

Analisi Klasifikasi Dataset (CNN)

Dhanendra Nivadirrokhman (1103220113)

1. Vanishing Gradient dan Pengaruh Batch Normalization terhadap Generalisasi

Pada eksperimen arsitektur CNN dengan beberapa lapisan konvolusi, diperoleh akurasi training sangat tinggi (98%) namun akurasi validasinya rendah (62%). Hal ini mencerminkan adanya gejala *vanishing gradient*, yaitu kondisi di mana gradien di lapisan awal menjadi terlalu kecil sehingga bobot tidak terupdate secara efektif. Biasanya ini muncul karena pemakaian fungsi aktivasi seperti sigmoid atau tanh secara berlapis, yang secara matematis membuat turunan (gradien) makin mengecil.

Untuk memitigasi hal ini, pendekatan yang bisa digunakan adalah mengganti fungsi aktivasi menjadi ReLU atau turunannya seperti LeakyReLU dan ELU, karena mereka tidak mengalami saturasi negatif. Selain itu, strategi seperti skip connections (misalnya pada ResNet) juga bisa menjaga aliran gradien tetap stabil hingga ke lapisan awal. Inisialisasi bobot juga penting, seperti He Initialization yang dirancang untuk jaringan ReLU.

2. Stagnasi Loss dan Strategi dengan Cyclic Learning Rate (CLR)

Dalam kasus lain, ketika CNN dilatih dari awal selama ratusan epoch namun loss tidak juga turun, ada kemungkinan disebabkan oleh tiga faktor utama:

1. **Learning rate terlalu kecil**, sehingga bobot tidak berubah signifikan di setiap iterasi.
2. **Inisialisasi bobot yang buruk**, yang menyebabkan aktivasi masuk zona mati (ReLU) atau saturasi (sigmoid).
3. **Model terlalu kompleks**, membuat ruang parameter terlalu besar untuk dieksplorasi oleh dataset yang kecil.

Penggunaan Cyclic Learning Rate (CLR) menjadi solusi yang cukup efektif. CLR bekerja dengan mengubah learning rate secara periodik naik-turun. Mekanisme ini membantu model keluar dari jebakan local minima atau saddle point dan menjelajahi area lain dalam parameter space. Dengan begitu, model punya peluang lebih besar menemukan solusi yang lebih optimal.

3. Dying ReLU dalam CNN

Fenomena lain yang cukup umum adalah *dying ReLU*, yaitu kondisi di mana sejumlah neuron di CNN tidak aktif secara permanen. Ini biasanya terjadi ketika neuron menerima input negatif terus-menerus, menyebabkan output dan gradiennya nol. Jika ini terjadi dalam jumlah besar, maka kapasitas representasi model berkurang drastis.

Untuk mengatasi hal ini, bisa digunakan varian ReLU seperti LeakyReLU, ELU, atau PReLU yang masih memberikan output meskipun input-nya negatif. Selain itu, penting juga

memastikan bahwa inisialisasi bobot tidak mendorong terlalu banyak neuron masuk ke area negatif sejak awal. Penggunaan BatchNorm juga bisa menstabilkan input ke ReLU agar tetap berada dalam jangkauan aktif.

4. Kinerja AUC-ROC Rendah pada Salah Satu Kelas

Pada klasifikasi spesies ikan, ditemukan satu kelas (misalnya Spesies X) dengan AUC-ROC yang stagnan di 0.55, jauh di bawah kelas lain yang sudah mencapai >0.85 . Meskipun sudah digunakan class-weighted loss function, performa kelas tersebut tetap rendah.

Hal ini kemungkinan besar karena class-weight hanya menambah penalti saat model salah prediksi, tapi tidak serta-merta membuat model lebih memahami fitur penting dari kelas tersebut. Beberapa penyebab utamanya adalah:

1. **Variasi dalam kelas terlalu tinggi**, atau citra buram dan mirip dengan spesies lain.
2. **Jumlah sampel terlalu sedikit**, membuat model kurang belajar dari kelas tersebut.
3. **Model tidak mampu menangkap detail lokal**, seperti warna sirip yang unik.

Strategi yang bisa digunakan antara lain mengganti loss function menjadi focal loss, karena lebih fokus ke contoh yang susah diklasifikasikan. Selain itu, attention mechanism atau arsitektur seperti EfficientNet bisa meningkatkan kemampuan deteksi fitur kecil. Augmentasi data juga penting agar model melihat lebih banyak variasi dari kelas minoritas.

5. Overfitting karena Model Terlalu Kompleks

Pada eksperimen lain, peningkatan kompleksitas model CNN justru menurunkan akurasi validasi dari 85% ke 65%, padahal akurasi training tetap tinggi di 98%. Ini adalah kasus klasik *overfitting*, di mana model terlalu besar dan malah menghafal data alih-alih belajar pola umum.

Masalah utama dalam desain arsitektur ini meliputi:

1. **Terlalu banyak neuron di fully connected layer**, menambah parameter tanpa nilai tambah.
2. **Tidak adanya regularisasi seperti dropout atau L2**, membuat model terlalu bebas mempelajari noise.
3. **Terlalu dalam tanpa skip connection atau normalisasi**, yang memperparah masalah vanishing gradient.

Solusinya adalah menyederhanakan arsitektur, tambahkan regulasi eksplisit, dan sesuaikan kompleksitas model dengan ukuran dataset. Dalam deep learning, lebih banyak tidak selalu lebih baik.

Kesimpulan

Dari berbagai eksperimen dan fenomena yang dialami selama training CNN, terlihat jelas bahwa pemahaman mendalam tentang arsitektur, loss function, optimisasi, dan karakteristik data sangat penting. Setiap komponen saling terhubung dan keputusan desain harus selalu disesuaikan dengan kondisi dataset yang digunakan. Pendekatan adaptif dan observasi metrik seperti AUC, loss, dan validasi akurasi wajib dilakukan secara berkala agar training tetap terkendali dan menghasilkan model yang benar-benar generalisasi.