

BAB V PENUTUP

5.1 Kesimpulan

Berdasarkan hasil penelitian, implementasi, dan pengujian komparatif yang telah dilakukan mengenai prediksi tingkat kemenangan pada *Game Magic Chess Go Go* menggunakan algoritma *Machine Learning*, dapat ditarik 3 (Tiga) kesimpulan utama yang menjawab rumusan masalah secara komprehensif sebagai berikut:

1) Evaluasi Kuantitatif Performa Algoritma (Skenario 30x Iterasi).

Berdasarkan pengujian klasifikasi kuantitatif menggunakan skenario 30 kali iterasi pengujian (*30-times running*), algoritma *Random Forest* terbukti secara empiris lebih stabil dan memberikan performa rata-rata yang optimal mengungguli *XGBoost*. *Random Forest* mencatatkan rata-rata Akurasi $53,27\% \pm 3,95\%$ dan Presisi $57,12\% \pm 6,00\%$. Sementara itu, *XGBoost* mencatatkan rata-rata Akurasi $53,14\% \pm 6,26\%$ dan Presisi $53,60\% \pm 6,68\%$. Keunggulan *Random Forest* yang memiliki simpangan baku (Standar Deviasi) lebih kecil membuktikan bahwa mekanisme *bagging* lebih tahan banting dan stabil dalam menangani tingginya variabilitas elemen acak (*RNG*) dalam ekosistem *auto battler* dibandingkan algoritma *boosting*, sehingga rekomendasi prediksinya jauh lebih dapat dipercaya.

2) Identifikasi Faktor Strategi Dominan (*Feature Importance*).

Melalui ekstraksi bobot *Feature Importance* dari pemodelan *Random Forest* terbaik, terbukti secara kuantitatif bahwa faktor yang memberikan kontribusi paling dominan terhadap probabilitas kemenangan di *Season 4* adalah pemilihan karakter utama, yaitu atribut *commander_name* (16,99%) dan *gogo_comander* (9,80%). Temuan empiris ini menjawab rumusan masalah sekaligus mematahkan asumsi awal bahwa komposisi *synergy* adalah penentu mutlak segalanya. Model secara objektif membuktikan bahwa *skill* fundamental bawaan dari *commander* memberikan basis pertahanan yang jauh lebih krusial di awal permainan, sedangkan *synergy* seperti *Starwing* (5,59%) dan *Defender* (5,59%) hanya bertindak sebagai elemen pelengkap strategis.

3) Implementasi Sistem Prediksi (*Deployment*).

Tahap akhir penelitian telah berhasil merancang dan mengimplementasikan model *Random Forest* berkinerja terbaik tersebut ke dalam sebuah purwarupa (*Prototype*) aplikasi antarmuka interaktif berbasis web menggunakan *framework Streamlit*. Sistem fungsional ini memungkinkan pengguna untuk memasukkan parameter strategi secara mandiri dan langsung mendapatkan keluaran berupa probabilitas prediksi kemenangan. Capaian ini sukses menerjemahkan analisis komputasional menjadi sebuah perangkat lunak pendukung keputusan praktis.

5.2 Saran

Meskipun sistem prediksi telah berhasil dibangun, penelitian ini masih memiliki sejumlah keterbatasan yang diakibatkan oleh tingginya elemen *Random Number Generator (RNG)* dalam permainan. Untuk pengembangan penelitian selanjutnya, peneliti memberikan saran sebagai berikut:

1) Eskalasi Volume Dataset melalui Ekstraksi *Computer Vision* atau *Crowdsourcing*.

Model *Machine Learning* yang dibangun saat ini dilatih menggunakan ratusan sampel data yang dikumpulkan secara semi-manual akibat tidak adanya repositori data publik maupun *API* (Antarmuka Pemrograman Aplikasi) terbuka dari pihak pengembang *game*. Untuk menstabilkan variansi model dan mencapai target ribuan entri data, peneliti selanjutnya disarankan untuk membangun *tool* ekstraksi data otomatis berbasis *Computer Vision* atau *Optical Character Recognition (OCR)* yang mampu membaca tangkapan layar hasil pertandingan (*Post-Match Screen*). Alternatif lainnya adalah menggunakan pendekatan *crowdsourcing* dengan melibatkan komunitas pemain *e-sports* untuk berkolaborasi mengumpulkan data historis secara masif.

2) Penambahan Variabel Dinamis dan Data *Time-Series*.

Penelitian ini berfokus pada analisis data statis pasca-pertandingan (*Post-Match Summary*). Guna meningkatkan tingkat akurasi prediksi melampaui angka 56,73%, sangat disarankan untuk mengintegrasikan data per ronde (*Round-By-Round*) yang bersifat *time-series*. Selain itu, penambahan fitur mikro di dalam permainan (*In-Game Features*) seperti tata letak posisi *hero* (*Positioning*),

distribusi *item* perlengkapan, serta sisa *health points (HP)* mutlak diperlukan untuk menangkap dinamika perubahan strategi di tengah pertandingan (*Mid-Game Transition*).

3) Eksplorasi Pendekatan *Deep Learning*.

Mengingat tingginya kompleksitas interaksi antar-variabel dan sifat *non-linear* dari data permainan ini, penerapan arsitektur *Deep Learning* seperti *Artificial Neural Networks (ANN)* sangat direkomendasikan. Jika data yang digunakan beralih ke format *Time-Series* (per ronde), algoritma *Long Short-Term Memory (LSTM)* dapat menjadi alternatif yang lebih superior dibandingkan *Machine Learning* konvensional dalam mengenali pola adaptasi taktik pemain seiring berjalannya durasi pertandingan.

