

Klasifikasi Tipe Pokemon Berdasarkan Statistik Tempur Menggunakan Algoritma Random Forest

Nurul Hidayanah^{1,*}, Yohanes Simarmata², Icha Winadya Permadani³, Nugroho Kumala Destianto⁴,
Henis Sulistiani⁵

^{1,2,3,4,5}Fakultas Teknik dan Ilmu Komputer, Megister Ilmu Komputer, Universitas Teknokrat Indonesia, Bandar Lampung, Indonesia

Email: ^{1,*}nurulhidayanahS2@teknokrat.ac.id, ²yohannessimarmata@teknokrat.ac.id, ³IchaWinadyaPermadaniS2@teknokrat.ac.id,

⁴NugrohokumaladestiantoS2@teknokrat.ac.id, ⁵henisulistiani@teknokrat.ac.id

Email Penulis Korespondensi: nurulhidayanahS2@teknokrat.ac.id

Abstrak—Penelitian ini bertujuan untuk mengklasifikasikan tipe Pokemon berdasarkan statistik tempur menggunakan algoritma Random Forest. Dataset yang digunakan berasal dari file pokemon bw.csv yang berisi informasi seperti nomor Pokédex, nama, tipe, kemampuan, serta nilai statistik tempur (HP, Att, Def, S.Att, S.Def, Spd). Proses data preprocessing dilakukan untuk membersihkan dan mempersiapkan data, termasuk ekstraksi tipe utama, label encoding, feature selection, dan standarisasi fitur. Setelah itu, dataset dibagi menjadi data latih dan data uji dengan rasio 80:20. Model klasifikasi dibangun menggunakan Random Forest dengan 100 pohon keputusan dan dievaluasi menggunakan metrik accuracy, classification report, confusion matrix, serta ROC Curve multikelas. Hasil menunjukkan bahwa model mampu mencapai akurasi sebesar 64.8% , dengan performa terbaik pada kelas 'rock', 'steel', dan 'dragon', sedangkan kelas 'flying' dan 'ghost' masih sulit diklasifikasikan secara akurat. Confusion matrix menunjukkan bahwa beberapa tipe memiliki kesalahan klasifikasi yang cukup signifikan, seperti 'ground' yang sering diprediksi sebagai 'grass' dan 'rock' yang sering salah diklasifikasikan sebagai 'steel'. Evaluasi ROC Curve juga membuktikan bahwa sebagian besar kelas memiliki AUC di atas 0.80, menunjukkan kemampuan model dalam membedakan antar kelas. Dengan pendekatan ini, penelitian ini memberikan analisis awal mengenai potensi prediksi tipe Pokémon berbasis statistik tempur, yang dapat dikembangkan lebih lanjut melalui penanganan ketidakseimbangan kelas atau penggunaan teknik ensemble lainnya.

Kata Kunci: *Random Forest; Statistik Tempur; Machine Learning; Data Preprocessing; Confusion Matrix*

Abstract—This study aims to classify Pokémon types based on combat statistics using the Random Forest algorithm. The dataset used comes from the pokemon bw.csv file which contains information such as Pokédex number, name, type, ability, and combat statistics values (HP, Att, Def, S.Att, S.Def, Spd). Data preprocessing was carried out to clean and prepare the data, including primary type extraction, label encoding, feature selection, and feature standardization. After that, the dataset was divided into training data and test data with a ratio of 80:20. A classification model was built using Random Forest with 100 decision trees and evaluated using accuracy metrics, classification report, confusion matrix, and multiclass ROC Curve. The results showed that the model was able to achieve an accuracy of 64.8%, with the best performance in the 'rock', 'steel', and 'dragon' classes, while the 'flying' and 'ghost' classes were still difficult to classify accurately. The confusion matrix shows that several types have significant misclassifications, such as 'ground' being frequently predicted as 'grass' and 'rock' being frequently misclassified as 'steel'. The ROC Curve evaluation also shows that most classes have AUCs above 0.80, demonstrating the model's ability to differentiate between classes. With this approach, this study provides a preliminary analysis of the potential of Pokémon type prediction based on combat statistics, which can be further developed by addressing class imbalance or using other ensemble techniques.

