BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 *English Premier League*

Menurut sumber *id.wikipedia.org*, *English Premier League* adalah liga sepak bola profesional di Inggris dan tingkat tertinggi pada sistem liga sepak bola Inggris. Diikuti oleh 20 klub, liga ini beroperasi berdasarkan sistem promosi dan degradasi dengan *English Football League* (EFL). Liga ini menjadi liga olahraga dengan penonton terbanyak di dunia, yang disiarkan di 212 wilayah ke 643 juta pemirsa di rumah dan memiliki jumlah penonton potensial sebanyak 4,7 miliar. Pada musim 2023–2024, rata-rata jumlah penonton stadion pertandingan Liga Utama Inggris adalah 38.375 penonton, kedua setelah Bundesliga Jerman yang mencapai 39.512 penonton. Mayoritas stadion terisi hampir mendekati kapasitas penuhnya. Pada 2024–2025, Liga Utama Inggris menduduki peringkat pertama dalam peringkat koefisien UEFA berdasarkan penampilan pada kompetisi antarklub Eropa selama lima musim terakhir, mengungguli La Liga Spanyol.

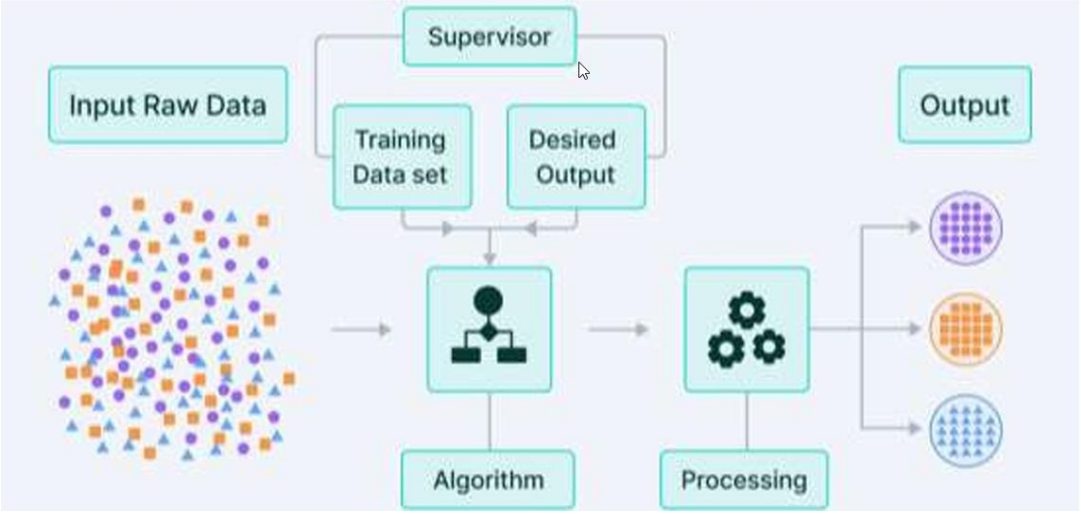
2.1.2 *Machine learning*

Machine Learning (ML) adalah sebuah aplikasi dari *Artificial Intelligence* (AI) atau kecerdasan buatan yang berfokus pada pengembangan sebuah system yang mampu belajar “sendiri” tanpa harus berulang kali di program oleh manusia (Prabowo 2020). *Machine Learning* dibagi menjadi beberapa kategori, di antaranya *supervised learning* (Geubbelmans dkk., 2023), *unsupervised learning* (Rousseau

dkk., 2023), dan *reinforcement learning* (Matsuo dkk. 2022). Dalam *supervised learning*, model dilatih menggunakan data yang memiliki label atau target yang sudah diketahui sehingga model belajar memetakan input menuju output tertentu. Metode ini banyak digunakan dalam permasalahan klasifikasi, termasuk prediksi hasil pertandingan sepak bola yang menjadi fokus penelitian ini. Sebaliknya, *unsupervised learning* digunakan untuk menemukan pola atau pengelompokan dalam data yang tidak memiliki label, sementara *reinforcement learning* berfokus pada proses pembelajaran berbasis umpan balik dari lingkungan melalui mekanisme *reward* dan *penalty*.

2.1.3 *Supervised Learning*

Supervised learning adalah metode dalam *machine learning* yang melibatkan dua tahap utama, yaitu pelatihan dan pengujian. Pada tahap pelatihan, model belajar dari data yang sudah memiliki label untuk mengenali pola dalam fitur-fitur yang diberikan. Setelah itu, model digunakan untuk memprediksi data baru pada tahap pengujian. Proses ini sering digunakan dalam jaringan saraf tiruan dan pohon keputusan, terutama untuk aplikasi yang memanfaatkan data historis guna memprediksi kejadian di masa depan (Syed dan Lokhande 2024).



