Analisis Kinerja Algoritma *Naïve Bayes* dan *Random Forest* dalam Memprediksi Hasil Klasemen *English Premier League*

1st Muhamad Ibnu Rizky
Universitas Buana Perjuangan Karawang
Karawang, Indonesia
if21.muhamadrizky@mhs.ubpkarawang.ac.id

2nd Sutan Faisal
Universitas Buana Perjuangan Karawang
Karawang, Indonesia
sutan.faisal@ubpkarawang.ac.id

3rd Iman Sanjaya
Universitas Buana Perjuangan Karawang
Karawang, Indonesia
iman.sanjaya@ubpkarawang.ac.id

4th Deden Wahidin
Universitas Buana Perjuangan Karawang
Karawang, Indonesia
deden.wahiddin@ubpkarawang.ac.id

***Abstract*— *English Premier League* dikenal sebagai liga sepak bola paling kompetitif dan menarik untuk dianalisis secara statistik. Penelitian ini dilakukan untuk perbandingan kinerja dua algoritma *machine learning*, yakni *Naïve Bayes* dan *Random Forest*, dalam memprediksi posisi akhir klasemen berdasarkan data statistik pertandingan. Data dikumpulkan melalui teknik *web scraping* dari situs *FBref.com*, mencakup tiga musim kompetisi serta sejumlah fitur relevan lainnya. Setelah melalui tahap *preprocessing*, data kemudian dibagi menjadi data pelatihan dan data pengujian dengan rasio 80:20. Evaluasi terhadap model dilakukan menggunakan berbagai metrik, seperti akurasi, presisi, recall, *F1-score*, serta *confusion matrix*. Hasil dari pengujian memperlihatkan algoritma *Naïve Bayes* meraih akurasi sebesar 69,83% dan menunjukkan kinerja yang cukup baik dalam memprediksi hasil seri. Di sisi lain, algoritma *Random Forest* menunjukkan hasil performa yang unggul dengan akurasi mencapai 99,57%, serta nilai presisi, *recall*, *dan F1-score* yang tinggi dan konsisten. Prediksi klasemen akhir yang dihasilkan oleh *Random Forest* juga lebih mendekati hasil sebenarnya. Berdasarkan temuan ini, dapat disimpulkan bahwa *Random Forest* lebih mampu menangani kompleksitas data pertandingan sepak bola dan lebih direkomendasikan untuk digunakan dalam sistem prediksi di bidang analitik olahraga.**

Kata kunci — Machine Learning, Naïve Bayes, Random Forest, Prediksi Klasemen, Premier League

I. PENDAHULUAN (*HEADING 1*)

Sepak bola menjadi cabang olahraga yang paling banyak diminati oleh masyarakat di berbagai belahan dunia, dan *English Premier League* (EPL) terkenal sebagai salah satu kompetisi sepak bola paling kompetitif secara global. Setiap tahunnya, sebanyak 20 tim papan atas dari Inggris saling bersaing untuk mendapatkan gelar juara EPL. Persaingan yang ketat serta pertandingan yang penuh tensi tinggi menjadikan prediksi pemenang liga sangat menarik namun juga menantang. Hingga saat ini, klub yang selalu berpartisipasi dalam setiap musim EPL, yaitu *Arsenal, Chelsea, Everton, Liverpool, Manchester United, dan Tottenham Hotspur.*

Pada sebuah data dalam pertandingan, statistik adalah komponen yang utama [1]. Statistik dalam dunia olahraga, terutama sepak bola, mencakup berbagai elemen yang dapat dimanfaatkan untuk menilai kinerja sebuah tim. Untuk meningkatkan ketepatan dalam memprediksi hasil pertandingan, peran analisis statistik menjadi sangat krusial. Tingkat akurasi prediksi sangat dipengaruhi oleh relevansi data statistik pertandingan serta keandalan analisis yang dilakukan sebelum pertandingan berlangsung [2]. Hasil pertandingan sepak bola bersifat tidak pasti, namun melalui analisis data dari berbagai pertandingan sebelumnya, dapat ditemukan pola-pola tertentu yang dapat dimanfaatkan untuk memperkirakan hasil pertandingan yang akan datang [3].

Prediksi hasil pertandingan kerap menjadi topik yang menarik bagi para penggemar sepak bola. Seiring dengan kemajuan teknologi, berbagai data penting terkait pertandingan kini dapat diakses dengan lebih mudah. Informasi tersebut dapat dianalisis dan dimanfaatkan untuk memperkirakan hasil pertandingan di masa depan. Salah satu cabang dalam ilmu komputer yang sering kali digunakan untuk membuat prediksi berbasis data adalah *Machine Learning* [4].

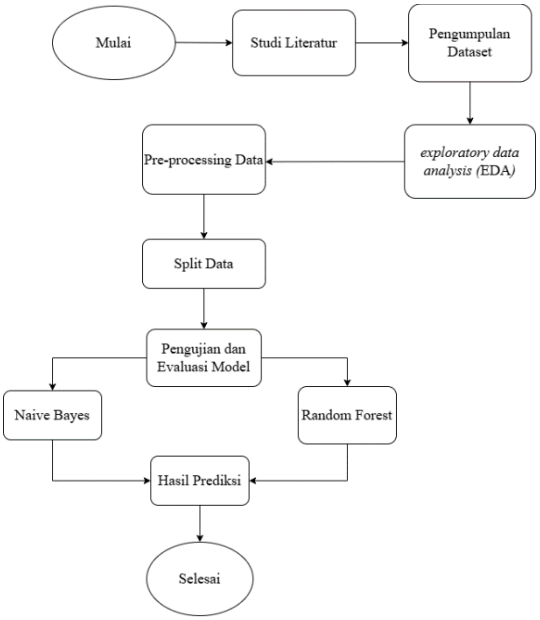
Algoritma *Naive Bayes* dan *Random Forest* merupakan dua algoritma yang cukup populer dalam bidang prediksi. Algoritma *Naive Bayes* mengandalkan konsep probabilitas dalam membuat prediksi, sementara *Random Forest* bekerja dengan menggabungkan sejumlah pohon keputusan guna menghasilkan prediksi akurat. Kedua algoritma ini telah diterapkan dalam berbagai penelitian terkait prediksi hasil pertandingan sepak bola, dan hasilnya menunjukkan bahwa keduanya mampu memberikan tingkat akurasi prediksi yang tinggi.