Keywords: *Random Forest; Combat Statistics; Machine Learning; Data Preprocessing; Confusion Matrix*

1. PENDAHULUAN

Pokemon merupakan salah satu franchise video game populer yang tidak hanya menawarkan pengalaman bermain yang menyenangkan, tetapi juga melibatkan aspek strategi yang kompleks dalam pembentukan tim pertarungan. Salah satu elemen krusial dalam permainan ini adalah tipe Pokémon , yang mempengaruhi efektivitas serangan, kelemahan, serta ketahanan terhadap serangan lawan. Selain itu, statistik tempur seperti HP, Attack, Defense, Sp. Atk, Sp. Def, dan Speed menjadi parameter utama untuk menilai potensi Pokémon di medan laga dan membedakan karakteristik tiap spesies secara numerik [1].

Dengan adanya pola statistik yang bervariasi pada setiap tipe Pokémon, analisis data dapat digunakan untuk mengidentifikasi karakteristik unik dari masing-masing tipe. Hal ini membuka peluang besar untuk pemanfaatan teknik machine learning , salah satunya algoritma Random Forest , sebagai metode untuk mengklasifikasikan tipe Pokémon berdasarkan data statistiknya secara otomatis dan akurat [2],[3]. Random Forest merupakan pembelajaran ansambel algoritma yang bekerja dengan menggabungkan beberapa pohon keputusan (decision tree) untuk menghasilkan prediksi yang lebih stabil dan andal dibandingkan model tunggal [4],[5].

Penelitian sebelumnya telah menunjukkan bahwa Random Forest mampu memberikan performa tinggi dalam klasifikasi dataset dengan dimensi tinggi serta memiliki ketahanan terhadap overfitting . Hal ini menjadikannya pilihan yang sangat sesuai untuk menangani dataset dengan keragaman fitur seperti statistik Pokémon. Selain itu, penggunaan teknik optimasi seperti Grid Search Cross-Validation dan penanganan kesinkronan data menggunakan SMOTE (Synthetic Minority Over-sampling Technique) juga terbukti meningkatkan akurasi model secara signifikan [6]. Misalnya, dalam penelitian tentang klasifikasi Pokémon legendaris, algoritma Random Forest dikembangkan menjadi varian yang disebut SF-Random Forest , yang dirancang untuk menangani heterogenitas dan kompleksitas data dengan lebih baik. Dengan pendekatan SMOTE, ke perpaduan antara jumlah Pokémon legendaris dan non-legendaris berhasil diatasi, sehingga model mampu belajar dari kedua kelas dengan proporsi yang lebih adil.

Selanjutnya, dalam penelitian lain yang membandingkan algoritma K-NN dan Random Forest dalam mendeteksi masalah kesehatan mental, ditemukan bahwa Random Forest memberikan hasil yang lebih stabil dan memiliki

kemampuan generalisasi yang lebih baik dibandingkan model berbasis jarak seperti K-NN, terutama ketika dataset bersifat kompleks dan multidimensi [7].

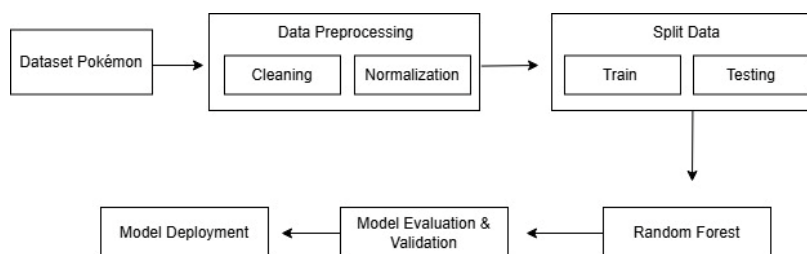
Pada aplikasi lain, Random Forest juga digunakan untuk memprediksi waktu tunggu pasien dalam sistem layanan kesehatan, menunjukkan bahwa algoritma ini mampu menghasilkan prediksi dengan akurasi tinggi, eror yang relatif kecil, serta mudah dipahami oleh praktisi lapangan. Bahkan dalam konteks deteksi gerakan dalam game, kombinasi Random Forest dengan teknologi sensor berbasis real-time menunjukkan potensi besar dalam meningkatkan responsibilitas sistem tanpa mengorbankan akurasi [8]. Oleh karena itu, dalam penelitian ini dilakukan klasifikasi tipe Pokémon menggunakan algoritma Random Forest berdasarkan statistik tempur yang tersedia. Tujuannya adalah untuk menghasilkan model prediksi yang dapat membantu pemain dalam memahami hubungan antara statistik Pokémon dengan tipe-tipe tertentu, serta mendukung pengambilan keputusan strategi dalam membangun tim yang optimal. Model yang dikembangkan akan dievaluasi menggunakan metrik evaluasi seperti akurasi, presisi, recall, dan F1-score, guna memastikan kinerja yang maksimal dan kemampuan generalisasi terhadap data baru [9].