Gambar 2.1 *Supervised Learning*

Pada gambar 2.1 *supervised learning* terbagi menjadi dua jenis utama, yaitu klasifikasi, yang digunakan untuk memprediksi kategori tertentu, dan regresi, yang digunakan untuk memprediksi nilai numerik. Setelah model dilatih, ia akan mencoba membuat prediksi berdasarkan pola yang telah dipelajari, tetapi jika ada informasi yang hilang, akurasi prediksi dapat menurun.

2.1.4 *Naïve Bayes*

Metode *Naïve Bayes* merupakan salah satu algoritma klasifikasi yang banyak digunakan dalam data mining. *Naïve Bayes* didasarkan pada prinsip probabilitas dan statistik yang dikembangkan oleh Thomas Bayes, yang memungkinkan prediksi kejadian di masa depan berdasarkan data historis (Putra dan Fatimah 2023). Seperti penelitian (Prabowo 2020), yang menggunakan *Naive Bayes* untuk memprediksi hasil pertandingan sepak bola berdasarkan berbagai variabel, seperti jumlah gol, statistik tim, riwayat pertemuan, dan variabel lainnya. Keunggulan metode ini terletak pada kesederhanaan, efisiensi perhitungan, dan kemampuannya dalam menangani dataset dengan jumlah terbatas, sehingga cocok

untuk klasifikasi berbasis probabilistik. Teorema tersebut ditunjukan oleh persaamaan 1.

$$P(A|B) = \frac{(\mathrm{P(B|A)xP(A)})}{\mathrm{P(B)}} \quad (1)$$

Keterangan :

P(A|B) : Probabilitas label A berdasarkan fitur B (*posterior probability)*

P(B|A) : Probabilitas fitur B berdasarkan label A (*likelihood*)

P(A) : Probabilitas label A (*prior probability)*

P(B) : Probabilitas fitur B (*evidence)*

Dalam penelitian ini, terdapat tiga label kelas untuk variabel "FTR" (*Full Time Result*), yaitu "H" yang menunjukkan kemenangan tim tuan rumah, "A" untuk kemenangan tim tamu, dan "D" untuk hasil imbang dan semuanya dalam bentuk presentase. Sementara itu, faktor-faktor yang digunakan dalam penelitian ini berasal dari data statistik pertandingan yang umum ditemukan dalam suatu laga. Statistik ini dipilih karena mampu merepresentasikan jalannya pertandingan serta menjadi indikator objektif dalam menilai performa tim yang bertanding. Faktor-faktor yang dipilih adalah *Home Team, Away Team, Home Shot, Away Shot, Home Shot on Target, Away Shot on Target, Home Corner, Away Corner, Home Red, Away Red.*

$$\mathrm{P(H|}X_1, X_2, \ldots\ldots X_{16}) = \frac{\mathrm{P}(X_1 X_1, X_2, \ldots\ldots X_{10}|\mathrm{H}) \times \mathrm{P(H)}}{\mathrm{P}(X_1, X_2, \ldots\ldots X_{16})} \quad (2)$$

Rumus diatas digunakan untuk menghitung probabilitas bahwa pertandingan akan berakhir dengan kemenangan tim tuan rumah, di mana nilai tersebut diperoleh dengan mengalikan probabilitas kemunculan fitur-fitur pertandingan apabila hasilnya adalah kemenangan tim home dengan probabilitas dasar kemenangan home yang diperoleh dari data historis, lalu dibagi dengan probabilitas keseluruhan fitur.

$$\mathrm{P}(\mathrm{A}|X_1, X_2, \dots \dots X_{16}) = \frac{\mathrm{P}(X_1 X_1, X_2, \dots \dots X_{10}|\mathrm{H}) \times \mathrm{P}(\mathrm{A})}{\mathrm{P}(X_1, X_2, \dots \dots X_{16})} \qquad (3)$$

Rumus diatas menghitung probabilitas bahwa pertandingan akan dimenangkan oleh tim tamu, elalui proses yang sama, tetapi menggunakan probabilitas kemunculan fitur dalam kondisi hasil pertandingan kemenangan tim away serta probabilitas prior dari kemenangan tim away.

$$\mathrm{P}(\mathrm{D}|X_1, X_2, \dots \dots X_{16}) = \frac{\mathrm{P}(X_1 X_1, X_2, \dots \dots X_{10}|\mathrm{H}) \times \mathrm{P}(\mathrm{D})}{\mathrm{P}(X_1, X_2, \dots \dots X_{16})} \qquad (4)$$

Rumus diatas digunakan untuk menghitung probabilitas pertandingan berakhir seri, dengan mempertimbangkan peluang kemunculan fitur apabila pertandingan berakhir imbang serta probabilitas prior dari hasil tersebut.

