

1. Arsitektur CNN dengan X lapisan konvolusi menghasilkan akurasi training 98% tetapi akurasi validasi 62%. Jelaskan fenomena vanishing gradient yang mungkin terjadi pada lapisan awal, dan bagaimana cara memitigasinya! Mengapa penambahan Batch Normalization setelah lapisan konvolusi ke-Y justru memperburuk generalisasi, serta strategi alternatif untuk menstabilkan pembelajaran?
2. Ketika melatih CNN dari nol, loss training stagnan di nilai tinggi setelah XXX(3 digit epoch) epoch. Identifikasi tiga penyebab potensial terkait laju pembelajaran (learning rate), inisialisasi berat, atau kompleksitas model! Mengapa penggunaan Cyclic Learning Rate dapat membantu model keluar dari local minima, dan bagaimana momentum pada optimizer SGD memengaruhi konvergensi?
3. Pada klasifikasi spesies ikan menggunakan CNN, penggunaan fungsi aktivasi ReLU tidak menunjukkan peningkatan akurasi setelah 50 epoch, meskipun learning rate telah dioptimasi. Jelaskan fenomena dying ReLU yang mungkin terjadi dan bagaimana hal ini mengganggu aliran gradien selama backpropagation!
4. Pada pelatihan CNN untuk klasifikasi XX spesies ikan, grafik AUC-ROC menunjukkan satu kelas (Spesies X) stagnan di 0.55 sementara kelas lain mencapai >0.85 setelah YYY epoch. Analisis mengapa class-weighted loss function gagal meningkatkan kinerja Spesies X, dan identifikasi tiga faktor penyebab potensial terkait karakteristik data dan arsitektur model!
5. Pada arsitektur CNN untuk klasifikasi ikan, peningkatan kompleksitas model justru menyebabkan penurunan akurasi validasi dari 85% ke 65%, meskipun akurasi training mencapai 98%. Jelaskan fenomena overfitting yang terjadi, dan mengapa penambahan kapasitas model tidak selalu meningkatkan generalisasi! Identifikasi 3 kesalahan desain arsitektur yang memicu degradasi performa

## **JAWABAN**

1. Fenomena Vanishing Gradient: Pada lapisan awal, gradien menjadi sangat kecil selama backpropagation, menyebabkan pembaruan berat yang minimal dan memperlambat pembelajaran. Ini terjadi karena aktivasi seperti sigmoid/tanh yang mengecilkan gradien. Gunakan fungsi aktivasi ReLU/LeakyReLU, inisialisasi berat (He/Glorot), atau arsitektur residual (ResNet). Batch Normalization (BN): BN bisa memperburuk generalisasi jika diterapkan berlebihan atau dengan batch size kecil, karena estimasi statistik batch menjadi tidak stabil. Alternatifnya gunakan Layer Normalization atau Group Normalization untuk data dengan batch size kecil, atau tambahkan Dropout untuk regularisasi.
2. Learning rate terlalu tinggi (gradien berosilasi) atau terlalu rendah (perlambatan konvergensi). Inisialisasi berat tidak optimal (misalnya, semua berat awal nol), Model terlalu sederhana untuk data kompleks (underfitting). Cyclic Learning Rate dapat membantu keluar dari local minima dengan variasi learning rate secara periodik, memungkinkan eksplorasi landscape loss yang lebih luas. Momentum pada SGD

mempercepat konvergensi dengan mengakumulasi gradien sebelumnya, mengurangi osilasi dan membantu melewati local minima.

3. Neuron ReLU menjadi tidak aktif (output = 0 untuk semua input) karena gradien nol selama backpropagation, menghentikan pembelajaran, dampaknya adalah aliran gradien terputus untuk neuron tersebut, mengurangi kapasitas model.
4. Class imbalance ekstrem (contoh: Spesies X memiliki sampel sangat sedikit), fitur Spesies X kurang informatif atau mirip dengan kelas lain, Model terlalu kompleks sehingga overfit ke kelas mayoritas.
5. Model menghafal data training tetapi gagal generalisasi ke data validasi, ditandai dengan akurasi training tinggi tetapi validasi rendah. Penyebab Penurunan Generalisasi adalah Kapasitas model berlebihan (terlalu banyak lapisan/parameter), Kurangnya regularisasi (Dropout, L2 regularization), Data training tidak beragam (augmentasi kurang).

Kesalahan Desain: Lapisan konvolusi terlalu dalam tanpa skip connection (ResNet), Tidak ada mekanisme Dropout/Batch Normalization, Kernel size atau stride tidak sesuai dengan ukuran gambar input.