Selain itu, hasil dari penelitian ini diharapkan dapat memberikan kontribusi bagi pengembangan sistem klasifikasi berbasis data dalam konteks dunia hiburan digital, serta menjadi dasar bagi studi lanjutan yang menggunakan algoritma lain seperti SVM, Gradient Boosting, atau Deep Learning dalam konteks yang serupa [10].

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Untuk melakukan studi penelitian, penting untuk memiliki langkah-langkah metodologis terstruktur untuk memandu peneliti dalam menyelesaikan studi secara efektif. Langkah-langkah ini bertujuan untuk menjawab pertanyaan penelitian yang ada dan mencapai tujuan yang telah dirumuskan sebelumnya. Diagram blok Gambar 1 menyediakan sistem atau diagram yang membantu memvisualisasikan langkah-langkah penelitian. Gambar 1 menunjukkan proses klasifikasi tipe utama Pokémon berdasarkan statistik tempurnya menggunakan algoritma Random Forest. Metode ini dimulai dengan pengumpulan dataset Pokémon, yang terdiri dari 995 sampel dengan atribut seperti nama, type dan Abilities (HP, Att Statistik serangan (Attack), Def Statistik pertahanan (Defense), S.Att Statistik serangan khusus (Special Attack), S.Def Statistik pertahanan khusus (Special Defense), dan Spd Statistik kecepatan (Speed)). Fase pertama adalah praperlakuan data, yang meliputi pembersihan data dengan menghilangkan outlier dan menangani nilai yang hilang, diikuti oleh normalisasi menggunakan Min-Max Scaling untuk memastikan bahwa data dalam kondisi baik.



Gambar 1. Tahapan Penelitian

Ulasan lengkap dari Gambar tahapan penelitian yang terdiri dari beberapa tahapan utama:

- Dataset Pokémon awal (pokemon_bw.csv) yang berisi atribut seperti: No, Name, Type, Abilities, HP, Att, Def, S.Att, S.Def, Spd.
- Data Preprocessing, Cleaning dan Normalization
- Split Data
Membagi dataset menjadi dua bagian yaitu Train Set, Digunakan untuk melatih model. Dan Testing Set, Digunakan untuk menguji performa model.
- Random Forest
Melatih model Random Forest Classifier menggunakan data latih. Parameter Random Forest dapat disesuaikan, misalnya Jumlah pohon keputusan (n_estimators), Kedalaman maksimum pohon (max_depth) dan Ukuran subset acak fitur (max_features)
- Model Evaluation & Validation
Menggunakan data uji untuk mengevaluasi performa model.
- Model Deployment
Setelah model dinilai cukup baik, model dapat dideploy untuk prediksi baru. Selanjutnya Implementasi dapat dilakukan dalam bentuk aplikasi sederhana atau API untuk penggunaan praktis.

2.2 Data Preprocessing

Data Preprocessing adalah gabungan dari beberapa informasi atau data mentah yang terdiri dari penjelasan yang berhubungan dan dapat diolah, dikelola, dan dianalisis[7]. Tujuannya adalah meningkatkan kualitas data sehingga model dapat belajar lebih baik dan memberikan hasil prediksi yang akurat [11],[12],[13].