Ketiga probabilitas posterior ini kemudian dibandingkan untuk menentukan kelas hasil pertandingan yang paling mungkin. Dengan demikian, ketiga rumus tersebut berfungsi sebagai mekanisme penghitungan probabilistik yang

memungkinkan model *Naïve Bayes* melakukan prediksi hasil pertandingan berdasarkan pola-pola yang ditemukan dari data historis.

Keterangan:

X_1 : *AS (Away Shot)*

X_2 : *HS (Home Shot)*

X_3 : *HST (Home Shot on Target)*

X_4 : *AST (Away Shot on Target)*

X_5 : *ht_goal_diff (Selisih Gol HT)*

X_6 : *shot_ratio (Rasio Tembakan)*

X_7 : *HTHG (Home Half-Time Goals)*

X_8 : *HTAG (Away Half-Time Goals)*

X_9 : *AF (Away Fouls)*

X_{10} : *HC (Home Corner)*

X_{11} : *AC (Away Corner)*

X_{12} : *HF (Home Fouls)*

X_{13} : *HY (Home Yellow Cards)*

X_{14} : *AY (Away Yellow Cards)*

X_{15} : *HR (Home Red Cards)*

X_{16} : *AR (Away Red Cards)*

Dalam implementasinya, *Naïve Bayes* memiliki beberapa varian, dan salah satu yang paling relevan untuk penelitian ini adalah *Gaussian Naïve Bayes* . *Gaussian Naïve Bayes* digunakan ketika variabel-variabel prediktor berbentuk data

numerik kontinu, seperti jumlah tembakan ke gawang, jumlah tendangan pojok, kartu merah, atau statistik performa lainnya. Varian ini mengasumsikan bahwa setiap fitur mengikuti distribusi Gaussian atau distribusi normal, yaitu distribusi probabilitas berbentuk kurva lonceng yang simetris terhadap nilai rata-rata.

Asumsi Gaussian ini memudahkan proses penghitungannya karena peluang kemunculan setiap nilai fitur $X_i$untuk suatu kelas tertentu dapat dihitung menggunakan fungsi densitas Gaussian ditunjukan pada persamaan 5.

$$\boldsymbol{P(X_i \mid C) = \frac{1}{2\pi\sigma_C^2} exp(-\frac{(Xi - \mu C)^2}{2\sigma_C^2})} \quad (5)$$

Keterangan:

μC : rata-rata nilai fitur pada kelas C

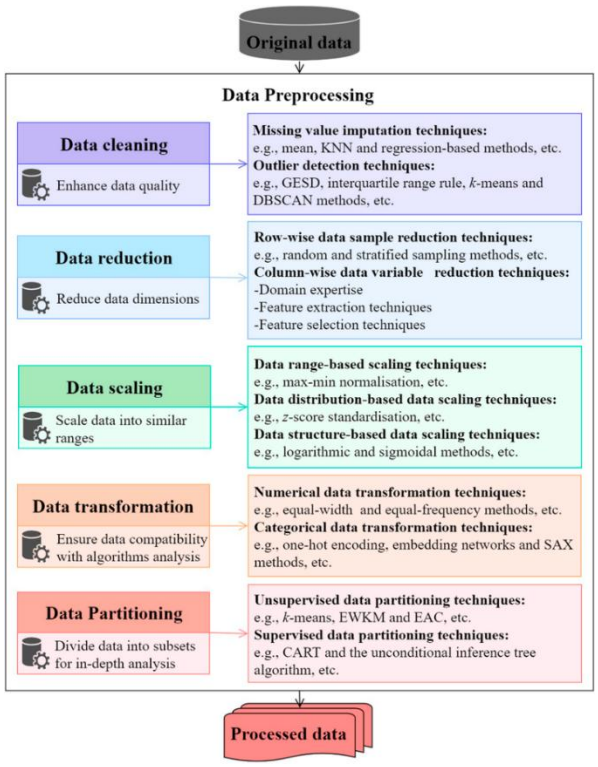
σC : standar deviasi nilai fitur pada kelas C

Penerapan *Gaussian Naïve Bayes* sangat sesuai dalam penelitian prediksi hasil pertandingan sepak bola karena sebagian besar fitur berupa angka kontinu dan cenderung berdistribusi mendekati normal. Maka, *Gaussian Naïve Bayes* mampu memodelkan hubungan probabilistik antara statistik pertandingan dan hasil akhirnya secara lebih akurat. Selain itu, varian Gaussian tetap mempertahankan efisiensi perhitungan sebagaimana karakteristik utama *Naïve Bayes* , sehingga dapat bekerja dengan baik meskipun dataset tidak terlalu besar.

