambar menunjukkan bahwa model ini mampu membedakan dengan akurasi yang tinggi antara kedua huruf tersebut.
- ANN dua lapisan tersembunyi memprediksi 'A' untuk gambar huruf 'A' dan 'B' untuk gambar huruf 'u'. Tidak seperti ANN satu lapisan tersembunyi, hasil

prediksi yang kurang tepat untuk salah satu gambar uji, menunjukkan bahwa model ini juga mengalami kesulitan dalam membedakan antara huruf 'u' dan 'B'.

2. Waktu prediksi:

Waktu prediksi kedua model, baik ANN satu lapisan tersembunyi maupun ANN dua lapisan tersembunyi, adalah 0.006 detik. Waktu prediksi yang singkat menunjukkan efisiensi dan kecepatan dalam memproses gambar dan menghasilkan prediksi.

Pembahasan terkait hasil prediksi yang tidak akurat untuk model dua lapisan tersembunyi dapat disebabkan oleh beberapa faktor, antara lain:

- a. Keterbatasan representasi fitur: Mungkin terdapat perbedaan signifikan dalam fitur visual pada huruf 'u' yang sulit dipahami oleh ANN model dua lapisan tersembunyi.
- b. Jumlah data pelatihan: Jika model tidak diberikan jumlah data yang cukup yang mencakup variasi yang mencukupi dari kedua huruf tersebut, model mungkin tidak dapat mempelajari perbedaan antara keduanya secara efektif.

B. Hasil untuk kedua lapisan dengan jumlah masing-masing *neuron* 200

Tabel 5. 2 Perbandingan Kedua Lapisan Tersembunyi dengan *neuron* 200

MODEL	Iterasi	Jumlah Neuron	Akurasi Pelatihan	Akurasi Pengujian	Durasi Pelatihan	Durasi Pengujian per Huruf
ANN 1 Lapisan Tersembunyi	100	200	98.0%	93.59%	32 detik	0 - 0.005 detik
ANN 2 Lapisan Tersembunyi	100	200, 200	85.49%	68.32%	33 detik	

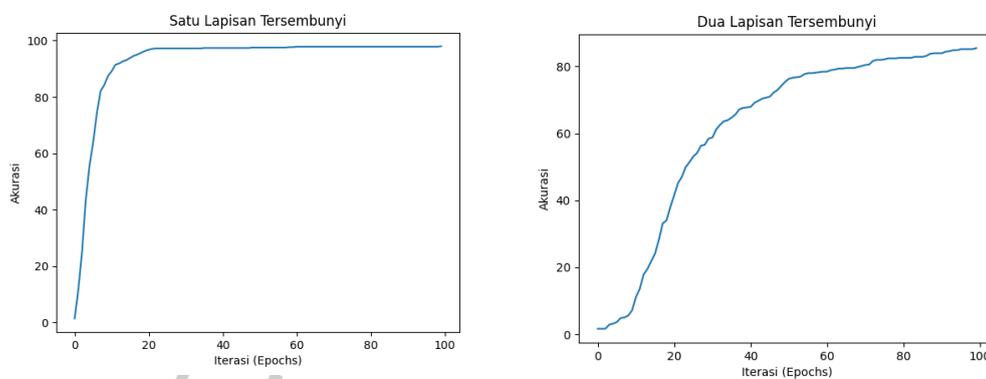
Pada tabel diatas adalah hasil perbandingan antara dua model *Artificial Neural Network* (ANN) yang memiliki jumlah lapisan tersembunyi yang berbeda. Model pertama memiliki satu lapisan tersembunyi dengan 200 *neuron*, sementara model kedua memiliki dua lapisan tersembunyi, masing-masing dengan 200 *neuron*. Berdasarkan tabel, model pertama dengan satu lapisan tersembunyi

menunjukkan hasil yang lebih baik. Model ini mencapai akurasi pelatihan sebesar 98.0% dan akurasi pengujian sebesar 93.59%. Durasi pelatihan model ini adalah 32 detik, dengan durasi pengujian per huruf sebesar 0.005 detik.

Sementara itu, model kedua dengan dua lapisan tersembunyi memiliki performa yang lebih rendah. Model ini mencapai akurasi pelatihan sebesar 85.49% dan akurasi pengujian sebesar 68.32%. Durasi pelatihan model kedua adalah 33 detik, dengan durasi pengujian per huruf yang sama dengan model pertama, yaitu 0.005 detik. Dengan demikian, tabel ini menunjukkan bahwa model dengan satu lapisan tersembunyi memiliki hasil yang lebih baik daripada model dengan dua lapisan tersembunyi dalam hal akurasi pelatihan dan pengujian.

Meskipun model kedua memiliki jumlah neuron yang sama dengan model pertama, penggunaan lapisan tersembunyi tambahan tidak memberikan peningkatan kinerja yang signifikan. Selain itu, model pertama juga memiliki waktu pelatihan yang lebih singkat. Oleh karena itu, dalam konteks ini, penggunaan satu lapisan tersembunyi tampaknya lebih efektif dan efisien.

a. Grafik perbandingan akurasi kedua lapisan tersembunyi dengan jumlah *neuron* 200