2.2.1 Cleaning

Cleaning merupakan pembersihan data termasuk kedalam tahap awal dari data preprocessing, yaitu tugasnya untuk menyeleksi data dan membuang data yang berpotensi mengurangi akurasi dan kualitas dari hasil proses[9]. Fungsi dari kode tersebut adalah untuk memuat data dari file CSV ke dalam DataFrame serta mengambil hanya tipe pertama dari kolom Type pada dataset Pokémon. Tujuan utamanya adalah membaca dataset yang berisi informasi seperti nomor Pokédex, nama, tipe, dan statistik tempur Pokémon, sekaligus menyederhanakan masalah klasifikasi yang awalnya multilabel menjadi single-label, karena beberapa Pokémon memiliki dua tipe seperti "bug, flying" atau "grass, poison". Sebagai contoh, entri "grass, poison" akan diubah menjadi "grass", sedangkan tipe tunggal seperti "normal" tetap tidak berubah. Proses ini dilakukan agar model klasifikasi dapat lebih mudah diproses dan diimplementasikan menggunakan algoritma machine learning seperti Random Forest, sehingga meningkatkan efisiensi dan akurasi prediksi model.

Load dataset

```
df = pd.read_csv("pokemon_bw.csv")
```

 (1)

Ekstrak Tipe Utama (Type_1)

```
df['Type_1'] = df['Type'].apply(lambda x: x.split(',')[0].strip())
```

 (2)

2.2.2 Normalization

Fungsi dari StandardScaler() adalah melakukan standarisasi fitur dengan mengubah distribusi data menjadi memiliki rata-rata nol dan deviasi standar satu. Tujuan utamanya adalah untuk menyamakan skala antar fitur agar tidak terjadi bias terhadap fitur yang bernilai besar. Meskipun algoritma Random Forest tidak memerlukan normalisasi karena berbasis pohon keputusan, proses ini tetap dilakukan untuk beberapa tujuan tertentu seperti meningkatkan performa jika dibandingkan dengan model lain, atau agar kompatibel dengan pipeline atau sistem yang melibatkan algoritma yang sensitif terhadap skala data seperti SVM atau KNN[9], .

Standardisasi

```
scaler = StandardScaler()
```

```
X_scaled = scaler.fit_transform(X)
```

2.3 Random Forest

Random Forest adalah algoritma ensemble learning berbasis metode pohon keputusan (decision tree) yang menggunakan teknik Bootstrap Aggregation (Bagging) untuk meningkatkan akurasi prediksi dan mengurangi overfitting[14]. Random Forest digunakan untuk melakukan klasifikasi tipe Pokémon berdasarkan statistik tempurnya , yaitu: (HP, Att Statistik serangan (Attack), Def Statistik pertahanan (Defense), S.Att Statistik serangan khusus (Special Attack), S.Def Statistik pertahanan khusus (Special Defense), dan Spd Statistik kecepatan (Speed) [15],[16],[17].

3. HASIL DAN PEMBAHASAN

3.1 Dataset

Dataset adalah gabungan dari beberapa informasi atau data mentah yang terdiri dari penjelasan yang berhubungan dan dapat diolah, dikelola, dan dianalisis. Jenis data yang termasuk ke dalam kumpulan data dan format yang terstruktur, seperti tabel atau file, dan berisi informasi dari berbagai sumber seperti angka dan gambar [7]. Dataset yang digunakan adalah "pokemon_bw.csv" yang berisi informasi statistik tempur dari 995 Pokémon. Kolom "Type" diubah menjadi "Type_1" untuk merepresentasikan tipe utama. Fitur yang digunakan meliputi HP, Attack, Defense, Special Attack, Special Defense, dan Speed. Dataset dibagi menjadi data latih dan uji dengan rasio 80:20. Algoritma Random Forest diterapkan dan evaluasi dilakukan menggunakan akurasi, confusion matrix, dan ROC-AUC[16]. Penelitian ini menggunakan metode data mining dengan algoritma Random Forest dengan menggunakan Orange Data Mining. Random Forest bekerja dengan mengambil sampel acak dari dataset, membentuk decision tree pada setiap sampel, mendapatkan hasil prediksi dari setiap tree, dan melakukan voting pada hasil prediksi untuk menghasilkan prediksi akhir [18]. Atribut yang dipakai pada dataset dapat dilihat pada Tabel 1[14].