Penelitian tentang prediksi pada pertandingan olahraga menggunakan algoritma *Naïve Bayes* sudah banyak dilakukan. Salah satu penelitian yang berjudul "Prediksi Hasil Pertandingan Liga *Serie A* Menggunakan Metode *Naïve Bayes*" oleh Ridwan Adi Pratama, Winda Apriandari, Didik Indrayana. Dalam penelitian ini, menggunakan sejumlah atribut atau variabel dalam dataset, antara lain: HT, AT, HTHG, HTAG, HS, AS, HST, AST, HF, AF, HC, AC, HY, AY, HR, AR, dan FTR. Penelitian memprediksi hasil

pertandingan ini berdasarkan klasemen akhir Liga *Serie A*. Dengan menerapkan algoritma *Naïve Bayes*, diperoleh tingkat akurasi sebesar 75,79% [5]. Penelitian berjudul "Perbandingan Metode *Random Forest, K-Nearest Neighbor*, dan *SVM* Dalam Prediksi Akurasi Pertandingan Liga Italia" yang dilakukan oleh Ahmad Assril Karim, Muhammad Ary Prasetyo, dan Muhammad Rohid Saputro membagi dataset sebanyak 380 data menjadi dua data, yaitu 80% sebagai data latih dan 20% sebagai data uji. Dari hasil prediksi algoritma *Random Forest*, diperoleh akurasi sebesar 62%. Persentase akurasi tersebut diperoleh berdasarkan perhitungan dari data pelatihan yang digunakan [6].

Keakuratan prediksi Tingkat akurasi dalam prediksi sangat dipengaruhi oleh kualitas data yang digunakan. Jika data bersifat tidak lengkap, tidak terstruktur dengan baik, atau kurang relevan, hal tersebut dapat menurunkan kinerja model prediktif yang dibangun. Selain itu, pemahaman terhadap hasil prediksi juga perlu diperhatikan, karena dalam penerapan *Machine Learning*, tidak hanya akurasi yang penting, tetapi juga bagaimana cara model membuat keputusan. Berdasarkan hal tersebut, penelitian ini akan memusatkan perhatian pada analisis serta perbandingan antara dua algoritma untuk menilai akurasi, efektivitas, dan tingkat interpretabilitas dalam memprediksi peringkat akhir di English Premier League. Diharapkan hasil dari studi ini dapat bermanfaat dalam mengembangkan model prediksi yang lebih andal dan efisien di bidang sepak bola.

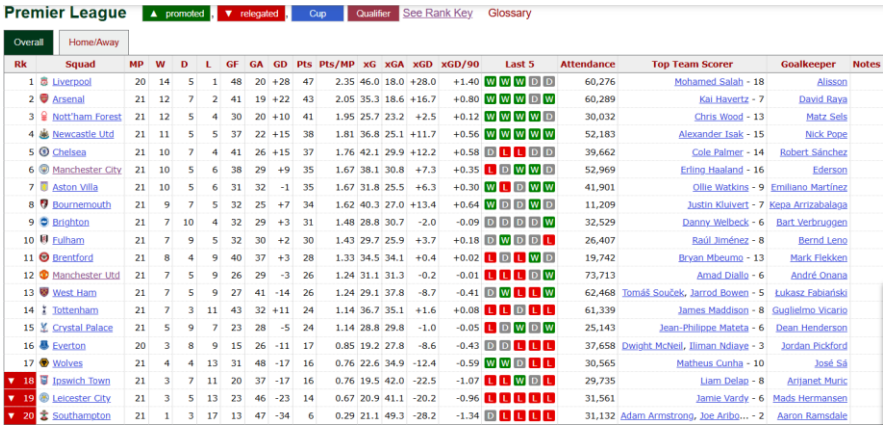
II. METODE PENELITIAN



Gambar 1. Flowchart Alur Penelitian

A. Pengumpulan Data

Untuk mendapatkan dataset berkualitas, komprehensif, dan relevan guna mendukung proses model, perlu tahap pengumpulan data. Dalam studi ini, data dikumpulkan melalui *web scraping* dari situs *FBref.com*, yang dikenal sebagai sumber terpercaya dalam menyediakan statistik lengkap pertandingan sepak bola, termasuk kompetisi *English Premier League* (EPL). Peneliti menentukan halaman web spesifik yang mengandung data yang diperlukan, seperti halaman yang mencakup statistik pertandingan, skor, dan performa tim di setiap musim EPL.



Premier League ▲ promoted ▼ relegated Cup Qualifier See Rank Key Glossary

Overall | Home/Away

Rk	Squad	MP	W	D	L	GF	GA	GD	Pts	Pts/MP	xG	xGA	xGD	xGD/90	Last 5	Attendance	Top Team Scorer	Goalkeeper	Notes
1	Liverpool	20	14	5	1	48	20	+28	47	2.35	46.0	18.0	+28.0	+1.40	W W W D D	60,276	Mohamed Salah - 18	Alisson	
2	Arsenal	21	12	7	2	41	19	+22	43	2.05	35.3	18.6	+16.7	+0.80	W W W D W	60,289	Kai Havertz - 7	David Raya	
3	Nott'ham Forest	21	12	5	4	30	20	+10	41	1.95	25.7	23.2	+2.5	+0.12	W W W W D	30,032	Chris Wood - 13	Matz Sels	
4	Newcastle Utd	21	11	5	5	37	22	+15	38	1.81	36.8	25.1	+11.7	+0.56	W W W W W	52,183	Alexander Isak - 15	Nick Pope	
5	Chelsea	21	10	7	4	41	26	+15	37	1.76	42.1	29.9	+12.2	+0.58	D L L D D	39,662	Cole Palmer - 14	Robert Sánchez	
6	Manchester City	21	10	5	6	38	29	+9	35	1.67	38.1	30.8	+7.3	+0.35	L D W W D	52,969	Erling Haaland - 16	Ederson	
7	Aston Villa	21	10	5	6	31	32	-1	35	1.67	31.8	25.5	+6.3	+0.30	W L D W W	41,901	Ollie Watkins - 9	Emiliano Martínez	
8	Bournemouth	21	9	7	5	32	25	+7	34	1.62	40.3	27.0	+13.4	+0.64	W D D W D	11,209	Justin Kluivert - 7	Kepa Arrizabalaga	
9	Brighton	21	7	10	4	32	29	+3	31	1.48	28.8	30.7	-2.0	-0.09	D D D D W	32,529	Danny Welbeck - 6	Bart Verbruggen	
10	Fulham	21	7	9	5	32	30	+2	30	1.43	29.7	25.9	+3.7	+0.18	D W D D L	26,407	Raúl Jiménez - 8	Bernd Leno	
11	Brentford	21	8	4	9	40	37	+3	28	1.33	34.5	34.1	+0.4	+0.02	L D L W D	19,742	Bryan Mbeumo - 13	Mark Flekken	
12	Manchester Utd	21	7	5	9	26	29	-3	26	1.24	31.1	31.3	-0.2	-0.01	L L L D W	73,713	Amad Diallo - 6	André Onana	
13	West Ham	21	7	5	9	27	41	-14	26	1.24	29.1	37.8	-8.7	-0.41	D W L L W	62,468	Tomáš Souček, Jarrod Bowen - 5	Łukasz Fabiański	
14	Tottenham	21	7	3	11	43	32	+11	24	1.14	36.7	35.1	+1.6	+0.08	L L D L L	61,339	James Maddison - 8	Guglielmo Vicario	
15	Crystal Palace	21	5	9	7	23	28	-5	24	1.14	28.8	29.8	-1.0	-0.05	L D W D W	25,143	Jean-Philippe Mateta - 6	Dean Henderson	
16	Everton	20	3	8	9	15	26	-11	17	0.85	19.2	27.8	-8.6	-0.43	D D L L L	37,658	Dwight McNeil, Iliman Ndiaye - 3	Jordan Pickford	
17	Wolves	21	4	4	13	31	48	-17	16	0.76	22.6	34.9	-12.4	-0.59	W W D L L	30,565	Matheus Cunha - 10	José Sá	
▼ 18	Ipswich Town	21	3	7	11	20	37	-17	16	0.76	19.5	42.0	-22.5	-1.07	L L W D L	29,735	Liam Delap - 8	Arijanet Muric	
▼ 19	Leicester City	21	3	5	13	23	46	-23	14	0.67	20.9	41.1	-20.2	-0.96	L L L L L	31,561	Jamie Vardy - 6	Mads Hermansen	
▼ 20	Southampton	21	1	3	17	13	47	-34	6	0.29	21.1	49.3	-28.2	-1.34	D L L L L	31,132	Adam Armstrong, Joe Aribo... - 2	Aaron Ramsdale	