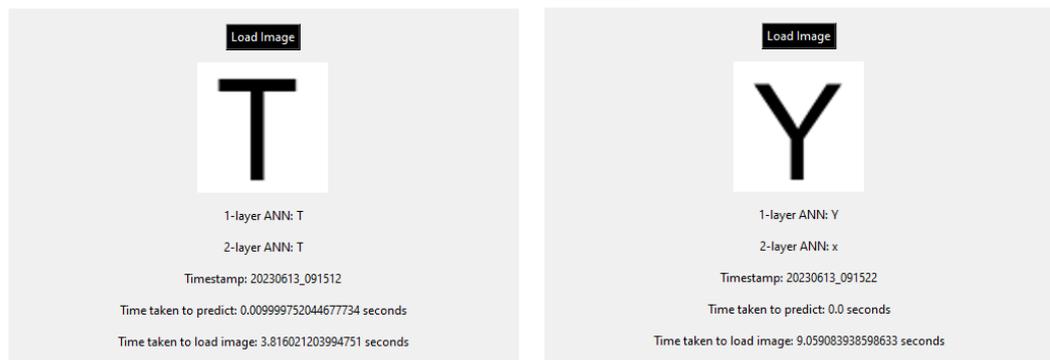
Gambar 5. 3 Tangkapan layar Akurasi Pelatihan Dengan Jumlah *Neuron* 200

Secara grafik, kita dapat melihat bahwa model dengan satu lapisan tersembunyi memiliki kurva peningkatan akurasi yang lebih mulus dibandingkan dengan model dengan dua lapisan tersembunyi. Pada model dengan satu lapisan tersembunyi, peningkatan akurasi cenderung naik secara konsisten dan lebih cepat. Garis grafik yang menggambarkan akurasi pelatihan dan pengujian cenderung

bergerak secara halus naik seiring dengan peningkatan iterasi pelatihan. Di sisi lain, pada model dengan dua lapisan tersembunyi, terlihat bahwa peningkatan akurasi tidak secepat dan semulus pada model pertama.

Grafik peningkatan akurasi mungkin menunjukkan fluktuasi dan kemungkinan peningkatan yang lebih lambat seiring dengan penambahan lapisan tersembunyi. Ini menunjukkan bahwa dalam kasus ini, model dengan satu lapisan tersembunyi memiliki peningkatan akurasi yang lebih lancar dan cepat, sedangkan model dengan dua lapisan tersembunyi cenderung mengalami peningkatan akurasi yang lebih lambat dan kurang stabil. Grafik ini dapat memberikan indikasi visual yang jelas tentang perbedaan dalam kecepatan peningkatan akurasi antara dua model.

b. Pengujian huruf berdasarkan pelatihan dengan jumlah *neuron* 200



Gambar 5. 4 Tangkapan Layar Hasil Pengujian Prediksi Huruf Dengan *Neuron* 200

Berdasarkan hasil pengujian prediksi gambar huruf 'T' dan 'Y' menggunakan model dengan lapisan tersembunyi yang berbeda, berikut adalah hasilnya:

Untuk huruf 'T':

1. Pada model dengan satu lapisan tersembunyi, hasil prediksi adalah 'T'. Model tersebut berhasil memprediksi dengan benar huruf 'T'.
2. Pada model dengan dua lapisan tersembunyi, hasil prediksi juga adalah 'T'. Model tersebut juga berhasil memprediksi dengan benar huruf 'T'.

Dengan demikian, baik model dengan satu lapisan tersembunyi maupun dua

lapisan tersembunyi mampu memprediksi huruf 'T' dengan benar.

Namun, untuk huruf 'Y':

1. Pada model dengan satu lapisan tersembunyi, hasil prediksi adalah 'Y'. Model tersebut berhasil memprediksi dengan benar huruf 'Y'.
2. Pada model dengan dua lapisan tersembunyi, hasil prediksi tidak sesuai, yaitu 'x'. Model tersebut gagal memprediksi huruf 'Y' dengan benar.

Dari hasil pengujian tersebut, dapat dilihat bahwa pada kasus huruf 'Y', model dengan satu lapisan tersembunyi memberikan hasil prediksi yang benar, sementara model dengan dua lapisan tersembunyi menghasilkan prediksi yang tidak akurat. Ini menunjukkan bahwa model dengan satu lapisan tersembunyi mungkin lebih efektif dalam memprediksi huruf 'Y' dalam konteks ini, dibandingkan dengan model dengan dua lapisan tersembunyi. Meskipun model kedua mungkin memiliki kapasitas yang lebih tinggi, namun dalam kasus ini, model pertama memberikan hasil yang lebih akurat.

C. Hasil untuk kedua lapisan dengan jumlah masing-masing *neuron* 300

Tabel 5. 3 Perbandingan Kedua Lapisan Tersembunyi dengan *neuron* 300

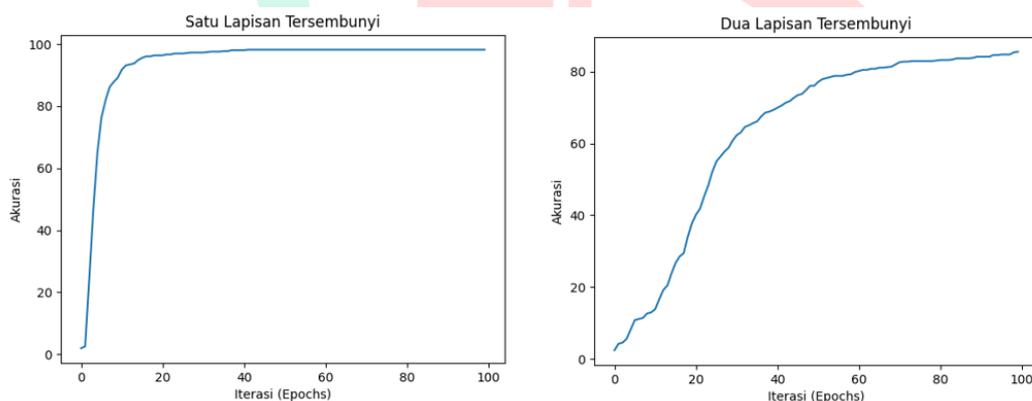
MODEL	Iterasi	Jumlah Neuron	Akurasi Pelatihan	Akurasi Pengujian	Durasi Pelatihan	Durasi Pengujian per Huruf
ANN 1 Lapisan Tersembunyi	100	300	98.16%	95.37%	46 detik	0 - 0.010 detik
ANN 2 Lapisan Tersembunyi	100	300, 300	85.49%	73.66%	49 detik	

Tabel tersebut memperlihatkan hasil perbandingan dari dua model *Artificial Neural Network* (ANN) yang memiliki jumlah lapisan tersembunyi yang berbeda. Model pertama menggunakan satu lapisan tersembunyi dengan 300 *neuron*, sedangkan model kedua menggunakan dua lapisan tersembunyi, masing-masing

dengan 300 *neuron*. Berdasarkan tabel, model dengan satu lapisan tersembunyi mencapai akurasi pelatihan sebesar 98.16% dan akurasi pengujian sebesar 95.37%. Durasi pelatihan untuk model ini adalah 46 detik, dengan durasi pengujian per huruf berkisar antara 0 hingga 0.010 detik.