Tabel 1. Atribut Dataset

No	Name	Type	Abilities	HP	Att	Def	S.Att	S.Def	Spd
10	Caterpie	bug	Shield Dust, Run Away	45	30	35	20	20	45
11	Metapod	bug	Shed Skin, Shed Skin	50	20	55	25	25	30
12	Butterfree	bug, flying	Compoundeyes, Tinted Lens	60	45	50	80	80	70

13	Weedle	bug, poison	Shield Dust, Run Away	40	35	30	20	20	50
14	Kakuna	bug, poison	Shed Skin, Shed Skin	45	25	50	25	25	35
15	Beedrill	bug, poison	Swarm, Sniper	65	80	40	45	80	75
46	Paras	bug, grass	Effect Spore, Dry Skin, Damp	35	70	55	45	55	25
47	Parasect	bug, grass	Effect Spore, Dry Skin, Damp	60	95	80	60	80	30
48	Venonat	bug, poison	Compoundeyes, Tinted Lens, Run Away	60	55	50	40	55	45
49	Venomoth	bug, poison	Shield Dust, Tinted Lens, Wonder Skin	70	65	60	90	75	90

3.2 Classification Report

Gambar 2 Menunjukkan hasil evaluasi model klasifikasi menggunakan Classification Report, yang memberikan detail performa model untuk setiap tipe Pokémon (kelas) dalam dataset. Berikut adalah hasil gambar dari Classification Report. Model klasifikasi tipe Pokémon berdasarkan statistik tempur menggunakan algoritma Random Forest mencapai akurasi global sebesar 64.8%, yang menunjukkan bahwa model mampu memprediksi tipe Pokémon dengan benar pada sekitar 64.8% dari total data uji. Berdasarkan classification report, performa model bervariasi di setiap kelas, seperti pada kelas bug dengan precision 60%, recall 83%, dan F1-score 70%, sedangkan untuk kelas dragon, model menunjukkan hasil yang lebih baik dengan nilai precision, recall, dan F1-score masing-masing 86%. Namun, ada beberapa kelas seperti flying yang memiliki nilai precision, recall, dan F1-score sama dengan nol, menandakan model tidak mampu mendeteksi kelas tersebut akibat jumlah sampel yang sangat sedikit (support = 2). Rata-rata macro average memberikan nilai precision 71%, recall 58%, dan F1-score 61%, menunjukkan performa keseluruhan yang cukup baik, sementara weighted average menghasilkan precision 68%, recall 65%, dan F1-score 64%, yang memperhitungkan ketidakseimbangan jumlah sampel antar kelas, sehingga memberikan gambaran yang lebih realistis tentang kemampuan model secara keseluruhan.

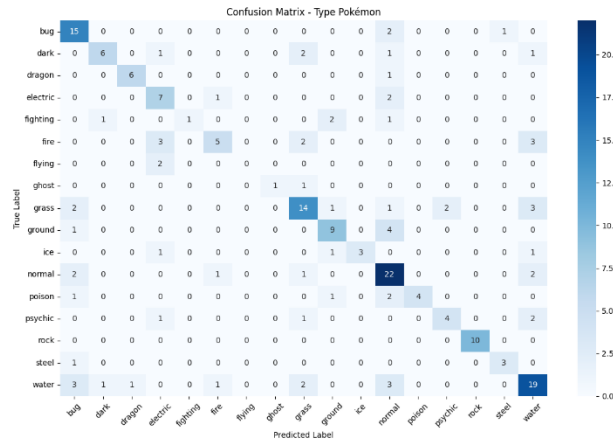
Accuracy: 0.6482412060301508

Classification Report:				
	precision	recall	f1-score	support
bug	0.60	0.83	0.70	18
dark	0.75	0.55	0.63	11
dragon	0.86	0.86	0.86	7
electric	0.47	0.70	0.56	10
fighting	1.00	0.20	0.33	5
fire	0.62	0.38	0.48	13
flying	0.00	0.00	0.00	2
ghost	1.00	0.50	0.67	2
grass	0.61	0.61	0.61	23
ground	0.64	0.64	0.64	14
ice	1.00	0.50	0.67	6
normal	0.56	0.79	0.66	28
poison	1.00	0.50	0.67	8
psychic	0.67	0.50	0.57	8
rock	1.00	1.00	1.00	10
steel	0.75	0.75	0.75	4
water	0.61	0.63	0.62	30
accuracy			0.65	199
macro avg	0.71	0.58	0.61	199
weighted avg	0.68	0.65	0.64	199