Gambar 2. Website Fbref.com

Proses pengumpulan data dilakukan dengan menggunakan pustaka *Python BeautifulSoup* untuk mengekstrak data dari halaman *web* yang relevan dan disimpan menjadi file *CSV*. Data yang diambil mencakup statistik dari 3 (tiga) musim yaitu musim 2021-2022, 2022-2023, 2023-2024. Data yang diekstrak mencakup berbagai informasi historis dari pertandingan sepak bola, yang meliputi skor akhir, jumlah gol yang tercetak, serta berbagai statistik permainan seperti penguasaan bola, jumlah tembakan, jumlah pelanggaran, jumlah offside, dan statistik lainnya yang relevan.

B. Proses Validasi Data

Untuk memastikan kualitas dan keakuratan data yang diperoleh dari scraping, langkah-langkah validasi data dilakukan dengan teliti, khususnya dalam memastikan bahwa data yang diambil sesuai dengan standar yang diinginkan. Berikut adalah penjelasan lebih detail mengenai proses validasi data

Dalam *scraping data* dari website seperti *FBref.com*, data yang diambil harus melewati tahap pengecekan untuk memastikan bahwa setiap atribut memiliki data yang sesuai. Kolom seperti skor akhir, jumlah gol, statistik permainan, dan informasi tim harus dilengkapi dengan baik agar tidak ada data yang terlewatkan. Data yang telah diekstrak melalui scraping dari *FBref.com* perlu dibandingkan dengan sumber lain yang lebih terpercaya, seperti laporan resmi dari pertandingan atau data dari platform lain yang sejenis. Jika terdapat ketidaksesuaian, maka perlu dilakukan pemeriksaan ulang dan penyesuaian pada data hasil scraping agar sesuai dengan fakta yang benar.

C. *Exploratory Data Analysis (EDA)*

Tahap *Exploratory Data Analysis (EDA)* adalah langkah awal yang esensial dalam proses analisis data. EDA bertujuan untuk memahami karakteristik dataset secara mendalam, mengidentifikasi pola dan anomali, serta menemukan hubungan antar fitur. Tahapan ini sangat penting sebelum melanjutkan ke proses analisis atau pemodelan machine learning, karena kualitas dan pemahaman data memiliki dampak langsung pada hasil akhir penelitian. Tahap *EDA* memberikan wawasan mendalam tentang karakteristik dataset, membantu dalam penyesuaian dan pembersihan data, serta memberikan landasan yang kuat untuk proses pemodelan machine learning.

D. *Pre-Processing*

Model *Preprocessing* data merupakan tahap krusial dalam siklus penelitian, agar data siap dipakai dalam analisis serta pemodelan *machine learning*. Proses ini bertujuan untuk memastikan data berkualitas baik, bebas dari kesalahan, dan diformat sesuai dengan kebutuhan algoritma yang akan diterapkan. Selain itu, *preprocessing* juga berfungsi untuk mengenali potensi permasalahan pada data yang dapat mempengaruhi hasil analisis, seperti data yang hilang, nilai yang tidak masuk akal, atau variabel yang kurang relevan.

1. . Identifikasi Missing Values

   Langkah pertama adalah melakukan pengecekan secara menyeluruh untuk menemukan nilai kosong yang ada di dalam dataset. Pengecekan dilakukan pada setiap kolom dan baris untuk mengidentifikasi seberapa banyak data yang hilang. Proses ini biasanya menggunakan fungsi-fungsi dari pustaka seperti pandas, misalnya isnull() atau info(), yang memberikan gambaran lengkap tentang jumlah nilai kosong pada setiap kolom. Tujuan dari langkah ini adalah memahami sejauh mana missing values tersebar dalam dataset sehingga dapat ditentukan strategi penanganannya.

2. Penanganan Missing Values

   Setelah nilai kosong diidentifikasi, langkah berikutnya adalah mengatasi nilai kosong tersebut dengan strategi yang sesuai berdasarkan jenis data dan tingkat keberadaannya. Berikut adalah pendekatan yang dilakukan:

3. . Seleksi Fitur

   Seleksi fitur merupakan proses pemilihan atribut atau kolom dalam dataset yang memiliki tingkat relevansi tinggi terhadap variabel target yang akan diprediksi. Tujuan dari langkah ini adalah untuk menyederhanakan model, mengurangi beban komputasi, serta meningkatkan akurasi prediksi dengan hanya menggunakan fitur-fitur yang memang diperlukan. Pada penelitian ini, pemilihan fitur didasarkan pada kaitannya dengan hasil pertandingan (Result) sebagai variabel target prediksi.

4. Encoding Variabel Kategorikal

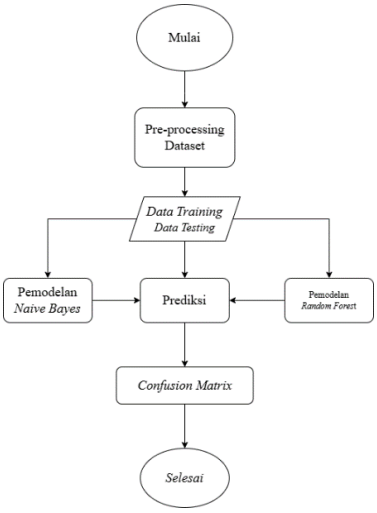
   Pada penelitian ini, variabel target result, yang berisi kategori hasil pertandingan, dikonversi menjadi nilai numerik menggunakan metode Label Encoding.