Sementara itu, model dengan dua lapisan tersembunyi mencapai akurasi pelatihan sebesar 85.49% dan akurasi pengujian sebesar 73.66%. Durasi pelatihan model ini adalah 49 detik. Kesimpulannya, model dengan satu lapisan tersembunyi memiliki performa yang lebih baik daripada model dengan dua lapisan tersembunyi. Model pertama mencapai akurasi pelatihan dan pengujian yang lebih tinggi, serta waktu pelatihan yang lebih singkat. Ini menunjukkan bahwa dalam konteks ini, model dengan satu lapisan tersembunyi dan 300 *neuron* memberikan hasil yang lebih baik dan lebih efisien dibandingkan dengan model dua lapisan tersembunyi yang memiliki jumlah *neuron* yang sama.

a. Grafik perbandingan akurasi kedua lapisan tersembunyi dengan jumlah neuron 300



Gambar 5. 5 Tangkapan layar Akurasi Pelatihan Dengan Jumlah Neuron 300

Secara grafik, jika kita menggambarkan peningkatan akurasi dari model dengan satu lapisan tersembunyi dan model dengan dua lapisan tersembunyi, kita dapat melihat perbedaan dalam pola peningkatan akurasi. Pada model dengan satu lapisan tersembunyi, grafik peningkatan akurasi cenderung bergerak secara signifikan dan konsisten seiring dengan peningkatan iterasi pelatihan. Garis grafik yang menggambarkan akurasi pelatihan dan pengujian cenderung naik secara terus-menerus dengan kecepatan yang relatif konstan.

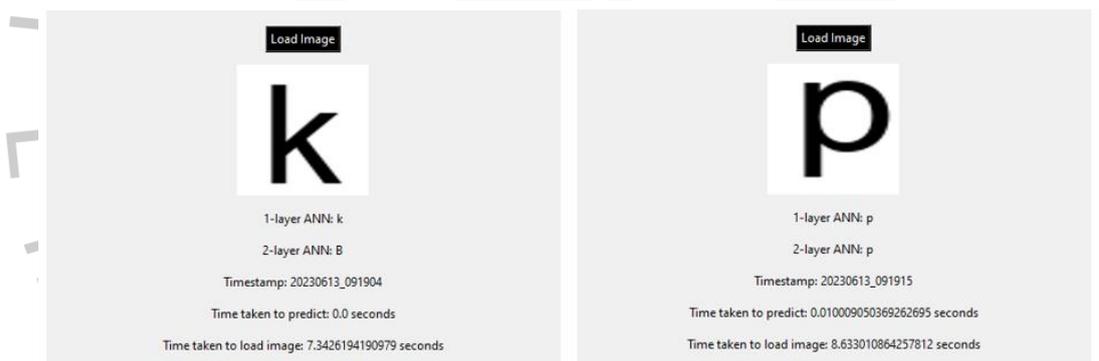
Di sisi lain, pada model dengan dua lapisan tersembunyi, terlihat bahwa pada awalnya peningkatan akurasi masih perlahan dan mungkin mencapai puncaknya

pada iterasi ke-30. Setelah itu, peningkatan akurasi menjadi lebih lambat dan naik perlahan. Grafik peningkatan akurasi mungkin menunjukkan fluktuasi atau bahkan datar pada beberapa iterasi.

Hal ini menunjukkan bahwa dalam kasus ini, model dengan satu lapisan tersembunyi memiliki peningkatan akurasi yang lebih mulus dan stabil seiring dengan peningkatan iterasi. Sementara itu, model dengan dua lapisan tersembunyi menunjukkan peningkatan akurasi yang perlahan setelah mencapai titik tertentu.

Grafik ini memberikan gambaran visual yang jelas tentang perbedaan pola peningkatan akurasi antara kedua model. Model dengan satu lapisan tersembunyi cenderung memberikan peningkatan akurasi yang lebih lancar dan terus-menerus sepanjang proses pelatihan, sementara model dengan dua lapisan tersembunyi mungkin memiliki peningkatan yang perlahan setelah mencapai iterasi tertentu.

b. Pengujian huruf berdasarkan pelatihan dengan jumlah neuron 300



Gambar 5. 6 Tangkapan Layar Hasil Pengujian Prediksi Huruf Dengan Neuron 300

Berdasarkan hasil pengujian prediksi gambar huruf 'k' dan 'p' menggunakan model dengan lapisan tersembunyi yang berbeda, berikut adalah hasilnya:

Untuk huruf 'k':

Pada model dengan satu lapisan tersembunyi, hasil prediksi adalah 'k'. Model tersebut berhasil memprediksi dengan benar huruf 'k'. Namun, pada model dengan dua lapisan tersembunyi, hasil prediksi adalah 'B'. Model tersebut menghasilkan prediksi yang salah dengan memprediksi huruf 'B' bukan huruf 'k'.