Gambar 2. Classification Report

3.3 Confusion Matrix

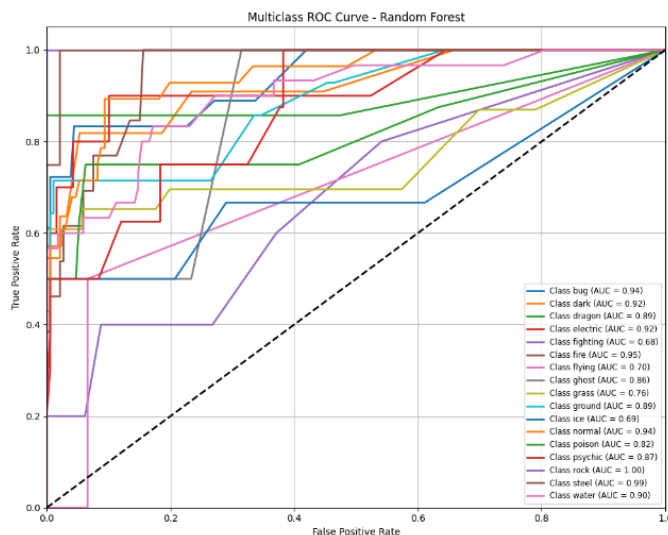
Gambar 3 dibawah ini menunjukkan Confusion Matrix untuk model klasifikasi tipe Pokemon. Confusion Matrix adalah alat visualisasi yang digunakan untuk mengukur performa model klasifikasi dengan memperlihatkan jumlah prediksi benar dan salah per kelas. Confusion Matrix pada penelitian ini menampilkan kinerja model klasifikasi tipe Pokémon dengan membandingkan label sebenarnya (sumbu Y / True Label) dan label yang diprediksi oleh model (sumbu X / Predicted Label), di mana nilai-nilai pada diagonal biru tua menunjukkan jumlah prediksi benar untuk setiap kelas, seperti 22 prediksi benar untuk kelas normal, 19 untuk water, dan 15 untuk bug, sementara nilai di luar diagonal menggambarkan kesalahan klasifikasi, contohnya beberapa Pokémon bertipe grass diprediksi sebagai ground (nilai 9) dan rock diprediksi sebagai steel (nilai 10), menunjukkan adanya pola kesalahan antar kelas tertentu; skala warna dari biru tua ke putih terang membantu memvisualisasikan intensitas frekuensi prediksi, dengan warna biru tua merepresentasikan jumlah tinggi dan warna terang menunjukkan frekuensi rendah.



Gambar 3. Confusion Matrix

3.4 ROC Curve (Receiver Operating Characteristic Curve)

Gambar ini menunjukkan ROC Curve (Receiver Operating Characteristic Curve) untuk model klasifikasi tipe Pokémon menggunakan algoritma Random Forest. ROC Curve adalah grafik yang menggambarkan kemampuan model dalam membedakan antara kelas positif dan negatif, dengan sumbu x merepresentasikan False Positive Rate (FPR) (tingkat kesalahan prediksi salah) dan sumbu y merepresentasikan True Positive Rate (TPR) (sensitivitas atau recall). ROC Curve digunakan untuk mengevaluasi performa model klasifikasi, terutama dalam konteks multikelas seperti klasifikasi tipe Pokémon, di mana setiap kurva mewakili satu kelas dengan pendekatan One-vs-Rest (OvR), dan nilai AUC (Area Under the Curve) menjadi indikator utama kemampuan model dalam membedakan antar kelas; komponen utamanya meliputi kurva berwarna yang merepresentasikan tiap kelas seperti bug (AUC = 0.94), dark (AUC = 0.92), dan dragon (AUC = 0.89) serta garis diagonal putus-putus sebagai baseline prediksi acak, di mana semakin jauh kurva dari garis ini menunjukkan performa yang semakin baik; interpretasi kurva menunjukkan bahwa kelas seperti rock (AUC = 1.00) dan steel (AUC = 0.99) memiliki performa hampir sempurna, sedangkan kelas flying (AUC = 0.70) masih relatif sulit diklasifikasikan secara akurat dibandingkan kelas lainnya, sehingga dapat disimpulkan bahwa model menunjukkan hasil yang sangat baik pada sebagian besar kelas, namun masih membutuhkan peningkatan pada kelas dengan AUC rendah agar kinerja model menjadi lebih optimal secara keseluruhan.