2.1.5 *Data Preprocessing*

Data preprocessing merupakan tahap awal dalam pengolahan data yang bertujuan untuk memastikan kualitas data sebelum diproses lebih lanjut. Pada tahap

ini, data yang akan digunakan difilter agar terhindar dari gangguan seperti *noise* atau ketidakkonsistenan yang dapat mempengaruhi hasil analisis (Alghifari dan Juardi 2021). *Data propecessing* memiliki berberapa tahapan seperti pada gambar 2.2.



Gambar 2.2 *Data Propecessing*

a. *Data cleaning* adalah proses untuk mendeteksi dan memperbaiki kesalahan dalam data, seperti data yang hilang, duplikat, atau tidak valid
b. *Data reduction* adalah proses untuk mengurangi ukuran dataset tanpa menghilangkan nilai pentingnya
c. *Data scaling* adalah proses untuk menyesuaikan skala atau nilai fitur dalam dataset agar lebih seimbang mudah diproses oleh model *machine learning*.
d. *Data transformation* adalah proses untuk mengubah format, struktur, atau nilai data agar lebih sesuai untuk analisis.

e. *Data partitioning* adalah proses membagi dataset keberbagai bagian yang bertujuan untuk pelatihan, validasi, dan pengujian dalam model *machine learning.*

2.1.6 *Feature Engineering*

Feature engineering biasanya diterapkan setelah pengumpulan dan pembersihan data setelah diinputkan. Penelitian (Chinta dkk. 2025), menggunakan *feature engineering* untuk melakukan deteksi serangan phising melalui email. Sementara itu, penelitian (Esenogho dkk. 2022) menggunakan teknik *feature engineering* untuk melakukan penelitian terhadap sistem deteksi penipuan kartu kredit. Sehingga dapat disimpulkan, bahwa *feature engineering* bertujuan untuk merancang fitur cerdas dengan salah satu dari dua cara yang memungkinkan, dengan menyesuaikan fitur yang ada menggunakan berbagai transformasi atau dengan melakukan ekstraksi dan membuat fitur baru yang bermakna dari berbagai sumber.

2.1.7 *Cross-Validation*

Cross-validation, atau yang juga dikenal sebagai estimasi rotasi, merupakan teknik validasi model yang digunakan untuk menilai sejauh mana hasil analisis statistik dapat digeneralisasi ke kumpulan data yang independen. Metode ini terutama dimanfaatkan dalam pembuatan model prediktif untuk mengukur tingkat akurasi model ketika diterapkan dalam praktik nyata (Azis dkk. 2020). Teknik ini membagi data menjadi beberapa subset untuk pelatihan dan pengujian secara bergantian, sehingga dapat mengurangi risiko *overfitting* dan memberikan estimasi

performa model yang lebih akurat. Dengan demikian, *cross-validation* membantu dalam memilih model terbaik yang memiliki keseimbangan antara bias dan varian, sehingga model dapat bekerja dengan baik pada data baru yang belum pernah dilihat sebelumnya.

2.1.8 *Confusion Matrix*

Confusion Matrix adalah tabel yang menunjukkan jumlah data uji yang diklasifikasikan dengan benar maupun salah, sehingga mempermudah penilaian akurasi suatu sistem klasifikasi (Nurhidayat dan Dewi 2023). Tabel ini berisi informasi mengenai jumlah prediksi yang benar dan kesalahan yang dilakukan oleh model, yang biasanya dibagi ke dalam kategori *True Positive*, *True Negative*, *False Positive*, dan *False Negative* untuk memberikan gambaran lebih jelas tentang kinerja klasifikasi seperti pada tabel 2.1.

Tabel 2.1 *Confusion Matrix*

	Prediksi	
Aktual	*True*	*False*
True	TP	FP
False	FN	TN

Dimana:

TP : *True Positive* yaitu data positif dan diprediksi benar

TN : *True Negative* yaitu data negatif dan diprediksi benar

FP : *False Positive* yaitu data negatif tetapi diprediksi sebagai data positif

FN : *False Negative* yaitu data positif tetapi diprediksi sebagai data *negative*

Pada penelitian ini menggunakan 3 label, sehingga Tabel 2.1 berubah menjadi sesuai label klasifikasi yang ditunjukan pada Tabel 2.2

Tabel 2.2 *Confusion Matrix 3x3*

		Predict Label		
		Label D	Label H	Label A
Actual Label	Label D	DD	DH	DA
	Label H	HD	HH	HA
	Label A	AD	AH	AA

2.1.9 *Akurasi dan Error*

Akurasi mengukur seberapa dekat nilai prediksi dengan nilai aktual. Nilai akurasi dapat dihitung menggunakan persamaan 6.

$$Akurasi = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (6)$$

Pada penelitian ini label yang digunakan berjumlah 3 maka *Confusion matrix* menjadi 3x3. Seperti persamaan 7.

$$Akurasi = \frac{(DD + HH + AA)}{(DD + DH + DA + HD + HH + HA + AD + AH + AA)} \quad (7)$$

