

Komparasi Kinerja Algoritma *Random Forest*, *Decision Tree*, *Naïve Bayes*, dan KNN dalam Prediksi Tingkat Depresi Mahasiswa Menggunakan *Student Depression Dataset*

Ismail Setiawan¹⁾, Ilham Fatah Yasin ²⁾, Yuninda Tri Desianti³⁾

^{1,2,3)}Program Studi Sistem Dan Teknologi Informasi , Fakultas Sains Dan Teknologi, Universitas
'Aisyiyah Surakarta

¹⁾ismailsetiawan@aiska-university.ac.id ✉, ²⁾ilhamfatah.students@aiska-university.ac.id, ³⁾
yunindadesianti.students@aiska-university.ac.id

ABSTRACT

Mental health, especially depression in college students, is an important issue that requires serious attention. This study aims to compare the performance of several classification algorithms in predicting the level of depression in college students using the Student Depression Dataset (SDD). The algorithms analyzed include Random Forest, Decision Tree, Naïve Bayes, and K-Nearest Neighbors (KNN). The methods used include evaluating model performance based on accuracy, Precision, Recall, and F1-score metrics. The results show that Random Forest has the best performance in balancing Precision and Recall, while Naïve Bayes is more suitable for applications that prioritize Precision. These findings provide new insights into the application of classification algorithms to mental health data and form the basis for recommendations for educational institutions in formulating more effective policies to support college students' mental health.

Keywords: college students' depression, classification algorithm, Random Forest, Decision Tree, Naïve Bayes, K-Nearest Neighbors.

ABSTRAK

Kesehatan mental, khususnya depresi pada mahasiswa, menjadi isu penting yang membutuhkan perhatian serius. Penelitian ini bertujuan untuk membandingkan kinerja beberapa algoritma klasifikasi dalam memprediksi tingkat depresi mahasiswa menggunakan *Student Depression Dataset* (SDD). Algoritma yang dianalisis meliputi *Random Forest*, *Decision Tree*, *Naïve Bayes*, dan *K-Nearest Neighbors* (KNN). Metode yang digunakan mencakup evaluasi performa model berdasarkan metrik akurasi, *Precision*, *Recall*, dan *F1-score*. Hasil penelitian menunjukkan bahwa *Random Forest* memiliki performa terbaik dalam menyeimbangkan antara *Precision* dan *Recall*, sementara *Naïve Bayes* lebih cocok untuk aplikasi yang memprioritaskan *Precision*. Temuan ini memberikan wawasan baru dalam penerapan algoritma klasifikasi untuk data kesehatan mental dan menjadi dasar rekomendasi bagi institusi pendidikan dalam menyusun kebijakan yang lebih efektif untuk mendukung kesehatan mental mahasiswa.

Kata kunci: depresi mahasiswa, algoritma klasifikasi, Random Forest, Decision Tree, Naïve Bayes, K-Nearest Neighbors.

I. PENDAHULUAN

Kesehatan mental, khususnya depresi pada mahasiswa, menjadi isu penting yang membutuhkan perhatian serius [1]. Tekanan akademik [2], kurangnya dukungan sosial [3], dan gaya hidup yang tidak sehat [4] sering kali menjadi faktor utama yang memicu depresi. Dalam era teknologi informasi, analisis data berbasis *machine learning* menjadi alat yang potensial untuk memahami dan memprediksi kondisi ini [5]. Salah satu pendekatan yang relevan adalah menggunakan algoritma klasifikasi untuk memprediksi tingkat depresi mahasiswa berdasarkan pola yang terdapat dalam data [6]. Dengan memanfaatkan dataset yang memiliki nilai polinomial, seperti *Student Depression Dataset (SDD)* (www.kaggle.com/datasets/hopesb/student-depression-dataset), analisis ini memungkinkan pengelompokan data berdasarkan berbagai faktor yang memengaruhi tingkat depresi.

Penelitian ini berfokus pada melakukan komparasi algoritma klasifikasi untuk mengevaluasi kinerja beberapa algoritma populer, yaitu *Random Forest* [7], *Decision Tree* [8], *Naïve Bayes* [9], *xgboost* dan *K-Nearest Neighbors (KNN)* [10]. Pemilihan algoritma ini didasarkan pada kelebihan masing-masing dalam menangani berbagai jenis dataset. Dengan pendekatan ini, diharapkan dapat diperoleh pemahaman yang mendalam mengenai algoritma yang paling sesuai untuk menganalisis SDD.

Setiap algoritma yang digunakan memiliki karakteristik dan kelebihan masing-masing. *Random Forest* [7] dikenal karena kemampuannya dalam mengatasi *overfitting* dengan menggabungkan prediksi dari banyak pohon keputusan, sehingga menghasilkan model yang stabil dan akurat. *Decision Tree* [8] menawarkan interpretasi yang mudah dipahami karena menghasilkan model berbentuk pohon yang menunjukkan jalur keputusan dari fitur ke hasil klasifikasi. *Naïve Bayes* unggul dalam menangani data kompleks dengan memanfaatkan pendekatan iteratif untuk memperbaiki kesalahan sebelumnya. Sementara itu, *K-Nearest Neighbors (KNN)* [11] menawarkan kesederhanaan implementasi dan kinerja yang baik pada dataset kecil dengan memanfaatkan kedekatan data untuk menentukan kategori berdasarkan tetangga terdekat.