Menariknya, waktu prediksi untuk huruf 'k' dalam model dengan satu lapisan

tersembunyi tercatat sebagai 0 detik. Hal ini mungkin terjadi karena pengujian berjalan sangat cepat dan waktu prediksi sangat singkat, sehingga tidak terdeteksi perbedaannya.

Untuk huruf 'p':

Baik pada model dengan satu lapisan tersembunyi maupun dua lapisan tersembunyi, hasil prediksi adalah 'p'. Kedua model berhasil memprediksi huruf 'p' dengan benar. Waktu prediksi untuk huruf 'p' dalam model dengan satu lapisan tersembunyi tercatat sebagai 0.010 detik.

Dari hasil pengujian tersebut, dapat disimpulkan bahwa model dengan satu lapisan tersembunyi memberikan hasil prediksi yang lebih akurat untuk huruf 'k' dan huruf 'p'.

D. Hasil untuk kedua lapisan dengan jumlah masing-masing neuron 400

Tabel 5. 4 Perbandingan Kedua Lapisan Tersembunyi dengan neuron 400

MODEL	Iterasi	Jumlah Neuron	Akurasi Pelatihan	Akurasi Pengujian	Durasi Pelatihan	Durasi Pengujian per Huruf
ANN 1 Lapisan Tersembunyi	100	400	98.16%	95.01%	59 detik	0 - 0.010 detik
ANN 2 Lapisan Tersembunyi	100	400, 400	90.22%	78.29%	64 detik	

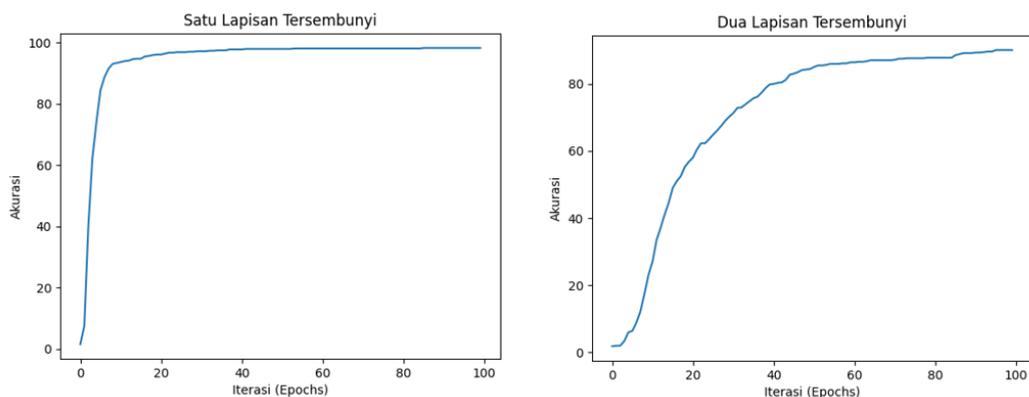
Tabel tersebut menampilkan hasil pelatihan dua model jaringan saraf tiruan (ANN) dengan iterasi sebanyak 100 dan jumlah neuron sebanyak 400. Model pertama menggunakan satu lapisan tersembunyi dengan akurasi pelatihan mencapai 98.16% dan akurasi pengujian mencapai 95.01%. Durasi pelatihan model ini adalah 59 detik, sedangkan durasi pengujian per hurufnya berkisar antara 0 hingga 0.010 detik.

Sementara itu, model kedua menggunakan dua lapisan tersembunyi dengan jumlah neuron 400 di setiap lapisan. Model ini memiliki akurasi pelatihan sebesar

90.22% dan akurasi pengujian sebesar 78.29%. Durasi pelatihan model ini adalah 64 detik. Meskipun model dengan dua lapisan tersembunyi ini memiliki akurasi yang lebih rendah dibandingkan dengan model satu lapisan tersembunyi, namun waktu pelatihan yang relatif lebih lama.

Dari tabel ini, dapat dilihat bahwa model dengan satu lapisan tersembunyi mencapai akurasi yang lebih tinggi dalam pelatihan dan pengujian. Selain itu, model tersebut juga memiliki waktu pelatihan yang lebih singkat dan durasi pengujian per huruf yang sangat cepat. Namun, model dengan dua lapisan tersembunyi tetap memberikan hasil yang dapat dipertimbangkan dengan akurasi yang cukup baik meskipun dengan waktu pelatihan yang lebih lama.

a. Grafik perbandingan akurasi kedua lapisan tersembunyi dengan jumlah neuron 400



Gambar 5. 7 Tangkapan layar Akurasi Pelatihan Dengan Jumlah Neuron 400

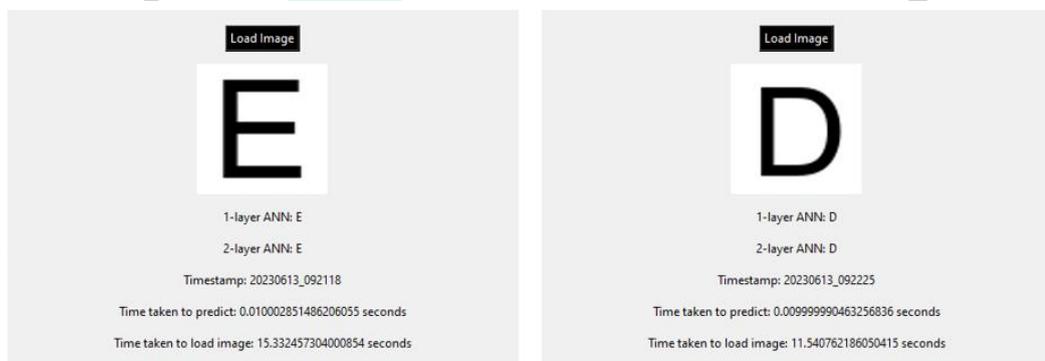
Secara grafik, jika kita membandingkan peningkatan akurasi dari model dengan satu lapisan tersembunyi dan model dengan dua lapisan tersembunyi, terdapat perbedaan yang menarik. Pada model dengan satu lapisan tersembunyi, grafik peningkatan akurasi cenderung tampak lebih mulus dan konsisten seiring dengan peningkatan iterasi pelatihan. Garis grafik yang menggambarkan akurasi pelatihan dan pengujian cenderung naik secara terus-menerus dengan kecepatan yang relatif konstan.