Error merupakan jumlah seluruh data prediksi yang salah dan dibagi dengan keseluruhan data prediksi. Hasil *error* dapat dihitung dalam persamaan 8.

$$Error = \frac{(FP + FN)}{(TP + TN + FP + FN)} \quad (8)$$

Rumus diatas merupakan definisi dasar dari *error rate* dalam sistem klasifikasi, yaitu ukuran sejauh mana model melakukan kesalahan dalam melakukan prediksi. Nilai error dihitung dengan membandingkan jumlah kesalahan prediksi, yang terdiri dari *false positive* (FP) dan *false negative* (FN) terhadap keseluruhan jumlah data yang diuji. Maka, semakin besar nilai FP dan FN, semakin tinggi tingkat kesalahan model. Sebaliknya, jika nilai FP dan FN rendah, maka model memiliki kinerja yang lebih baik. Rumus ini menunjukkan bahwa error merupakan proporsi dari prediksi yang tidak tepat terhadap total seluruh prediksi yang dievaluasi, yang meliputi *true positive* (TP), *true negative* (TN), FP, dan FN.

$$Error = \frac{(DH + DA + HD + HA + AD + AH)}{(DD + DH + DA + HD + HH + HA + AD + AH + AA} \quad (9)$$

Sementara itu, rumus diatas merupakan bentuk turunan dari error rate yang disesuaikan untuk kasus klasifikasi tiga kelas pada prediksi hasil pertandingan, yaitu Home (H), Draw (D), dan Away (A). Pada rumus ini, DH, DA, HD, HA, AD, dan AH merupakan representasi dari seluruh kesalahan prediksi ketika model memprediksi satu kelas tetapi kenyataannya hasil berada pada kelas lain. Misalnya, DH berarti model memprediksi Draw, tetapi hasil sebenarnya adalah Home, dan demikian seterusnya untuk kombinasi lainnya. Semua nilai ini dijumlahkan sebagai total kesalahan prediksi. Sementara itu, penyebutnya terdiri dari keseluruhan hasil

prediksi untuk semua kelas—baik prediksi yang benar seperti DD, HH, AA maupun prediksi yang salah. Dengan demikian, rumus ini menghitung proporsi kesalahan dalam konteks klasifikasi multikelas, di mana model harus membedakan tiga kemungkinan hasil pertandingan. Rumus tersebut memberikan gambaran menyeluruh mengenai tingkat kesalahan model dalam memprediksi hasil pertandingan berdasarkan data historis.

2.1.10 *Presisi dan Recall*

Presisi dan recall merupakan dua metrik evaluasi penting dalam sistem klasifikasi yang digunakan untuk mengukur kualitas prediksi model terhadap masing-masing kelas. Presisi mengukur tingkat ketepatan prediksi positif, yaitu sejauh mana data yang diprediksi sebagai positif benar-benar merupakan data positif. Nilai presisi dihitung berdasarkan perbandingan antara jumlah *true positive* dengan jumlah keseluruhan prediksi positif sesuai persamaan 10.

$$Presisi = \frac{TP}{(TP + FP)} \tag{10}$$

Dalam konteks penelitian ini, setiap kelas (D = *Draw*, H = *Home Win*, A = *Away Win*) menghasilkan prediksi positif masing-masing. Oleh karena itu, presisi tidak cukup dihitung satu kali, tetapi perlu dihitung untuk setiap label secara terpisah. Berdasarkan Tabel 2.2, presisi untuk label ke-i dapat dinyatakan dalam persamaan 11.

$$Presisi = \frac{Di}{Di + Hi + Ai} \tag{11}$$

Setelah memperoleh presisi untuk setiap kelas, nilai presisi total dihitung menggunakan rata-rata dari ketiga presisi kelas, sebagaimana ditunjukkan pada persamaan 12.

$$Presisi = \frac{Presisi\ D + Presisi\ H + Presisi\ A}{3} \quad (12)$$

Sementara itu, recall mengukur kemampuan model dalam menemukan seluruh data positif yang sebenarnya ada pada setiap kelas. Recall dihitung dengan membandingkan jumlah true positive terhadap total data positif sebenarnya, sesuai dengan persamaan 13.

$$Recall = \frac{(TP)}{(TP + FN)} \quad (13)$$

Sama seperti presisi, perhitungan recall pada penelitian ini juga dilakukan untuk masing-masing label, karena setiap kelas memiliki prediksi positif sendiri. Mengacu pada Tabel 2.2, recall untuk label ke-i dirumuskan dalam persamaan 14.

$$Recall = \frac{iD}{iD + iH + iA} \quad (14)$$

Nilai recall keseluruhan diperoleh dengan menghitung rata-rata recall dari ketiga kelas, sebagaimana ditunjukkan dalam persamaan 15.

$$Recall = \frac{Recall\ D + Recall\ H + Recall\ A}{3} \quad (15)$$

Maka demikian, penggabungan presisi dan recall dalam satu subbab memberikan gambaran yang lebih komprehensif mengenai kinerja model, khususnya dalam konteks klasifikasi tiga kelas hasil pertandingan sepak bola.