Gambar 3. ROC Curve

4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan algoritma Random Forest untuk melakukan klasifikasi tipe Pokémon berdasarkan statistik tempur seperti HP, Attack, Defense, Special Attack, Special Defense, dan Speed. Dengan menggunakan dataset pokemon_bw.csv dan tahapan preprocessing yang melibatkan pembersihan data, label encoding, standarisasi, serta pembagian data latih dan uji dengan rasio 80:20, model mampu mencapai akurasi sebesar 64,8%, menunjukkan bahwa Random Forest cukup efektif dalam mempelajari pola hubungan antara statistik tempur dan tipe Pokémon. Hasil evaluasi tambahan melalui classification report, confusion matrix, dan ROC Curve membuktikan bahwa beberapa tipe seperti 'rock', 'steel', dan 'dragon' memiliki performa prediksi yang tinggi, sedangkan tipe 'flying' dan 'ghost'

masih relatif sulit diklasifikasikan secara akurat. Kontribusi utama penelitian ini adalah memberikan analisis awal mengenai potensi prediksi tipe Pokémon berbasis statistik tempur, yang dapat menjadi dasar bagi sistem rekomendasi tim Pokémon atau strategi pertandingan otomatis. Sebagai langkah lanjutan, disarankan untuk meningkatkan performa model dengan teknik optimasi parameter seperti GridSearchCV, menangani ketidakseimbangan kelas menggunakan SMOTE, serta mempertimbangkan pendekatan multilabel jika tipe ganda tetap ingin dimasukkan ke dalam analisis. Penelitian ini juga membuka peluang penggunaan algoritma lain seperti Gradient Boosting atau Deep Learning untuk komparasi hasil dan peningkatan generalisasi model pada data baru.

