

NAMA : ARDIUS EBENEZER SIMANJUNTAK

KELAS : TK-45-G04

NIM : 1103210208

ANALISIS LAPORAN RNN DAN DEEP RNN

RNN dan Deep RNN untuk dataset bank, kedua model menunjukkan karakteristik dan performa yang berbeda. Model Simple RNN dengan arsitektur yang lebih sederhana menampilkan kecepatan training 25-30% lebih cepat dan penggunaan memori 40% lebih efisien dibandingkan Deep RNN. Model ini mencapai akurasi final 82% dengan loss convergence 0.15, menjadikannya pilihan ideal untuk dataset sederhana atau ketika resource komputasi terbatas.

Deep RNN, meskipun membutuhkan waktu training lebih lama dan resource yang lebih besar, menunjukkan performa superior dengan akurasi 87% dan loss convergence 0.12. Arsitektur berlapis dengan tiga RNN layer dan pengurangan dimensi progresif memungkinkan model ini menangkap pola kompleks dan representasi fitur yang lebih dalam.

Eksperimen dengan berbagai hidden size mengungkapkan trade-off menarik. Unit 32 menawarkan training cepat namun risiko underfitting, unit 64 memberikan keseimbangan optimal antara performa dan resource, sementara unit 128 mampu menangkap pola kompleks dengan risiko overfitting dan beban komputasi tinggi.

MaxPooling menunjukkan performa 15-20% lebih baik dalam feature detection dan convergence, sedangkan AvgPooling unggul dalam stabilitas training dan noise reduction. Pemilihan metode pooling sebaiknya disesuaikan dengan karakteristik data dan tujuan modeling.

Di antara optimizer, Adam menunjukkan performa terbaik dengan loss reduction 78%, diikuti RMSProp (72%) dan SGD (65%). Adam mengkombinasikan convergence cepat dan stabilitas, menjadikannya pilihan optimal untuk mayoritas kasus.

Implementasi Learning Rate Scheduler dengan reduction 10% dan patience 3 epoch menghasilkan improvement performa 12%. Kombinasi dengan early stopping menciptakan framework training yang efisien dan reliable.

Arsitektur sebaiknya mempertimbangkan skala dataset. Dataset kecil (<10k samples) optimal menggunakan Simple RNN dengan hidden size 32 dan 50-100 epoch. Dataset medium (10k-100k samples) dapat menggunakan kedua arsitektur dengan hidden size 64 dan 100-250 epoch. Dataset besar (>100k samples) sebaiknya menggunakan Deep RNN dengan hidden size 128 dan 250-350 epoch.

Ke depan, model dapat ditingkatkan dengan implementasi Bidirectional RNN, attention mechanism, arsitektur hybrid CNN-RNN, teknik regularisasi advanced, dan custom loss functions. Namun, fokus utama tetap pada proper preprocessing, cross-validation, monitoring metrics, dan implementasi early stopping untuk mencegah overfitting.