2.1.11 *Feature Importance*

Feature importance merupakan konsep yang digunakan untuk mengidentifikasi tingkat kontribusi masing-masing fitur terhadap hasil prediksi model. Setelah model dibangun dan dievaluasi melalui metrik seperti akurasi, presisi, dan *recall*, langkah selanjutnya adalah memahami variabel mana yang memiliki pengaruh terbesar dalam pembentukan keputusan model. Penelitian (Pathak dkk., 2025), melakukan penelitian terkait *malware* pada perangkat andorid, dengan menggunakan *feature selection* berdasarkan skor yang diperoleh dari *feature importance* yang sudah ditentukan. Sehingga, penerapan *feature importance* membantu peneliti memahami variabel yang paling dominan memengaruhi prediksi, mengurangi fitur yang kurang relevan, serta meningkatkan efisiensi komputasi dengan menjaga hanya fitur yang memberikan dampak signifikan terhadap performa model.

2.1.12 *State of The Art*

Berdasarkan rumusan masalah dan tujuan penelitian yang telah dibuat, maka dilakukan penyusunan literature review dari penelitian sebelumnya yang berkaitan dengan topik penggunaan *machine learning* untuk melakukan prediksi hasil pertandingan sepakbola. Beberapa penelitian sebelumnya yang dilakukan adalah sebagai berikut:

Tabel 2.3 *State of The Art*

No	Peneliti	Judul	Objek Penelitian	Algoritma	Hasil Penelitian
1.	(Prabowo, 2020)	Prediksi Hasil Pertandingan Sepakbola *English Premier League* dengan Menggunakan Algoritma *K-Nearest Neighbor* dan *Naive Bayes Classifier*	Prediksi Hasil Pertandingan Liga Inggris	K *K-Nearest Neighbor, Naive Bayes*	Hasil dari pengujian yang dilaukan sebanyak 3 sekanrio menghasilkan benar 345 dari 570 data uji atau akurasinya 60,5% dan eror 39,5%, skenario 2 menghasilkan benar 276 dari 456 data uji atau akurasinya 60,5% dan eror 39,5%, skenario 3 menghasilkan benar 145 dari 228

No	Peneliti	Judul	Objek Penelitian	Algoritma	Hasil Penelitian
					data uji atau akurasinya 63,5% dan eror 36,5%. Hasil akurasi yang berbeda pada ketiga skenario menunjukan bahwa semakin banyak data latih pada model maka akan meningkatkan nilai akurasi
2.	(Karim, Prasetyo, dan Saputro 2023)	Perbandingan Metode *Random Forest*, *K-Nearest Neighbor*, dan SVM Dalam Prediksi Akurasi Pertandingan Liga Italia	Prediksi Akurasi Pertandingan Liga Italia	*Random Forest, K-Nearest Neighbor, SVM*	Hasil penelitian ini menggunakan metode *Random forest* dengan hasil akurasi sebesar 62%, metode SVM menghasilkan akurasi sebesar 64%, dan metode KNN menghasilkan akurasi sebesar 57%.

No	Peneliti	Judul	Objek Penelitian	Algoritma	Hasil Penelitian
3.	(Saputro dkk. 2024)	Perbandingan Metode *Adaptive Boosting* dan *Extreme Gradient Boosting* Untuk Prediksi Hasil Pertandingan Liga Spanyol	Prediksi Hasil Pertandingan Liga Spanyol	*Adaptive Boosting, Extreme Gradient Boosting*	Pengujian ini menggunakan metode *AdaBoost* dengan hasil akurasi sebesar 64,02% dan metode *XGBoost* sebesar 61,79%
4.	(A. F. Pratama dkk. 2023)	Prediksi Hasil Pertandingan Sepakbola Menggunakan Metode FNN dan LSTM	Prediksi Hasil Pertandingan Sepak Bola	*Feedfoward Neural Network, Long Short Term Memory.*	Evaluasi penelitian ini menunjukan bahwa metode FNN berhasil mencapai tingkat keberhasilan sebesar 84%, dan LSTM mencapai sebesar 76%

No	Peneliti	Judul	Objek Penelitian	Algoritma	Hasil Penelitian
5.	(Isriwanto 2022)	Analisis Kebutuhan Prediksi Pertandingan Bundesliga Menggunakan Metode *Fuzzy*	Prediksi Pertandingan Bundesliga	*Fuzzy*	Hasil dari analisis kebutuhan terdapat tujuh faktor yang mempengaruhi kemenangan yaitu kekuataan *squad*, klasemen musim-musim sebelumnya, produktivitas gol tim, pertahanan sebuah tim, taktik sebuah tim, kekompakan tim, serta *home* dan *away*.