Tabel 1. Label Numerik

Atribut	Keterangan	Label
Lose	**Kalah**	**0**
Draw	**Seri**	**1**
Win	**Menang**	**2**

E. Pengujian dan Evaluasi Model Algoritma

Pengujian model algoritma dalam penelitian ini dilakukan untuk mengevaluasi performa dari algoritma *machine learning*, yaitu ***Naïve Bayes*** dan ***Random Forest***. Tujuan dari pengujian ini adalah untuk mengukur sejauh mana kedua algoritma dapat memprediksi hasil pertandingan *English Premier League* (EPL) [7].



Gambar 3. *Flowchart* Pengujian dan Evaluasi

1. Pembagian Data
   Sebelum tahap pengujian, data yang telah diproses akan dibagikan menjadi dua bahan data utama:
   a. *Data Training*: Digunakan untuk melatih model. Data training biasanya terdiri dari sekitar 70-80% dari keseluruhan dataset.
   b. *Data Testing*: Untuk menguji performa model. Data testing terdiri dari 20-30% dari keseluruhan dataset

2. Naïve Bayes
   Model *Naïve Bayes* digunakan untuk melakukan klasifikasi dengan asumsi bahwa setiap fitur bersifat independen terhadap kelasnya. Model ini biasanya efektif untuk mengolah data numerik maupun kategori, seperti hasil pertandingan.

$$\boldsymbol{P(A|B)\frac{P(B|A)\ x\ P(A)}{P(B)}} \qquad (\mathbf{1})$$

3. *Random Forest*
   Metode *Random Forest* (RF) adalah teknik yang dapat meningkatkan akurasi hasil, Karena dalam proses pembentukan setiap simpul pada suatu *node* dilakukan dengan cara yang tidak tetap atau bersifat acak [8]. Model ini sangat ampuh dalam mencegah *overfitting* karena prediksi akhir diperoleh dari gabungan banyak pohon keputusan. *Random Forest* mampu mengolah data numerik maupun kategorikal, serta dapat menilai seberapa penting setiap fitur dalam memprediksi target. Metode ini terdiri dari sejumlah pohon keputusan yang digunakan bersama-sama untuk mengklasifikasikan data ke dalam kelas tertentu.

$$\boldsymbol{l(y) = argmac_c(\sum_{n=1}^{N} I\, n_n(y) = c)} \qquad \mathbf{(2)}$$

   Keterangan:
   a. Pohon pada *Random Forest* memberikan prediksi untuk input y.
   b. Setiap prediksi pohon tersebut dihitung jumlahnya berdasarkan kelas c.
   c. Kelas dengan suara terbanyak (*argmax*) dipilih sebagai prediksi akhir l(y).

F. *Confusion Matrix*

Evaluasi model menggunakan confusion matrix untuk menilai kemampuan model untuk memprediksi kelas pada data uji. Prediksi data positif dengan benar, maka disebut ***True Positive*** **(TP).** Sebaliknya, jika data positif diprediksi secara keliru sebagai negatif, maka disebut ***False Negative*** **(FN).** Sementara itu, jika data negatif diprediksi dengan tepat sebagai negatif, maka disebut ***True Negative*** **(TN),** dan jika data negatif diprediksi salah sebagai positif, maka disebut ***False Positive*** **(FP)** [9]**.**

Confusion matrix akan digunakan untuk evaluasi hasil prediksi klasemen *English Premier League* yang dibuat oleh algoritma *Naïve Bayes* dan *Random Forest*. Data dalam confusion matrix terdiri dari hasil prediksi model yang dihasilkan oleh kedua

algoritma *Machine Learning* tersebut serta data real yang merupakan hasil klasemen aktual dalam musim *English Premier League* yang dianalisis

1. Accuracy
   Menghitung rasio dari total prediksi yang benar terhadap semua prediksi.

$$\boldsymbol{Accuracy} = \frac{\boldsymbol{TP + TN}}{\boldsymbol{TP + FP + FN + TN}} \quad \textbf{(3)}$$

2. Precision
   Menghitung jumlah prediksi positif yang tepat dibandingkan dengan total keseluruhan prediksi positif yang dihasilkan.

$$\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP{=}FP}} \quad \textbf{(4)}$$

3. Recall
   Menghitung berapa banyak sampel positif yang dikenali oleh model dibandingkan dengan semua sampel positif.

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP{+}FN}} \quad \textbf{(5)}$$

4. FI-Score
   Menggabungkan presisi dan recall ke dalam satu metrik.

$$\boldsymbol{F1} = \frac{\boldsymbol{2\ x\ Recall\ x\ Precision}}{\boldsymbol{Recall{+}Precision}} \quad \textbf{(6)}$$

III. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

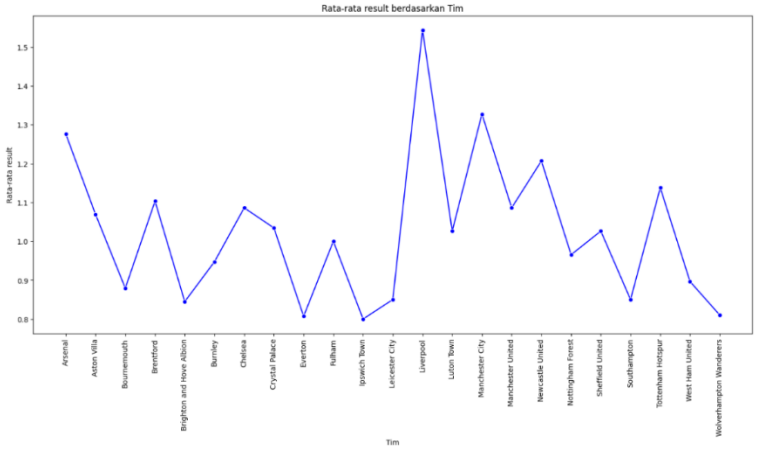
	Unnamed: 0	date	time	comp	round	day	venue	result	gf	ga	...	match report	notes	sh	sot	dist	fk	pk	pkatt	season	team
0	0	2024-08-17	12:30	Premier League	Matchweek 1	Sat	Away	W	2.0	0.0	...	Match Report	NaN	18.0	5.0	14.8	0.0	0	0	2022	Liverpool
1	1	2024-08-25	16:30	Premier League	Matchweek 2	Sun	Home	W	2.0	0.0	...	Match Report	NaN	19.0	8.0	13.6	1.0	0	0	2022	Liverpool
2	2	2024-09-01	16:00	Premier League	Matchweek 3	Sun	Away	W	3.0	0.0	...	Match Report	NaN	11.0	3.0	13.4	0.0	0	0	2022	Liverpool
3	3	2024-09-14	15:00	Premier League	Matchweek 4	Sat	Home	L	0.0	1.0	...	Match Report	NaN	14.0	5.0	14.9	0.0	0	0	2022	Liverpool
4	5	2024-09-21	15:00	Premier League	Matchweek 5	Sat	Home	W	3.0	0.0	...	Match Report	NaN	19.0	12.0	16.6	0.0	0	0	2022	Liverpool