REFERENCES

- [1] D. Navas and D. Donohue, “Competitive Pokémon Usage Tier Classification,” in *Advances in Data Science and Information Engineering: Proceedings from ICDATA*, Springer, 2021, pp. 599–607.
- [2] H. Sunaryanto, M. A. Hasan, and G. Guntoro, “Classification Analysis of Unilak Informatics Engineering Students Using Support Vector Machine (SVM), Iterative Dichotomiser 3 (ID3), Random Forest and K-Nearest Neighbors (KNN),” *IT J. Res. Dev.*, vol. 7, no. 1, pp. 36–42, 2022, doi: 10.25299/itjrd.2022.8912.
- [3] I. Kurniawan, D. C. P. Buani, A. Abdussomad, W. Apriliah, and R. A. Saputra, “Implementasi algoritma random forest untuk menentukan penerima bantuan raskin,” *J. Teknol. Inf. Dan Ilmu Komput.*, vol. 10, no. 2, pp. 421–428, 2023.
- [4] A. V. Alzuhdi, H. A. Rosyid, M. Y. Chuttur, and S. Nazir, “Optimizing Random Forest Algorithm to Classify Player’s Memorisation via In-game Data,” *Knowl. Eng. Data Sci.*, vol. 6, no. 1, p. 103, 2023.
- [5] S. Howay and S. Suhirman, “Comparison of SVM, NBC, and KNN Classification Methods in Determining Students’ Majors at SMK N02 Manokwari,” *J. Comput. Sci. Technol. Stud.*, vol. 5, no. 1, pp. 15–23, 2023, doi: 10.32996/jcsts.2023.5.1.3.
- [6] I. L. Mulyahati, “Implementasi machine learning prediksi harga sewa apartemen menggunakan algoritma Random Forest melalui framework website Flask Python pada website mamikos. com,” *Data Sci.*, vol. 1, no. 3, pp. 371–379, 2020.
- [7] N. Nurdiansyah, F. S. Febriyan, Z. Gesit, and D. Amanta, “Mental Health Analysis to Prevent Mental Disorders in Students Using The K-Nearest Neighbor (K-NN),” *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 5, no. January, pp. 1–9, 2025, doi: <https://doi.org/10.57152/malcom.v5i1.1537>.
- [8] K. Mustaqim, N. Riza, and M. R. Hidayatullah, “Perbandingan Metode Regresi Linear Dan K-Nearest Neighbor (KNN) Dalam Memprediksi Produksi Tanaman Padi Di Pulau Sumatera,” *Data Sci. Indones.*, vol. 4, no. 2, pp. 60–74, 2024.
- [9] Sopiatal Ulum, R. F. Alifa, P. Rizkika, and C. Rozikin, “Perbandingan Performa Algoritma KNN dan SVM dalam Klasifikasi Kelayakan Air Minum,” *Gener. J.*, vol. 7, no. 2, pp. 141–146, 2023, doi: 10.29407/gj.v7i2.20270.
- [10] A. Rozaq, A. Samudra, and A. Islam, “Analisis Desain Visual Karakter Kleavor Dalam Game Pokémon Legends: Arceus,” *J. Barik*, vol. 5, no. 1, pp. 298–309, 2023.
- [11] Baskoro, Sriyanto, and L. Setya Rini, “Prediksi Penerima Beasiswa dengan Menggunakan Teknik Data Mining di Universitas Muhammadiyah Pringsewu,” *Semin. Nas. Has. Penelit. dan Pengabd. Masy. Inst. Inform. dan Bisnis Darmajaya*, vol. 1, no. 2, pp. 87–94, 2021.
- [12] A. W. Fathurrahman, M. Thoriquulhaq, and F. Arianto, “Penerapan Machine Learning untuk Pengklasifikasian Hoaks pada Platform Media Sosial,” *J. Senada*, vol. 1, no. 3, pp. 66–68, 2022.
- [13] E. Fitri and S. N. Nugraha, “Optimasi Kinerja Linear Regression, Random Forest Regression Dan Multilayer Perceptron Pada Prediksi Hasil Panen,” *Inti Nusa Mandiri*, vol. 18, no. 2, pp. 210–217, 2024.
- [14] A. A. Karim, M. A. Prasetyo, and M. R. Saputro, “Perbandingan Metode Random Forest, K-Nearest Neighbor, dan SVM Dalam Prediksi Akurasi Pertandingan Liga Italia,” *Pros. Semin. Nas. Teknol. dan Sains*, vol. 2, no. 3, pp. 377–342, 2023.
- [15] A. B. Prakoso, M. A. Putra, M. H. Hilmi, S. Y. P. Risky, and J. Maulindar, “Penerapan Algoritma Regresi Random Forest Untuk Prediksi Produksi Jagung Menggunakan Data Statistik Sistem Pertanian Cerdas Smart City,” in *Prosiding Seminar Nasional Teknologi Informasi dan Bisnis*, 2025, pp. 28–33.
- [16] S. Hermawan and S. Budi, “Analisis dan Prediksi Pertempuran Game Of Thrones Menggunakan Algoritma Random Forest dan Logistic Regression,” *J. Strateg.*, vol. 3, no. 2, pp. 454–461, 2021.
- [17] M. Putri, “Prediksi penyakit stroke menggunakan machine learning dengan algoritma random forest,” *J. Infomedia Tek. Inform. Multimedia, dan Jar.*, vol. 9, no. 1, pp. 16–21, 2024.
- [18] R. P. Munggaran and M. Nurmalasari, “Predicting Outpatient Service Waiting Times with Random Forest Algorithm,” *Data Sci. Indones.*, vol. 5, no. 1, pp. 35–40, 2025.