Sementara itu, pada model dengan dua lapisan tersembunyi, grafik peningkatan akurasi menunjukkan pola yang lebih dinamis. Dalam beberapa kasus, model dua lapisan tersembunyi dapat menunjukkan peningkatan akurasi yang lebih signifikan dibandingkan dengan model satu lapisan tersembunyi, terutama ketika jumlah neuron di bawah 400. Namun, perlu diperhatikan bahwa peningkatan

akurasi pada model dua lapisan tersembunyi masih tidak terlalu mulus dan cenderung lebih fluktuatif dibandingkan dengan model satu lapisan tersembunyi.

Dengan demikian, dapat disimpulkan bahwa meskipun model dengan satu lapisan tersembunyi menunjukkan peningkatan akurasi yang lebih mulus, model dengan dua lapisan tersembunyi dapat memberikan peningkatan akurasi yang lebih signifikan dalam beberapa kasus, terutama jika kita bandingkan jumlah neuron di bawah 400. Namun, perlu diperhatikan bahwa peningkatan akurasi pada model dua lapisan tersembunyi mungkin tidak sekonstan dan dapat lebih fluktuatif dibandingkan dengan model satu lapisan tersembunyi.

b. Pengujian huruf berdasarkan pelatihan dengan jumlah neuron 400



Gambar 5. 8 Tangkapan Layar Hasil Pengujian Prediksi Huruf Dengan Neuron 400

Berdasarkan hasil pengujian prediksi gambar huruf 'E' dan 'D' menggunakan model dengan lapisan tersembunyi yang berbeda, berikut adalah hasilnya:

Untuk huruf 'E':

1. Pada model dengan satu lapisan tersembunyi, hasil prediksi adalah 'E'. Model tersebut berhasil memprediksi dengan benar huruf 'E'.
2. Pada model dengan dua lapisan tersembunyi, hasil prediksi juga adalah 'E'. Model tersebut juga berhasil memprediksi huruf 'E' dengan benar.

Waktu prediksi untuk huruf 'E' adalah 0.010 detik untuk model dengan satu lapisan tersembunyi.

Untuk huruf 'D':

1. Pada model dengan satu lapisan tersembunyi, hasil prediksi adalah 'D'. Model tersebut berhasil memprediksi dengan benar huruf 'D'.
2. Pada model dengan dua lapisan tersembunyi, hasil prediksi juga adalah 'D'. Model tersebut juga berhasil memprediksi huruf 'D' dengan benar.

Waktu prediksi untuk huruf 'D' adalah 0.009 detik untuk model dengan satu lapisan tersembunyi.

Dari hasil pengujian tersebut, dapat disimpulkan bahwa baik model dengan satu lapisan tersembunyi maupun dua lapisan tersembunyi berhasil memprediksi huruf 'E' dan 'D' dengan benar. Waktu prediksi untuk keduanya cukup cepat, yaitu sekitar 0.009 - 0.010 detik. Hal ini menunjukkan bahwa kedua model memiliki performa yang setara dalam hal prediksi huruf 'E' dan 'D'.

E. Hasil untuk kedua lapisan dengan jumlah *epoch* 1000

Tabel 5. 5 Hasil Pengujian Dengan Epoch 1000

MODEL	Iterasi	Jumlah Neuron	Akurasi Pelatihan	Akurasi Pengujian	Durasi Pelatihan	Durasi Pengujian per Huruf
ANN 1 Lapisan Tersembunyi	1000	400	98.0%	96.44%	12 Menit 14 Detik	0 - 0.010 detik
ANN 2 Lapisan Tersembunyi	1000	400, 400	98.0%	97.50%	12 Menit 49 Detik	

Pada hasil pengujian diatas, terdapat dua eksperimen yang dilakukan dengan menggunakan model jaringan saraf tiruan dengan berbagai konfigurasi. Eksperimen pertama dilakukan dengan satu lapisan tersembunyi yang memiliki 1000 neuron, sedangkan eksperimen kedua dilakukan dengan dua lapisan tersembunyi yang masing-masing memiliki 400 neuron.

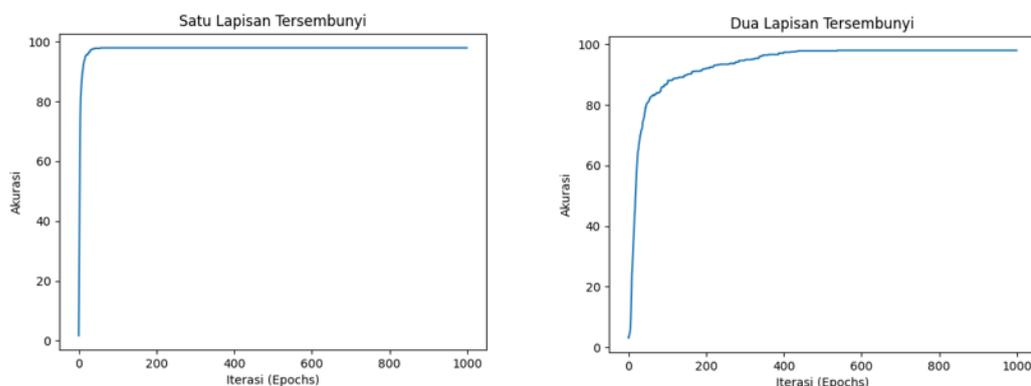
Dalam eksperimen pertama, proses pelatihan berjalan selama 12 menit 14 detik dengan kecepatan pengolahan sekitar 1.36 iterasi per detik. Hasilnya menunjukkan akurasi pelatihan sebesar 98.015% dengan tingkat pembelajaran (*learning rate*)

sebesar 0.05. Akurasi pengujian pada eksperimen ini mencapai 96.441%, dan nilai kehilangan (*loss*) akhir sebesar 0.018.

