

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan mengenai klasifikasi mood musik menggunakan metode Convolutional Neural Network (CNN), dapat ditarik beberapa kesimpulan sebagai berikut.

Pertama, sistem klasifikasi mood musik berhasil dibangun menggunakan arsitektur CNN dengan memanfaatkan fitur audio MFCC dan Mel-Spectrogram sebagai representasi input. Kedua fitur tersebut terbukti mampu merepresentasikan karakteristik audio dari masing-masing kelas mood secara distinktif, di mana setiap kelas mood memiliki pola MFCC dan Mel-Spectrogram yang dapat dibedakan secara visual maupun komputasional.

Kedua, hasil perbandingan antara Program 1 (baseline model) dan Program 2 (model dengan penanganan imbalance) menunjukkan bahwa penggunaan *balanced class weight* melalui fungsi `compute_class_weight` pada Program 2 terbukti meningkatkan kemampuan generalisasi model. Meskipun Program 1 mencatat Best Validation Accuracy yang lebih tinggi sebesar 82,12%, Program 2 mengungguli Program 1 pada data testing dengan Test Accuracy sebesar 79,93% dan Test F1-Score sebesar 79,45%, dibandingkan Program 1 yang hanya mencapai 79,68% pada kedua metrik tersebut. Hal ini mengindikasikan bahwa Program 1 cenderung mengalami *overfitting*, sedangkan Program 2 memiliki kemampuan generalisasi yang lebih baik terhadap data baru.

Ketiga, hasil evaluasi confusion matrix menunjukkan bahwa kelas Romantic memiliki akurasi per kelas tertinggi sebesar 87%, sedangkan kelas Happy memiliki akurasi terendah sebesar 73%. Pola kesalahan klasifikasi yang paling dominan terjadi antar kelas yang memiliki kedekatan karakteristik audio, seperti pasangan Happy-Dramatic dan Sad-Romantic, yang mengindikasikan adanya tumpang tindih (*overlapping*) fitur audio antar kelas-kelas tersebut.

Keempat, model terbaik (Program 2) berhasil di-deploy menggunakan

framework Flask dalam bentuk REST API yang dilengkapi antarmuka web interaktif bernama MoodCNN Audio Mood API. Sistem ini memungkinkan pengguna untuk melakukan prediksi mood musik secara langsung melalui browser tanpa memerlukan pengetahuan teknis pemrograman, sehingga membuktikan bahwa model yang dibangun tidak hanya memiliki performa yang baik secara teoritis, tetapi juga dapat diimplementasikan dalam aplikasi nyata yang fungsional.

5.2 Saran

Berdasarkan hasil penelitian dan keterbatasan yang ditemukan selama proses pengembangan, terdapat beberapa saran yang dapat dijadikan acuan untuk penelitian maupun pengembangan sistem di masa mendatang.

Pertama, perluasan dan diversifikasi dataset perlu dilakukan untuk meningkatkan performa model secara keseluruhan. Dataset yang lebih besar dan beragam, mencakup berbagai genre musik, budaya, dan instrumen, diharapkan dapat mengurangi bias model dan meningkatkan kemampuan generalisasi terhadap jenis musik yang belum pernah dilihat sebelumnya.

Kedua, eksplorasi fitur audio tambahan seperti Chroma Features, Spectral Contrast, Tonnetz, maupun Zero-Crossing Rate dapat dipertimbangkan untuk dikombinasikan dengan MFCC dan Mel-Spectrogram yang sudah digunakan. Penambahan fitur-fitur tersebut diharapkan dapat mengurangi tingkat *overlapping* antar kelas mood yang berdekatan, khususnya pada pasangan kelas Happy-Dramatic dan Sad-Romantic.

Ketiga, eksplorasi arsitektur model yang lebih kompleks seperti ResNet, EfficientNet, atau arsitektur berbasis *attention mechanism* seperti Transformer dapat dilakukan untuk meningkatkan akurasi klasifikasi. Selain itu, pendekatan *transfer learning* menggunakan model yang telah dilatih pada dataset audio berskala besar juga dapat menjadi alternatif yang menjanjikan.

Keempat, peningkatan teknik penanganan ketidakseimbangan kelas perlu dikaji lebih lanjut, misalnya dengan menggunakan teknik *oversampling* seperti

SMOTE (*Synthetic Minority Over-sampling Technique*) atau *data augmentation* yang lebih agresif pada kelas minoritas, sehingga distribusi data pelatihan menjadi lebih seimbang dan model dapat belajar secara lebih merata dari seluruh kelas.

Kelima, pengembangan sistem deployment dapat ditingkatkan dengan mengimplementasikan proses ekstraksi fitur secara *end-to-end* langsung dari file audio mentah pada sisi server, sehingga pengguna tidak perlu melakukan preprocessing secara manual sebelum mengirimkan data ke API. Selain itu, pengembangan aplikasi mobile berbasis Android maupun iOS dapat menjadi langkah lanjutan untuk memperluas jangkauan pengguna sistem ini.