Gambar 4. *Dataset* Terkumpul

Data yang dikumpulkan bersifat *historis* dan mencakup statistik pertandingan dari setiap tim dalam kompetisi *Premier League* musim tertentu. Informasi yang diperoleh mencakup tanggal dan waktu pertandingan, pekan pertandingan (*matchweek*), status kandang atau tandang (*venue*), hasil pertandingan (menang, seri, atau kalah), serta berbagai statistik performa tim seperti jumlah tembakan (*shots*), tembakan tepat sasaran (*shots on target*), rata-rata jarak tembakan (*shot distance*), dan statistik bola mati (tendangan bebas, penalti, dan percobaan penalti). Selain itu, data juga mencantumkan nama tim yang bermain dan musim kompetisi berlangsung. Sebanyak 1159 data statistik yang didapatkan setelah melakukan *scrapping web*. Data yang dikumpulkan bersifat historis dan mencakup statistik pertandingan untuk setiap tim dalam 3 musim (2022, 2023, 2024).

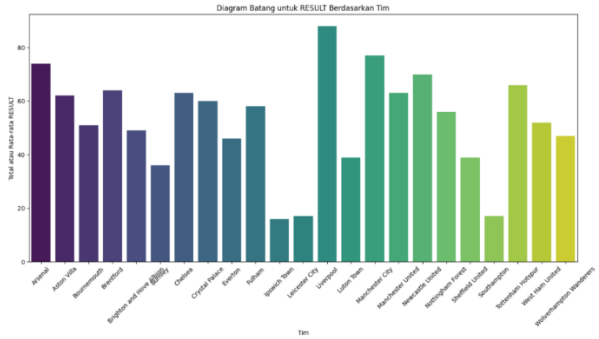
B. *Exploratory Data Analysis (EDA)*

Exploratory Data Analysis (EDA) dilakukan guna mengetahui karakteristik umum dari data yang telah dikumpulkan sebelum proses pelatihan model dilakukan. EDA bertujuan untuk mengidentifikasi pola, anomali, serta distribusi data agar proses pemodelan dapat lebih terarah dan akurat.



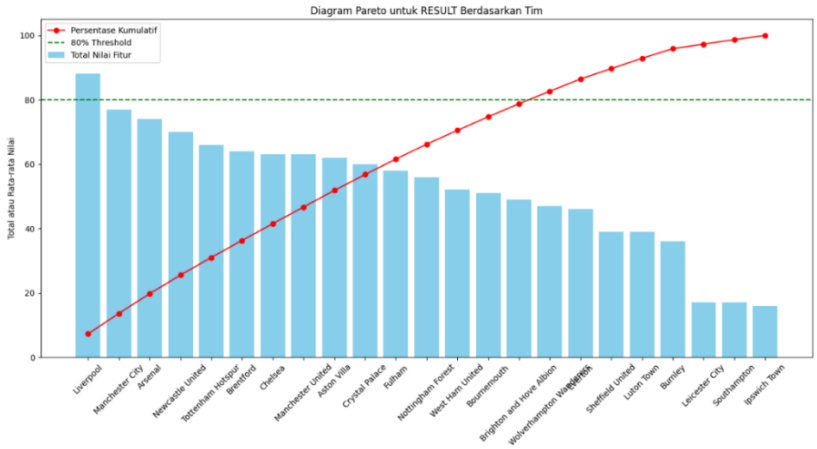
Gambar 5 Visualisasi *Line Plot*

Pada gambar 5, menunjukkan rata-rata nilai hasil pertandingan (*result*) untuk masing-masing tim dalam suatu kompetisi sepak bola. Grafik ini berbentuk *line plot* dengan sumbu horizontal *(x-axis*) menunjukkan nama-nama tim, dan sumbu vertikal (*y-axis*) menunjukkan rata-rata nilai "*result*".



Gambar 6. Visualisasi Diagram Batang

Gambar 6 merupakan diagram batang yang menampilkan total atau rata-rata hasil pertandingan (*result*) yang diperoleh oleh masing-masing tim dalam kompetisi *Premier League*. Visualisasi ini digunakan dalam tahap *Exploratory Data Analysis* (EDA) untuk melihat sejauh mana performa setiap tim berdasarkan hasil pertandingan yang dikategorikan sebagai menang (W), seri (D), atau kalah (L).



Gambar 7. Visualisasi Diagram Pareto

Pada gamba 7 merupakan *Diagram Pareto* yang digunakan untuk menganalisis distribusi hasil pertandingan (*result*) dari masing-masing tim dalam kompetisi *Premier League*. Diagram ini menggabungkan diagram batang (*bar chart*) dan garis kumulatif (*line chart*) untuk menunjukkan kontribusi relatif setiap tim terhadap total nilai hasil, serta untuk mengidentifikasi kelompok kecil yang memberikan dampak paling signifikan terhadap keseluruhan data.

C. Identifikasi *Missing Value*

Tahap identifikasi *missing values* merupakan langkah awal dalam proses pembersihan data (*data cleaning*). Tujuan tahap ini adalah untuk menemukan dan mengetahui kolom atau baris mana yang memiliki nilai kosong (*NaN/null*) sehingga dapat ditentukan tindakan penanganan yang sesuai.

```
Unnamed: 0          0
date                0
time                0
comp                0
round               0
day                 0
venue               0
result              0
gf                  0
ga                  0
opponent            0
xg                  0
xga                 0
poss                0
attendance          0
captain             0
formation           0
opp formation       0
referee             0
match report        0
notes            1158
sh                  0
sot                 0
dist                0
fk                  0
pk                  0
pkatt               0
season              0
team                0
dtype: int64
```

Gambar 8. Identifikasi *Missing Value*

Pada Gambar 8 merupakan hasil dari tahap identifikasi *missing values* dalam proses pembersihan data (*data cleaning*). Tahap identifikasi missing values ini menunjukkan bahwa dataset relatif bersih dan siap digunakan untuk proses pra-pemrosesan lanjutan dan pemodelan *machine learning*. Hanya kolom *notes* yang perlu dipertimbangkan penanganannya, sedangkan semua kolom penting lainnya tidak mengalami kehilangan data. Ini menjadi keuntungan karena mengurangi kebutuhan untuk teknik imputasi atau penghapusan baris, serta memastikan keakuratan model yang akan dibangun.