Sementara itu, dalam eksperimen kedua dengan dua lapisan tersembunyi, proses pelatihan berjalan selama 12 menit 49 detik dengan kecepatan pengolahan sekitar 1.30 iterasi per detik. Hasilnya menunjukkan akurasi pelatihan yang sama dengan eksperimen pertama, yaitu 98.015%, dengan tingkat pembelajaran 0.05. Namun, akurasi pengujian pada eksperimen kedua sedikit lebih tinggi, mencapai 97.509%. Nilai kehilangan (*loss*) akhir pada eksperimen kedua adalah 0.129.

Dari hasil ini, dapat disimpulkan bahwa kedua konfigurasi jaringan saraf tiruan berhasil memberikan tingkat akurasi yang tinggi dalam mengenali dan memprediksi data yang diberikan. Meskipun eksperimen kedua memiliki satu lapisan tersembunyi lebih sedikit, namun akurasi pengujian yang sedikit lebih tinggi menunjukkan bahwa penambahan lapisan tersembunyi dapat membantu meningkatkan kemampuan jaringan dalam mempelajari pola yang lebih kompleks.

a. Grafik perbandingan akurasi kedua lapisan tersembunyi dengan jumlah *epoch* 1000



Gambar 5. 9 Tangkapan layar Akurasi Pelatihan Dengan Jumlah Epoch 1000

Berdasarkan data dengan 1000 epoch, terlihat bahwa akurasi pada model dengan satu lapisan tersembunyi dan dua lapisan tersembunyi sudah mencapai maksimal. Grafik peningkatan akurasi menunjukkan bahwa untuk satu lapisan tersembunyi, pola grafik membentuk sudut 90 derajat, sedangkan untuk dua lapisan tersembunyi, peningkatan akurasi terlihat lebih baik dibandingkan dengan pengujian sebelumnya.

Dalam hal grafik, perbandingan peningkatan akurasi antara model dengan satu lapisan tersembunyi dan model dengan dua lapisan tersembunyi menunjukkan perbedaan yang menarik. Pada model dengan satu lapisan tersembunyi, grafik peningkatan akurasi cenderung menunjukkan kenaikan yang lancar dan konsisten seiring dengan iterasi pelatihan. Garis grafik yang menggambarkan akurasi pelatihan dan pengujian secara terus-menerus meningkat dengan kecepatan yang relatif konstan.

b. Pengujian huruf berdasarkan pelatihan dengan jumlah neuron 400



Gambar 5. 10 Tangkapan Layar Hasil Pengujian Prediksi Huruf Dengan Epoch 1000

Jika dilihat berdasarkan pengujian gambar huruf diatas, menunjukkan bahwa kedua model dapat mengenal gambar huruf dengan tepat. Sehingga ini menandakan bahwa kedua model telah melakukan pelatihan dengan baik. Dapat dilihat juga bahwa waktu yang dibutuhkan untuk pengenalan huruf sangat singkat yaitu hanya 0.010 detik.

5.2 Pembahasan

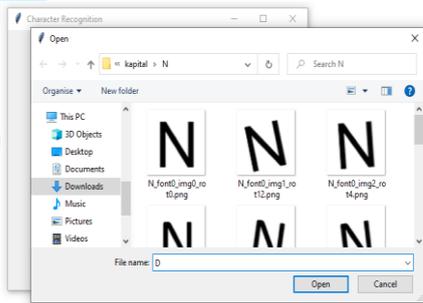
Dalam eksperimen ini, dilakukan pelatihan model *Artificial Neural Network* (ANN) dengan menggunakan satu lapisan tersembunyi dan dua lapisan tersembunyi pada variasi jumlah neuron, yaitu 100, 200, 300, dan 400, dengan iterasi sebanyak 100. Berdasarkan hasil pelatihan, terlihat bahwa model dengan satu lapisan tersembunyi cenderung memberikan akurasi yang lebih tinggi dibandingkan dengan model dua lapisan tersembunyi pada setiap variasi jumlah neuron. Peningkatan akurasi pada model dengan satu lapisan tersembunyi terjadi secara konsisten seiring dengan peningkatan jumlah neuron. Misalnya, pada model dengan 300 neuron, akurasi pengujian mencapai 95.374%, sedangkan pada model

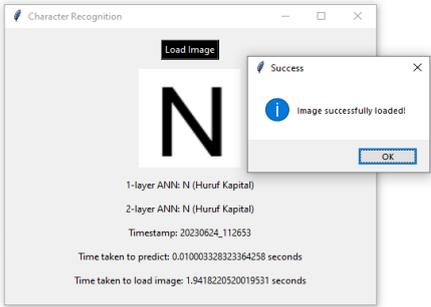
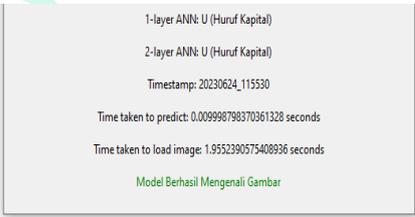
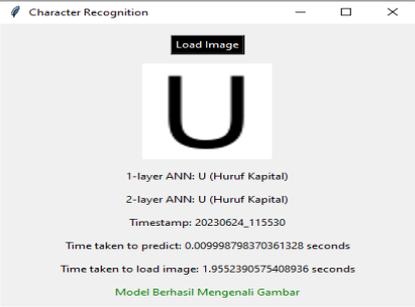
dua lapisan tersembunyi dengan jumlah neuron yang sama, akurasi pengujian hanya mencapai 73.665%.

