

Hasil eksperimen hyperparameter tuning menunjukkan perbedaan yang signifikan dalam performa model berdasarkan kombinasi hyperparameter yang diuji, termasuk jumlah hidden layers, jumlah neuron per layer, fungsi aktivasi, epoch, learning rate, dan batch size. Masing-masing konfigurasi memberikan wawasan yang penting tentang bagaimana memilih hyperparameter yang tepat untuk mencapai performa model yang optimal.

1. Pengaruh Fungsi Aktivasi (Sigmoid vs. ReLU)

- Pada model pertama, yang menggunakan fungsi aktivasi Sigmoid, terlihat bahwa meskipun jumlah epoch cukup tinggi (100) dan batch size besar (512), hasil train accuracy hanya mencapai 69.60%, sementara test accuracy lebih rendah lagi di 65.91%. Hal ini mengindikasikan bahwa model kesulitan untuk menemukan pola yang efektif pada data. Loss pada data pelatihan (0.6563) dan data uji (0.6617) juga relatif tinggi. Fungsi aktivasi Sigmoid mungkin kurang efektif dalam menangani masalah vanishing gradient dalam pelatihan model yang lebih dalam, sehingga pembaruan bobot menjadi lebih lambat dan konvergensi tidak optimal.
- Sebaliknya, model dengan ReLU activation menunjukkan hasil yang lebih menjanjikan. Pada model kedua (dengan 10 epoch dan learning rate 0.1), meskipun train accuracy sangat tinggi (79.92%), train loss tetap tinggi (2.3044), yang menunjukkan masalah exploding gradients karena learning rate yang terlalu besar. Hasil test accuracy 75.65% dan test loss 3.0174 mengindikasikan overfitting, di mana model terlalu menyesuaikan diri dengan data pelatihan namun tidak generalisasi dengan baik pada data uji.

2. Pengaruh Learning Rate

- Pengaturan learning rate sangat mempengaruhi kemampuan model untuk belajar dan konvergen. Pada model dengan learning rate 0.1 (model kedua hingga keempat), model menunjukkan peningkatan accuracy yang signifikan, baik pada data pelatihan maupun pengujian. Namun, pada model kedua dengan learning rate yang cukup besar (0.1), meskipun train accuracy tinggi, train loss sangat tinggi, menunjukkan bahwa pembaruan bobot yang terlalu besar menyebabkan model tidak stabil.
- Pada model ketiga dan keempat, dengan pengaturan learning rate 0.1 dan batch size yang lebih besar (128 dan 256), train accuracy tetap tinggi (86.89% dan 86.75%) dengan train loss yang lebih rendah (0.3549 dan 0.3367). Model ini mampu mempelajari fitur yang lebih baik tanpa mengalami masalah exploding gradient atau overfitting yang parah. Test accuracy mencapai 79.87% hingga 80.19%, dengan test loss yang lebih rendah (0.4794 dan 0.4609), yang menunjukkan bahwa model memiliki kemampuan generalisasi yang lebih baik pada data yang tidak terlihat selama pelatihan.

3. Pengaruh Batch Size

- Pengaruh batch size terhadap performa model juga terlihat jelas. Model dengan batch size lebih kecil (64) menunjukkan bahwa meskipun train accuracy lebih tinggi pada awalnya (79.92%), model mengalami kesulitan dalam konvergensi yang stabil, seperti yang tercermin dari train loss yang sangat tinggi (2.3044). Sebaliknya, ketika batch size ditingkatkan menjadi 128 atau 256, meskipun pelatihan berlangsung lebih lambat, model menunjukkan train loss yang jauh lebih rendah, yaitu 0.3549 dan 0.3367, serta test accuracy yang lebih baik (79.87% hingga 80.19%). Ini menunjukkan bahwa batch size yang lebih besar memberikan stabilitas lebih dalam proses pelatihan dan memperbaiki performa pada data uji.

4. Kesimpulan Umum

Berdasarkan hasil eksperimen, kombinasi ReLU activation, learning rate 0.1, dan batch size lebih besar (128 atau 256) menghasilkan model yang lebih stabil dan memiliki performa lebih baik. Model dengan Sigmoid activation dan learning rate sangat kecil (0.0001) cenderung mengalami masalah dalam konvergensi dan memiliki performa yang kurang optimal pada kedua set data (train dan test). Selain itu, batch size yang lebih besar (128 dan 256) membantu dalam mengurangi fluktuasi besar dalam pembaruan bobot dan mempercepat proses konvergensi tanpa mengorbankan stabilitas model. Oleh karena itu, eksperimen ini mengindikasikan bahwa pilihan fungsi aktivasi ReLU, learning rate yang lebih besar namun tidak terlalu ekstrim, serta batch size yang lebih besar adalah kombinasi yang optimal untuk model MLP dalam tugas klasifikasi ini.

Penting untuk mencatat bahwa meskipun model dengan ReLU dan pengaturan hyperparameter tertentu memberikan hasil yang baik, eksperimen lebih lanjut dengan cross-validation dan pengujian pada data yang lebih beragam akan lebih memperkuat hasil ini dan membantu memastikan model memiliki kemampuan generalisasi yang lebih baik.