

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?

Jawab:

Masalah Vanishing Gradient dan Penggunaan Batch Normalization

Fenomena Vanishing Gradient mengacu pada situasi ketika gradien fungsi kerugian yang diteruskan melalui jaringan menjadi sangat kecil, khususnya pada lapisan awal CNN. Akibatnya, pembaruan bobot pada lapisan tersebut menjadi tidak efektif, menghambat proses pembelajaran.

Faktor penyebab:

- Jumlah lapisan konvolusi yang terlalu banyak (nilai X besar)
- Penggunaan fungsi aktivasi tertentu (sigmoid/tanh)
- Tidak adanya mekanisme stabilisasi seperti koneksi pintas

Cara mengatasi:

- Menerapkan fungsi **ReLU** atau **variasinya** (LeakyReLU/ELU)
- Mengimplementasikan **residual connections** (contohnya ResNet)
- Menggunakan **inisialisasi He** untuk bobot jaringan

Dampak negatif BatchNorm setelah lapisan ke-Y: Meskipun Batch Normalization mempercepat dan menstabilkan pelatihan, penerapannya setelah lapisan tertentu dapat:

- Menyebabkan distorsi distribusi fitur, terutama pada mini-batch kecil
- Mengakibatkan **model overfit terhadap noise data** karena terlalu bergantung pada statistik batch

Alternatif stabilisasi:

- Menerapkan **Layer Normalization** (terutama untuk batch kecil)
- Menggunakan **Dropout** sebagai metode regularisasi
- Kombinasi dengan **koneksi pintas** atau teknik **augmentasi data** yang intensif

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?

Jawab:

Permasalahan Loss Stagnan pada Nilai Tinggi Setelah Lebih dari 100 Epoch

Tiga kemungkinan penyebab:

1. **Learning rate terlalu rendah** - Menghasilkan gradien kecil dan pembaruan bobot lambat
2. **Inisialisasi bobot tidak optimal** - Bobot awal menyebabkan saturasi output pada fungsi aktivasi
3. **Model terlalu kompleks atau dalam** - Sulit dioptimalkan dari kondisi awal

Manfaat Cyclic Learning Rate (CLR): CLR yang bervariasi secara periodik membantu:

- Menghindari terjebak di **local minima** atau titik sadel
- Menjelajahi ruang pencarian loss yang lebih luas

Fungsi Momentum dalam SGD:

- Menyimpan informasi arah gradien sebelumnya untuk percepatan konvergensi
- Membantu melewati area berlekuk dan datar dalam landscape loss
- Kombinasi momentum dengan CLR efektif mencegah stagnasi

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!

Jawab:

Permasalahan ReLU dan Dying ReLU

Dying ReLU:

- Kondisi dimana neuron dengan ReLU selalu menghasilkan output 0 (karena input negatif)
- Gradien bernilai 0 sehingga bobot tidak diperbarui, menyebabkan neuron "mati"

Dampak pada backpropagation:

- Lapisan atas tidak mentransmisikan gradien ke neuron ReLU yang mati
- Banyak neuron menjadi tidak aktif dalam proses pembelajaran

Solusi:

- Mengganti ReLU dengan **LeakyReLU, PReLU, atau ELU**
- Menerapkan **inisialisasi yang lebih baik** seperti inisialisasi He
- Menggabungkan dengan **BatchNorm** untuk menjaga distribusi input

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!

Jawab:

Stagnasi AUC-ROC untuk Spesies X

Mengapa loss berbobot kelas tidak efektif:

- Pembobotan kelas hanya meningkatkan penalti untuk kelas minoritas, namun tidak menyelesaikan **ketimpangan representasi fitur**

Tiga kemungkinan penyebab:

1. **Data tidak terdistribusi secara representatif** - Gambar Spesies X mungkin terlalu sedikit atau berkualitas rendah
2. **Karakteristik visual kelas X terlalu mirip dengan kelas lain** - CNN sulit membedakannya
3. **Arsitektur model kurang fokus pada fitur spesifik/lokal** - Misalnya tidak ada mekanisme perhatian atau receptive field terlalu besar

Solusi alternatif:

- **Oversampling atau augmentasi khusus untuk Spesies X**
- Penerapan **focal loss** yang fokus pada kesalahan prediksi
- Penambahan modul **attention** (CBAM, SE block) untuk memperkuat fitur kelas minoritas

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

Jawab:

Overfitting Akibat Kompleksitas Model

Gejala:

- Model kompleks (banyak parameter) mencapai performa sangat baik pada data training (98%) - mengindikasikan memorisasi
- Performa buruk pada generalisasi data baru - akurasi validasi rendah (65%)

Mengapa peningkatan kapasitas tidak selalu bermanfaat:

- Jumlah data tidak memadai, sehingga model mempelajari noise
- Ketidakesesuaian arsitektur dengan karakteristik data (misalnya receptive field tidak optimal)
- Ketidadaan regularisasi menyebabkan overfitting

Tiga kesalahan desain umum:

1. **Jumlah filter terlalu banyak di awal jaringan** - terlalu banyak parameter sejak awal

2. **Tidak menerapkan dropout atau augmentasi data** - rentan terhadap overfitting
3. **Jaringan terlalu dalam tanpa residual connection** - berpotensi menyebabkan vanishing gradient dan kesulitan optimisasi

Strategi mitigasi:

- Menerapkan teknik **dropout**, **early stopping**, dan **augmentasi data**
- Menggunakan model lebih sederhana atau CNN pre-trained dengan fine-tuning
- Menerapkan **transfer learning** untuk dataset kecil