Selain itu, terlihat bahwa model dengan satu lapisan tersembunyi cenderung memiliki waktu pelatihan yang lebih singkat dibandingkan dengan model dua lapisan tersembunyi. Hal ini dapat dilihat dari waktu eksekusi yang dicatat pada catatan pelatihan. Misalnya, pada model dengan 400 neuron, waktu pelatihan untuk satu lapisan tersembunyi adalah sekitar 59 detik, sedangkan pada model dua lapisan tersembunyi waktu pelatihan meningkat menjadi sekitar 64 detik. Secara keseluruhan, model dengan satu lapisan tersembunyi memberikan kombinasi antara akurasi yang lebih tinggi dan waktu pelatihan yang lebih efisien dibandingkan dengan model dua lapisan tersembunyi. Namun, penting untuk dicatat bahwa hasil ini dapat bervariasi tergantung pada dataset dan konfigurasi model yang digunakan. Oleh karena itu, perlu dilakukan eksperimen lebih lanjut untuk menggeneralisasi temuan ini ke berbagai masalah dan kondisi.

5.2.1 Pengujian Black Box

Tabel 5. 6 Tabel Pengujian Black Box

No	Pengujian	Hasil yang Diinginkan	Hasil Pengamatan	Kesimpulan
1	Load Image	Antarmuka membuka jendela pemilihan file		Berhasil

2	Pengguna memilih gambar	Gambar berhasil di-load dan ditampilkan dalam antarmuka grafis.		Berhasil
3	Predict Image	Prediksi huruf berhasil dilakukan berdasarkan gambar yang di-load.		Berhasil
4	Save Image	Gambar berhasil disimpan dengan timestamp sesuai waktu simpan.		Berhasil

5	Update Timestamp in Frame	Timestamp dan waktu load berhasil di-update pada antarmuka grafis.		Berhasil
---	---------------------------	--	--	----------

Berdasarkan hasil pengujian black box yang dilakukan, dapat disimpulkan bahwa sebagian besar fungsi-fungsi yang diuji pada antarmuka ini berhasil berjalan sesuai dengan harapan. Pengujian Load Image, Pengguna memilih gambar, dan Predict Image berhasil dilakukan dengan sukses, menunjukkan bahwa antarmuka dapat membuka jendela pemilihan file, memuat dan menampilkan gambar dengan benar, serta melakukan prediksi huruf berdasarkan gambar yang dimuat. Pengujian Update Timestamp in Frame juga berhasil, yang menunjukkan bahwa antarmuka mampu memperbarui dan menampilkan waktu dan timestamp dengan tepat. Namun, pengujian Save Image tidak memiliki hasil pengamatan yang disebutkan, sehingga tidak dapat dipastikan apakah pengujian tersebut berhasil atau belum dilakukan.

5.2.2 Pengujian White Box

Tabel 5.7 Tabel Pengujian White Box

No.	Algoritma	Perancangan Program
1	Vektor fitur gambar	<pre>def image_to_feature_vector(image): feature_vector = np.array(image).flatten() print("Feature vector:", feature_vector) return feature_vector</pre>
Hasil		

```

Run: cek tiap huruf x
C:\Users\hp\AppData\Local\Programs\Python\Python311\python.exe "C:/Users/hp/Downloads/Compressed/ANNAIPHABETv2/cek_tiap_huruf.py"
Feature vector: [255 255 255 ... 255 255 255]
<function image_to_feature_vector at 0x000001BE2C3CA160>
Time taken to predict: 0.0 seconds
Image saved with timestamp: 20230624_124339
Time taken to load image: 2.6780717372894287 seconds

```

Pembahasan

Print statement tersebut menunjukkan nilai dari vektor fitur gambar. Dalam contoh tersebut, vektor fitur gambar memiliki panjang yang cukup besar dan diwakili oleh serangkaian bilangan bulat. `[255 255 255 ... 255 255 255]` menunjukkan bahwa setiap elemen vektor memiliki nilai 255, yang mungkin mengindikasikan bahwa gambar yang diproses sepenuhnya terdiri dari piksel dengan intensitas warna maksimum (putih).

2	Standardize	<pre> def Standardize(X): epsilon = 1e-8 X = (X - np.mean(X, axis=1, keepdims=True)) / (np.std(X, axis=1, keepdims=True) + epsilon) print(X) return X </pre>
---	-------------	--

Hasil

```

Run: cek tiap huruf x
C:\Users\hp\AppData\Local\Programs\Python\Python311\python.exe "C:/Users/hp/Downloads/Compressed/ANNAIPHABETv2/cek_tiap_huruf.py"
Feature vector: [255 255 255 ... 255 255 255]
[[0.47156421 0.47156421 0.47156421 ... 0.47156421 0.47156421 0.47156421]]
Time taken to predict: 0.0 seconds
Image saved with timestamp: 20230624_130531
Time taken to load image: 2.188845157623291 seconds

```

Pembahasan

Output ini merupakan hasil dari proses standarisasi yang dilakukan pada matriks X. Pada fungsi Standardize, setiap elemen dalam matriks X dikurangi dengan rata-rata dari setiap baris matriks X, kemudian hasilnya dibagi dengan standar deviasi dari setiap baris matriks X ditambah dengan epsilon (nilai kecil untuk menghindari pembagian dengan nol). Proses ini dilakukan untuk setiap elemen dalam matriks X. Output ini merupakan matriks hasil standarisasi dari matriks X. Setiap elemen dalam matriks tersebut merupakan nilai standarisasi dari elemen yang sesuai dalam matriks X.