No	Peneliti	Judul	Objek Penelitian	Algoritma	Hasil Penelitian
6.	(Zein dan Gunawan 2022)	Prediksi Hasil *FIFA World Cup Qatar* 2022 Menggunakan *Machine Learning* dengan *Phtyon*	Prediksi hasil *FIFA World Cup Qatar* 2022	*Logisctic Regression, KNN, Naïve Bayes , SVM, Neural Network, Random Forest*	Penelitian ini menganalisis keakuratan algoritma *machine learning* dalam memprediksi hasil Piala Dunia *FIFA*. Algoritma dengan akurasi tertinggi bervariasi setiap turnamen, dengan SVM memiliki rata-rata tertinggi (59,37%). Untuk prediksi Piala Dunia 2022, digunakan Neural Network jika kedua tim terakhir bertemu 12 tahun lalu, dan SVM jika bertemu 4 tahun lalu atau belum pernah bertemu.

No	Peneliti	Judul	Objek Penelitian	Algoritma	Hasil Penelitian
7.	(Putra dan Fatimah 2023)	Penerapan Algoritme *Naïve Bayes* Dalam Memprediksi Juara Liga Primer Inggris Musim 2022/2023	Prediksi Juara Liga Primer Inggris Musim 2022/2023	*Naïve Bayes*	Hasilnya adalah model diuji dengan pembagian data latih dan uji 7:3, menghasilkan akurasi 91,7%, presisi 84,2%, dan *recall* 88,9%, menunjukkan performa yang tinggi dalam prediksi.
8.	(Pinasthika dan Fudholi 2023)	*World Cup 2022 Knockout Stage Prediction Using Poisson Distribution Model*	*World Cup 2022 Knockout Stage Prediction*	*Distribution Poisson*	Penelitian ini menggunakan Model Distribusi *Poisson* untuk memprediksi babak *knockout* Piala Dunia 2022, dengan rata-rata gol sebagai parameter lambda. Hasilnya, model berhasil memprediksi 6 dari 8 laga 16 besar dan 2 dari 4

No	Peneliti	Judul	Objek Penelitian	Algoritma	Hasil Penelitian
					perempat final dengan lebih baik saat hanya menggunakan data fase grup
9.	(Putri 2023)	Penerapan Model *Naïve Bayes* untuk Memprediksi Potensi Pendafataran Siswa di SMK Taman Siswa Teluk Betung Berbasis Web	Prediksi Potensi Pendfataran Siswa di SMK	*Naïve Bayes*	Penelitian ini menghasilkan akurasi 87% dalam memprediksi potensi pendaftaran siswa di SMK Taman Siswa
10.	(Annisa dan Sasongko 2020)	Prediksi Nilai Akademik Mahasiswa Menggunakan Algoritma *Naïve Bayes*	Prediksi Nilai Akademik Mahasiswa	*Naïve Bayes*	Penelitian ini menggunakan *Naïve Bayes* untuk memprediksi nilai akademis mahasiswa Universitas BSI Wilayah 3. Hasilnya menunjukkan akurasi 96,24%, presisi 95,76%,

No	Peneliti	Judul	Objek Penelitian	Algoritma	Hasil Penelitian
					dan 100%, dengan faktor utama seperti kuliah sambil bekerja, jadwal kerja, dan waktu kuliah.

Beberapa penelitian mengadopsi metode probabilistik, seperti *Naïve Bayes* , yang digunakan dalam studi (Karim, Prasetyo, dan Saputro 2023) dan (Putri 2023). Pendekatan berbasis deep learning juga mulai diterapkan, seperti FNN dan LSTM dalam penelitian (A. F. Pratama dkk. 2023). Selain itu, teknik statistik seperti *Distribusi Poisson* dan *Fuzzy Logic* muncul dalam studi (Isriwanto 2022) dan (Pinasthika dan Fudholi 2023). Secara keseluruhan, penelitian terdahulu menunjukkan bahwa berbagai teknik *machine learning* dan statistik terus berkembang dan diterapkan dalam berbagai bidang untuk meningkatkan akurasi prediksi. Prediksi telah banyak Berbagai metode *machine learning* dan statistik telah digunakan untuk analisis dan prediksi, termasuk *Naïve Bayes* , *Random Forest*, SVM, FNN, LSTM, *Logistic Regression*, *Neural Network*, K-NN, *XGBoost, Distribution Poisson*, dan *Fuzzy Logic.* Seperti pada tabel 2.3 yang meringkas berbagai penelitian terkait dengan penelitian yang sedang dilakukan.