D. Seleksi Fitur

Setelah dilakukan eksplorasi data dan pembersihan data, tahap selanjutnya adalah melakukan seleksi fitur untuk menentukan fitur mana yang memiliki kontribusi signifikan terhadap prediksi hasil pertandingan. Proses seleksi fitur dilakukan dengan mempertimbangkan korelasi antar fitur, signifikansi statistik, serta relevansi fitur terhadap target.

Tabel 2. Seleksi Fitur

Data	Deskripsi
GF (Goal For)	Fitur
GA (Goal Against)	Fitur
xG	Fitur
xGA	Fitur
Poss	Fitur
Sh (Shoots)	Fitur
SoT (Shoots On Target)	Fitur
Dist	Fitur
FK	Fitur
PK (Penalty Kick)	Fitur
Result	Target

Melalui tahap seleksi fitur, hanya fitur-fitur yang relevan dan berkualitas tinggi yang digunakan dalam pemodelan. Ini penting untuk menentukan model yang dibangun tidak hanya akurat, tapi juga efisien dan mudah diinterpretasikan. Hasil dari tahap ini menjadi dasar dalam membangun dan melatih model menggunakan algoritma *Naïve Bayes* dan *Random Forest*.

E. *Encoding Variabel*

Untuk memudahkan pemrosesan, dilakukan Label *Encoding* pada variabel *result*. Nilai kategorikal dikonversi menjadi nilai numerik berdasarkan label tertentu, seperti tabel 3 berikut:

Tabel 3. Encoding Variabel

Atribut	Keterangan	Label
Lose	Kalah	0

Draw	Seri	1
Win	Menang	2

a. *Lose* (Kalah) diberi label **0**, menunjukkan tim kalah dalam pertandingan.

b. *Draw* (Seri) diberi label **1**, menunjukkan hasil pertandingan imbang.

c. *Win* (Menang) diberi label **2**, menunjukkan kemenangan tim.

Dengan dilakukannya encoding pada *variabel* kategorikal, dataset menjadi sepenuhnya numerik dan siap untuk digunakan dalam proses pelatihan model prediktif seperti *Naïve Bayes* dan *Random Forest*. Pemilihan metode encoding yang sesuai juga berkontribusi besar dalam menjaga kualitas representasi data dan meningkatkan performa model secara keseluruhan.

F. *Split Data*

Tabel 4. *Split Data*

Data	Keterangan	Persentase
Training	926	79.90%
Testing	232	20.03%
Total Data	1159	

Pada tabel 4 memperlihatkan bahwa dataset dibagi ke dalam dua kelompok, yaitu data pelatihan (*training data*) dan data pengujian (*testing data*), yang digunakan dalam tahap pelatihan serta pengujian model *machine learning*. Sebanyak 926 data atau sekitar 79,90% dari total *dataset* digunakan sebagai data *training*. Data ini berfungsi untuk membangun model dengan mengenali pola-pola dan hubungan antar fitur terhadap target variabel (*result*). Sementara itu, sebanyak 232 data atau sekitar 20,03% digunakan sebagai data *testing*.

G. Pengujian dan Evaluasi *Naïve Bayes*

Tabel 5. Hasil Model *Naïve Bayes*

Label	*Precision*	*Recall*	*F1-Score*	*Accuracy*
0	0.40	0.57	0.47	
1	0.84	0.84	0.84	69,83%
2	0.83	0.62	0.71	

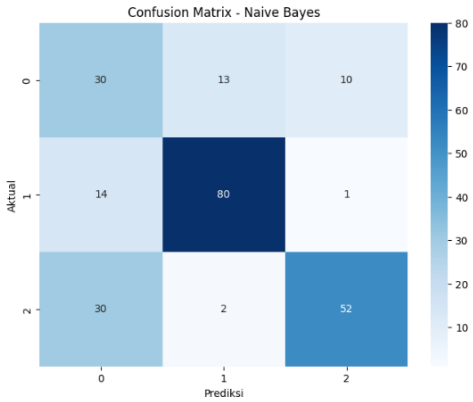
Berdasarkan tabel 5 pengujian terhadap data uji, model memperoleh tingkat akurasi sebesar 69,83%. Hal ini menunjukkan bahwa sekitar 70% dari hasil prediksi model sesuai dengan label yang sebenarnya. Dari hasil *classification report*, diketahui bahwa model memiliki performa terbaik dalam memprediksi label 1, yang kemungkinan merupakan kelas mayoritas. Hal ini ditunjukkan oleh nilai *precision* dan *recall* yang sama-sama tinggi, yaitu 0.84, sehingga menghasilkan *f1-score* sebesar 0.84**.** Di sisi lain, untuk kategori kelas 2, model juga menunjukkan kinerja yang cukup baik dengan nilai *precision* sebesar 0.83 dan *f1-score* sebesar 0.71, meskipun *recall*-nya lebih rendah yaitu 0.62**.** Sementara itu, performa terendah ditunjukkan pada prediksi untuk kelas **0**, dengan nilai *precision* hanya 0.41, *recall* 0.57, dan *f1-score* sebesar 0.47**,** yang mengindikasikan bahwa model masih sering salah dalam mengidentifikasi kelas ini.

Tabel 6. Rata- Rata *Naïve Bayes*

	Precision	*Recall*	*F1-Score*
Macro Avg	0.69	0.68	0.67
Weighted Avg	0.74	0.70	0.71

Pada tabel 6, secara keseluruhan, nilai *macro average* untuk *precision*, *recall*, dan *f1-score* masing-masing adalah 0.69, 0.68, dan 0.67, yang menunjukkan bahwa secara rata-rata, performa model pada setiap kelas masih cukup seimbang meskipun belum optimal. Sementara itu, nilai *weighted average* sedikit lebih tinggi, yaitu 0.74 untuk *precision*, 0.70 untuk *recall*, dan 0.71 untuk *f1-score*. Nilai rata-rata tertimbang ini mengindikasikan bahwa model cenderung memberikan prediksi yang lebih akurat pada kelas-kelas yang memiliki jumlah data lebih besar, karena perhitungannya mempertimbangkan proporsi sampel di setiap kelas.

ISSN: 2715-2766
Vol. 7 No. 1, Januari 2026



Gambar 9. *Confusion Matrix Naïve Bayes*

Berdasarkan gambar 9, hasil *confusion matrix* pada model *Naive Bayes*, diketahui bahwa untuk kelas 0, model berhasil memprediksi dengan benar sebanyak 30 data, namun masih terdapat 13 data yang salah diklasifikasikan sebagai kelas 1 dan 10 data sebagai kelas 2. Untuk kelas 1, performa model sangat baik dengan 80 data berhasil diklasifikasikan dengan benar, sementara 14 data salah diklasifikasikan sebagai kelas 0 dan 1 data sebagai kelas 2. Sedangkan untuk kelas 2, model berhasil mengklasifikasikan dengan benar sebanyak 52 data, namun terdapat 30 data yang diprediksi sebagai kelas 0 dan 2 data sebagai kelas 1. Dari hasil tersebut dapat disimpulkan bahwa performa model paling optimal berada pada kelas 1, yang ditunjukkan oleh tingginya jumlah klasifikasi benar dan tingkat kesalahan yang rendah, sedangkan untuk kelas 0 dan kelas 2, model masih menunjukkan kesulitan dalam membedakan keduanya, terlihat dari banyaknya prediksi silang di antara kedua kelas tersebut.