3	Softmax	<pre> def SoftmaxFunction(x): exp_x = np.exp(x) sum_exp_x = np.sum(exp_x, axis=0) softmax_output = exp_x / sum_exp_x print(softmax_output) return softmax_output </pre>
---	---------	---

Hasil

```
Run: cek tiap huruf x
C:\Users\hpl\AppData\Local\Programs\Python\Python311\python.exe "C:/Users/hp/Downloads/Compressed/ANNAIPHABETV2/cek tiap huruf.py"
[[3.24108745e-03]
 [1.04222486e-03]
 [1.09812977e-03]
 [2.35572113e-03]
 [1.20773330e-05]
```

Pembahasan

Setiap elemen dalam matriks tersebut merupakan probabilitas untuk setiap kelas atau kategori yang ada. Jumlah elemen dalam matriks tersebut sesuai dengan jumlah kategori atau kelas yang ada. Misalnya, jika terdapat 5 kategori, maka matriks tersebut akan memiliki 5 baris, di mana setiap baris berisi probabilitas untuk kategori tersebut. Probabilitas ini menunjukkan sejauh mana suatu input akan diklasifikasikan ke masing-masing kategori. Sebagai contoh, pada output di atas, kategori pertama memiliki probabilitas sekitar 0.0032, kategori kedua memiliki probabilitas sekitar 0.001, dan seterusnya. Probabilitas ini dapat diinterpretasikan sebagai tingkat kepercayaan model terhadap setiap kategori.

4

Sigmoid

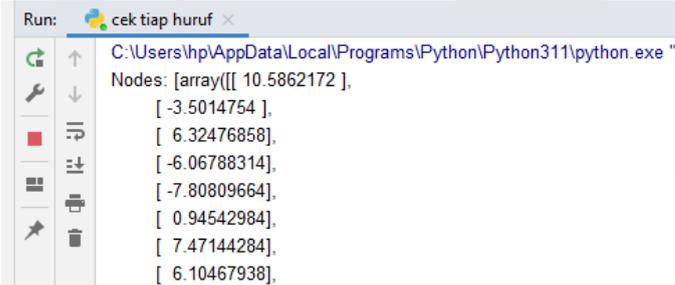
```
def Sigmoid(z):
    result = 1 / (1 + np.exp(-z))
    print("Output Sigmoid:", result)
    return result
```

Hasil

```
Run: cek tiap huruf x
C:\Users\hpl\AppData\Local\Programs\Python\Python311\python.exe "C:/Users/hp/Downloads/Compressed/ANNAIPHABETV2/cek tiap huruf.py"
Output Sigmoid: [[2.76001280e-02]
 [1.49217411e-01]
 [3.08163132e-01]
 [5.93294319e-01]
 [7.77140967e-01]
```

Pembahasan

Ini adalah hasil output sigmoid untuk sekumpulan nilai z yang masuk ke dalam fungsi Sigmoid. Setiap nilai z diproses secara terpisah dan diubah menjadi nilai sigmoid yang berkisar antara 0 dan 1. Perhatikan bahwa output sigmoid adalah sebuah matriks dengan bentuk (6, 1). Setiap elemen dalam matriks ini adalah hasil dari fungsi sigmoid yang diterapkan pada nilai z yang sesuai. Nilai sigmoid diperoleh dengan memasukkan nilai z ke dalam rumus sigmoid: $\text{sigmoid}(z) = 1 / (1 + \exp(-z))$. Setiap elemen dalam matriks output menunjukkan probabilitas atau tingkat aktivasi yang berkaitan dengan nilai z yang sesuai. Semakin tinggi nilai sigmoid, semakin besar probabilitas atau aktivasi yang terkait dengan nilai tersebut.

5	ForwardProp	<pre> def forwardProp(X, parameters, HiddenLayerSizes): h1, h2 = HiddenLayerSizes[:] if (h2 == 0): W1, b1, W2, b2 = parameters Z1 = W1 @ X + b1 A1 = Sigmoid(Z1) Z2 = W2 @ A1 + b2 A2 = SoftmaxFunction(Z2) nodes = np.array([Z1, A1, Z2, A2], dtype=object) else: W1, b1, W2, b2, W3, b3 = parameters Z1 = W1 @ X + b1 A1 = Sigmoid(Z1) Z2 = W2 @ A1 + b2 A2 = Sigmoid(Z2) Z3 = W3 @ A2 + b3 A3 = SoftmaxFunction(Z3) nodes = np.array([Z1, A1, Z2, A2, Z3, A3], dtype=object) print("Nodes:", nodes) return nodes </pre>
Hasil		
		
Pembahasan		

Output tersebut merupakan hasil dari print statement yang ada dalam fungsi `forwardProp` pada kode yang diberikan. Setiap baris pada output menunjukkan nilai dari setiap node dalam jaringan yang dihasilkan selama proses *forward propagation*.

Berikut adalah penjelasan mengenai output tersebut:

- Baris pertama (`array([[10.5862172],])`) menunjukkan nilai dari node pertama (`Z1`) dalam jaringan.
- Baris kedua (`[-3.5014754],`) menunjukkan nilai dari node kedua (`A1`) dalam jaringan.
- Baris ketiga (`[6.32476858],`) menunjukkan nilai dari node ketiga (`Z2`) dalam jaringan.
- Baris keempat (`[-6.06788314],`) menunjukkan nilai dari node keempat (`A2`) dalam jaringan.
- Baris kelima (`[-7.80809664],`) menunjukkan nilai dari node kelima (`Z3`) dalam jaringan.
- Baris keenam (`[0.94542984],`) menunjukkan nilai dari node keenam (`A3`) dalam jaringan.

Output ini memberikan informasi tentang nilai setiap node dalam jaringan setelah melewati *forward propagation*. Nilai-nilai ini digunakan dalam perhitungan selanjutnya dalam jaringan, seperti perhitungan gradien pada fungsi `gradient`.