2.1.13 *Matriks Penelitian*

Untuk memperjelas posisi penelitian ini terhadap penelitian-penelitian sebelumnya, disusun matriks penelitian yang membandingkan fokus, pendekatan, dan keluaran dari masing-masing penelitian. Matriks ini berfungsi untuk mengidentifikasi kesenjangan (research gap) yang masih ada, sehingga dapat terlihat kontribusi baru yang diberikan oleh penelitian ini.

Tabel 2.4 Matriks Penelitian

NO.	Penulis	*Naïve Bayes*	Random Forest	SVM	FNN	LSTM	Logistic Regression	Neural Network	K-NN	AdaBoost	XGBoost	Distribution Poisson	Fuzzy
1.	(Prabowo 2020b)	✓	-	-	-	-	-	-	-	-	-	-	-
2.	(Karim, Prasetyo, dan Saputro 2023)	✓	-	-	-	-	-	-	-	-	-	-	-

NO.	Penulis	*Naïve Bayes*	Random Forest	SVM	FNN	LSTM	Logistic Regression	Neural Network	K-NN	AdaBoost	XGBoost	Distribution Poisson	Fuzzy
3.	(Saputro dkk. 2024)	-	-	-	-	-	-	-	-	✓	✓	-	-
4.	(A. F. Pratama dkk. 2023)	-	-	-	✓	✓	-	-	-	-	-	-	-
5.	(Isriwanto 2022)	-	-	-	-	-	-		-	-	-	-	✓
6.	(Zein dan Gunawan 2022)	✓	✓	✓	-	-	✓	✓	✓	-	-	-	-
7.	(Putra dan Fatimah 2023)	✓	-	-	-	-	-	-	-	-	-	-	-
8.	(Pinasthika dan Fudholi 2023)	-	-	-	-	-	-	-	-	-	-	✓	-

NO.	Penulis	*Naïve Bayes*	Random Forest	SVM	FNN	LSTM	Logistic Regression	Neural Network	K-NN	AdaBoost	XGBoost	Distribution Poisson	Fuzzy
9.	(Hidayat 2024)	✓	-	-	-	-	-	-	-	-	-	-	-
10.	(Putri 2023)	✓	-	-	-	-	-	-	-	-	-	-	-
11.	(Annisa dan Sasongko 2020)	✓	-	-	-	-	-	-	-	-	-	-	-
12.	Penelitian ini	✓	-	-	-	-	-	-	-	-	-	-	-

Berdasarkan matriks penelitian pada Tabel 2.4, terlihat bahwa algoritma *Naïve Bayes* merupakan metode yang paling banyak digunakan oleh penelitian terdahulu, sebagaimana ditunjukkan pada penelitian (Prabowo, 2020), (Karim dkk, 2023), (Putra dan Fatimah, 2023), (Hidayat, 2024), (Putri, 2023), serta (Annisa dan Sasongko, 2020). Dominasi penggunaan algoritma *Naïve Bayes* tersebut menunjukkan bahwa metode ini masih relevan dan terus dipertimbangkan dalam pengembangan model prediksi, terutama karena sifatnya yang sederhana dan efektif pada data berdimensi terbatas. Di sisi lain, beberapa penelitian lain memilih pendekatan berbeda, seperti penggunaan FNN dan LSTM pada penelitian (A. F. Pratama dkk. 2023), atau pemanfaatan *Adaboost* dan *XGBoost* oleh (Saputro dkk. 2024). Terdapat pula penelitian yang menggunakan metode statistik maupun pendekatan *fuzzy* seperti (Zein dan Gunawan 2022), (Pinasthika dan Fudholi 2023), serta (Isriwanto 2022). Variasi metode ini menunjukkan bahwa studi prediksi hasil pertandingan atau analisis serupa memiliki ruang yang luas untuk dieksplorasi dari sisi algoritmik.

Melalui pemetaan tersebut, dapat dilihat bahwa penelitian ini tetap berada dalam jalur yang sama dengan beberapa penelitian sebelumnya yang juga menggunakan *Naïve Bayes* , namun menawarkan kontribusi baru karena tidak hanya melakukan klasifikasi hasil pertandingan, tetapi juga menyajikan estimasi probabilitas untuk setiap kemungkinan hasil. Sementara penelitian-penelitian sebelumnya umumnya hanya fokus pada akurasi klasifikasi, penelitian ini memperluas fungsi *Naïve Bayes* sebagai model prediksi probabilistik yang memberikan informasi lebih kaya dan lebih bermanfaat bagi pengambilan

keputusan. Maka demikian, posisi penelitian ini mengisi celah yang belum banyak disentuh dalam penelitian terdahulu, yaitu penerapan *Naïve Bayes* untuk memprediksi peluang kemenangan, hasil imbang, dan kekalahan secara kuantitatif, sehingga mampu memberikan nilai tambah yang signifikan dibandingkan penelitian sebelumnya.