H. *Pengujian dan Evaluasi Random Forest*

Tabel 7. Hasil Model *Random Forest*

Label	*Precision*	*Recall*	*F1-Score*	*Accuracy*
0	1.00	0.98	0.99	
1	1.00	1.00	1.00	99,57%
2	0.99	1.00	0.99	

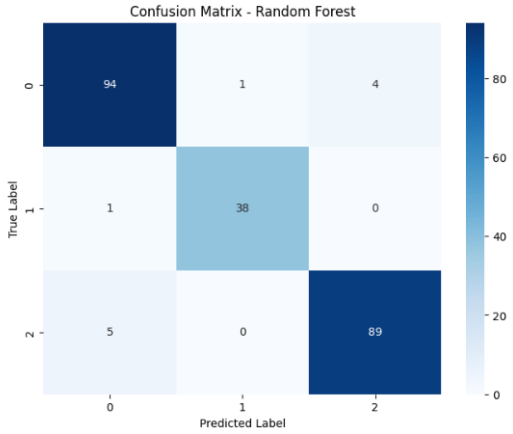
Pada table 7, hasil evaluasi memperlihatkan bahwa model memberikan performa yang sangat baik, dengan akurasi sebesar 99,57%. Ini berarti hampir seluruh prediksi model terhadap data uji sesuai dengan label aktual. Pada classification report, terlihat bahwa untuk kelas 0, precision mencapai 1.00, *recall* 0.98, dan *f1-score* sebesar 0.99, menunjukkan bahwa model sangat jarang salah dalam memprediksi kelas ini. Untuk kelas 1, model menunjukkan performa sempurna dengan *precision*, *recall*, dan *f1-score* semuanya sebesar 1.00, menandakan bahwa seluruh sampel pada kelas ini berhasil dikenali dengan tepat. Sementara itu, untuk kelas 2, model juga menunjukkan kinerja yang sangat baik dengan *precision* sebesar 0.99, *recall* 1.00, dan *f1-score* 0.99, menandakan bahwa hampir semua prediksi pada kelas ini tepat dengan sangat sedikit kesalahan.

Tabel 8. Rata-rata *Random Forest*

	Precision	*Recall*	*F1-Score*
Macro Avg	1.00	0.99	0.99
Weighted Avg	1.00	1.00	1.00

Berdasarkan tabel 8, skor rata-rata makro (*macro average*) untuk *precision*, *recall*, dan *f1-score* masing-masing bernilai 0,96, 0,96, dan 0,96. Sedangkan nilai rata-rata tertimbang (*weighted avg*) mencapai 0,95 untuk ketiga metrik tersebut. Hasil ini menunjukkan bahwa model *Random Forest* tidak hanya memiliki akurasi tinggi saja secara keseluruhan, namun juga mampu melakukan klasifikasi secara merata dan adil terhadap semua kelas, termasuk kelas minoritas.

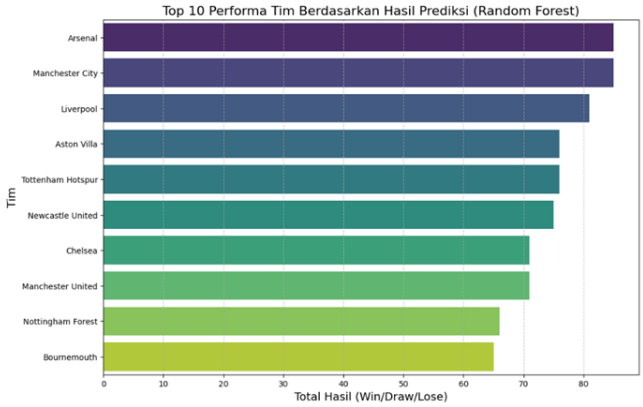
ISSN: 2715-2766
Vol. 7 No. 1, Januari 2026



Gambar 10. *Confusion Matrix Random Forest*

Berdasarkan gambar 10, terlihat bahwa model *Random Forest* mampu mengklasifikasikan data dengan sangat baik untuk ketiga kelas. Pada kelas 0 (kalah), dari total 53 data aktual, sebanyak 52 data berhasil diprediksi dengan benar, dan hanya 1 data yang salah diklasifikasikan sebagai menang (kelas 2), tanpa ada yang diprediksi sebagai imbang (kelas 1). Hal ini menunjukkan bahwa model memiliki kemampuan sangat baik dalam mengenali pola pada kelas kalah. Untuk kelas 1 (imbang), dari 95 data aktual, seluruhnya berhasil diprediksi dengan benar, yaitu sebanyak 95 data, tanpa kesalahan klasifikasi sama sekali ke kelas lainnya. Ini menandakan bahwa model sangat akurat dan konsisten dalam mengenali data imbang. Sementara itu, pada kelas 2 (menang), dari total 84 data, seluruh data juga diklasifikasikan dengan sangat baik dengan 84 data berhasil diprediksi dengan benar dan tidak ada kesalahan klasifikasi ke kelas lain. Secara keseluruhan, model menunjukkan performa klasifikasi yang sangat optimal dan presisi tinggi di semua kelas, dengan tingkat kesalahan yang nyaris nol, mencerminkan keandalan model dalam mengenali pola-pola dari data uji secara konsisten dan menyeluruh.

I. Hasil Klasemen



Gambar 11. Hasil Klasemen *Random Forest*

Hasil visualisasi menunjukkan 10 tim dengan performa terbaik berdasarkan hasil prediksi model *Random Forest* terhadap pertandingan sepak bola. Grafik bar *horizontal* memperlihatkan bahwa *Arsenal* menempati posisi teratas dengan jumlah hasil tertinggi (baik menang, imbang, maupun kalah), diikuti oleh *Manchester City* dan *Liverpool*. Ketiga tim ini menunjukkan performa yang paling dominan berdasarkan prediksi model. Secara keseluruhan, tim-tim papan atas seperti *Newcastle United, Aston Villa*, dan *Tottenham Hotspur* juga termasuk dalam daftar ini, menandakan bahwa model menganggap mereka tampil cukup konsisten. Di sisi lain, tim seperti *Manchester United* dan *Chelsea* berada di peringkat lebih bawah dalam 10 besar, yang menunjukkan bahwa performa mereka diprediksi sedikit lebih rendah dibandingkan tim-tim teratas

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, kesimpulan yang didapat dari penelitian ini adalah:

1. Untuk mengolah dan menganalisis data pertandingan *English Premier League* guna memprediksi hasil klasemen secara akurat, penelitian ini dilakukan melalui beberapa tahapan yang sistematis. Pertama, data pertandingan dikumpulkan dari sumber terpercaya yaitu situs *FBref.com*, yang mencakup berbagai statistik seperti jumlah kemenangan, hasil imbang, kekalahan, jumlah gol, tembakan, penguasaan bola, dan atribut lainnya yang relevan dengan performa tim. Selanjutnya, data yang diperoleh melalui proses *web scraping* ini dilakukan tahap pra-pemrosesan, yang meliputi pembersihan data (*data cleaning*), penanganan nilai yang hilang (*missing values*), *encoding* variabel kategorikal, serta seleksi fitur untuk menentukan atribut yang paling berpengaruh terhadap hasil pertandingan. Setelah data siap, dilakukan pemisahan antara data pelatihan

dan data pengujian dengan rasio 80:20, kemudian dibangun model prediksi menggunakan dua algoritma *machine learning*, yaitu *Naïve Bayes* dan *Random Forest*.

2. Algoritma *Naïve Bayes* menghasilkan akurasi 69,83% dalam memprediksi klasemen akhir *Premier League*. Namun, nilai *precision*, *recall*, dan *f1-score* masih rendah, terutama dalam memprediksi hasil imbang atau kalah. Hal ini menunjukkan bahwa model belum mampu membedakan beberapa kelas dengan baik. Salah satu penyebabnya adalah asumsi *Naïve Bayes* yang menganggap semua fitur saling bebas, padahal dalam sepak bola banyak fitur yang saling terkait. Sementara itu, algoritma *Random Forest* mendapatkan akurasi 99,57%, hasil sangat baik. Nilai *precision*, *recall*, dan *f1-score* juga tinggi dan seimbang, menandakan model mampu mengenali pola data dengan baik.
3. Visualisasi klasemen prediksi menunjukkan algoritma *Random Forest* lebih akurat dalam mengurutkan tim berdasarkan performa, menghasilkan hasil prediksi yang lebih mendekati kenyataan dibandingkan dengan *Naive Bayes*.

B. Saran

Saran yang dapat dijadikan acuan untuk pengembangan penelitian selanjutnya:

1. Penambahan Fitur

   Penelitian selanjutnya disarankan untuk menambahkan fitur-fitur yang lebih kompleks seperti performa pemain individu, kondisi cuaca, histori pertemuan antar tim, atau faktor non-teknis seperti cedera pemain dan jadwal pertandingan.

2. Penanganan Ketidakseimbangan Data

   Distribusi kelas hasil pertandingan (menang, seri, kalah) pada data dapat mempengaruhi performa model. Oleh karena itu, disarankan untuk melakukan *balancing dataset* melalui teknik seperti SMOTE (*Synthetic Minority Over-sampling Technique*) atau *undersampling* untuk meningkatkan akurasi model.

3. Eksplorasi Model Lain

   Di samping penggunaan *Naive Bayes* dan *Random Forest*, penelitian mendatang dapat diarahkan untuk mengevaluasi algoritma lain seperti *Support Vector Machine (SVM)*, *Gradient Boosting*, maupun metode *Deep Learning* seperti LSTM yang memiliki kemampuan dalam mengolah data berurutan.

Dengan memperhatikan beberapa saran tersebut, diharapkan hasil prediksi klasemen dapat lebih akurat dan aplikatif untuk keperluan analisis sepak bola atau sistem pendukung keputusan dalam konteks olahraga.

PENGAKUAN*(Heading 5)*

Makalah ini merupakan bagian dari penelitian Tugas Akhir penulis yang disusun sebagai kontribusi akademik dalam bidang visi komputer, dengan judul "Analisis Kinerja Algoritma *Naïve Bayes* dan *Random Forest* dalam Memprediksi Hasil Klasemen *English Premier League*". Penelitian ini dilakukan secara mandiri tanpa dukungan sponsor dari pihak mana pun, dan disusun sebagai bagian dari pemenuhan syarat akademik di Universitas Buana Perjuangan Karawang.

DAFTAR PUSTAKA

[1] M. N. Fauzan and M. N. Bawono, "Analisis Statistik Pertandingan Tim Nasional Sepak Bola Indonesia U-18 Di Piala Aff 2019," *J. Kesehat. Olahraga*, vol. 09, no. 03, pp. 371–380, 2021.

[2] E. Wheatcroft, "Forecasting football matches by predicting match statistics," *J. Sport. Anal.*, vol. 7, no. 2, pp. 77–97, 2021, doi: 10.3233/jsa-200462.

[3] Syahril Dwi Prasetyo, Shofa Shofiah Hilabi, and Fitri Nurapriani, "Analisis Sentimen Relokasi Ibukota Nusantara Menggunakan Algoritma Naïve Bayes dan KNN," *J. KomtekInfo*, vol. 10, pp. 1–7, 2023, doi: 10.35134/komtekinfo.v10i1.330.

[4] W. Winata, L. P. Dewi, and A. N. Tjondrowiguno, "Prediksi Skor Pertandingan Sepak Bola Menggunakan Neuroevolution of Augmenting Topologies dan Backpropagation," *J. Infra*, vol. 8, no. 1, 2020.

[5] R. A. Pratama, W. Apriandari, and D. Indrayana, "Prediksi Hasil Pertandingan Liga Serie A Menggunakan Metode Naïve Bayes," *J. SAINTIKOM (Jurnal Sains Manaj. Inform. dan Komputer)*, vol. 22, no. 2, p. 364, 2023, doi: 10.53513/jis.v22i2.8448.

[6] 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, pp. 377–342, 2023, [Online]. Available: http://www.football-data.co.uk.

[7] M. Ridho Handoko, "Sistem Pakar Diagnosa Penyakit Selama Kehamilan Menggunakan Metode Naive Bayes Berbasis Web," *J. Teknol. dan Sist. Inf.*, vol. 2, no. 1, pp. 50–58, 2021, [Online]. Available: http://jim.teknokrat.ac.id/index.php/JTSI

[8] C. Liu, Z. Gu, and J. Wang, "A Hybrid Intrusion Detection System Based on Scalable K-Means+ Random Forest and Deep Learning," *IEEE Access*, vol. 9, pp. 75729–75740, 2021, doi: 10.1109/ACCESS.2021.3082147.

[9] T. Nurmayanti, D. Hartini, T. Rohana, S. Arum, P. Lestari, and D. Wahiddin, "Comparison of K-Nearest Neighbors and Convolutional Neural Network Algorithms in Potato Leaf Disease Classification," *J. Sist. Inf. dan Ilmu Komput. Prima*, vol. 8, no. 1, pp. 360–372, 2024.