Dalam penelitian ini, SDD digunakan sebagai studi kasus. Dataset ini mengandung variabel-variabel polinomial seperti tingkat stres akademik, dukungan sosial, dan kebiasaan hidup mahasiswa, yang menjadi penerapan algoritma klasifikasi untuk data kesehatan mental, tetapi juga menjadi dasar rekomendasi bagi institusi pendidikan dalam menyusun kebijakan yang lebih efektif untuk mendukung kesehatan mental mahasiswa.

II. METODE

Penelitian ini mengikuti metodologi CRISP-DM [12] (*Cross-Industry Standard Process for Data Mining*) yang merupakan kerangka kerja yang populer dalam analisis data dan pengolahan informasi berbasis *machine learning*. CRISP-DM terdiri dari enam tahap utama, yang meliputi pemahaman bisnis, pemahaman data, persiapan data, pemodelan, evaluasi, dan penerapan. Setiap tahap dalam metode CRISP-DM diterapkan secara sistematis dalam penelitian ini untuk menganalisis dan membandingkan algoritma klasifikasi dalam memprediksi tingkat depresi mahasiswa menggunakan SDD.

A. Pemahaman Bisnis

Pada tahap pertama, penelitian ini bertujuan untuk memahami permasalahan yang dihadapi oleh mahasiswa terkait kesehatan mental, khususnya depresi. Depresi pada mahasiswa dapat dipicu oleh banyak faktor, seperti tekanan akademik, kurangnya dukungan sosial, dan gaya hidup yang tidak sehat. Oleh karena itu, penting untuk mengembangkan model klasifikasi yang dapat memprediksi tingkat depresi mahasiswa berdasarkan berbagai faktor yang terdapat dalam dataset [13]. Hal ini akan membantu institusi pendidikan dalam mengambil langkah-langkah preventif dan memberikan dukungan yang tepat kepada mahasiswa yang berisiko.

B. Pemahaman Data

Pada tahap ini [14], peneliti mengumpulkan dan mempelajari SDD yang berisi data terkait mahasiswa, termasuk variabel-variabel polinomial seperti tingkat stres akademik, dukungan sosial, kebiasaan hidup, dan faktor lainnya yang memengaruhi kesehatan mental. Dataset ini digunakan untuk mengidentifikasi pola dan hubungan antara fitur-fitur yang ada dan tingkat depresi yang dialami oleh mahasiswa. Dalam analisis ini, peneliti berfokus pada variabel-variabel yang memiliki pengaruh signifikan terhadap depresi, yang akan digunakan sebagai input untuk model klasifikasi.

C. Persiapan Data

Pada tahap persiapan data [15], data yang diperoleh diproses dan dipersiapkan untuk analisis lebih lanjut. Hal ini mencakup pembersihan data untuk menghilangkan nilai yang hilang atau tidak relevan, serta transformasi data untuk memastikan bahwa data siap digunakan dalam model klasifikasi. Beberapa teknik pengolahan seperti normalisasi atau encoding dilakukan pada data agar model klasifikasi dapat memprosesnya dengan baik. Dalam hal ini, data polinomial akan diubah menjadi format yang dapat dimengerti oleh algoritma yang digunakan.

D. Pemodelan

Pada tahap ini [16], berbagai algoritma klasifikasi digunakan untuk membangun model prediksi, termasuk *Random Forest*, *Decision Tree*, *Naïve Bayes*, dan *K-Nearest Neighbors* (KNN). Setiap algoritma dipilih berdasarkan karakteristik dan kelebihan masing-masing. *Random Forest* digunakan karena kemampuannya dalam mengatasi *overfitting* dan menghasilkan model yang stabil, sementara *Decision Tree* memberikan interpretasi yang mudah dipahami. *Naïve Bayes* digunakan untuk menangani data kompleks, sedangkan *K-Nearest Neighbors* (KNN) diterapkan pada dataset kecil dengan memanfaatkan kedekatan antar data. Semua algoritma ini dilatih menggunakan SDD untuk memprediksi tingkat depresi mahasiswa.

E. Evaluasi

Setelah model dibangun [17], tahap evaluasi dilakukan untuk menilai kinerja setiap algoritma. Evaluasi dilakukan menggunakan beberapa metrik performa, yaitu *Accuracy*, *Precision*, *Recall*, dan *F1-score*. *Accuracy* [18][19] digunakan untuk mengukur persentase prediksi yang benar, sedangkan *Precision* [20] dan *Recall* [21] memberikan gambaran lebih mendalam tentang kualitas prediksi positif. *F1-score* [22] digunakan untuk memberikan keseimbangan antara *Precision* dan *Recall*. Berdasarkan hasil evaluasi ini, peneliti dapat menentukan algoritma klasifikasi yang paling efektif dan sesuai untuk memprediksi tingkat depresi mahasiswa pada dataset ini.

F. Penerapan

Pada tahap penerapan, hasil dari penelitian ini diharapkan dapat diterapkan dalam lingkungan pendidikan untuk mendukung kebijakan kesehatan mental yang lebih baik. Algoritma yang terpilih dapat digunakan untuk memprediksi dan mengidentifikasi mahasiswa yang berisiko mengalami depresi, sehingga pihak institusi pendidikan dapat memberikan dukungan yang lebih tepat sasaran. Selain itu, hasil penelitian ini dapat menjadi dasar bagi pengembangan model prediksi kesehatan mental di masa mendatang, dengan penerapan teknik *machine learning* yang lebih canggih dan data yang lebih komprehensif.

III. HASIL DAN PEMBAHASAN

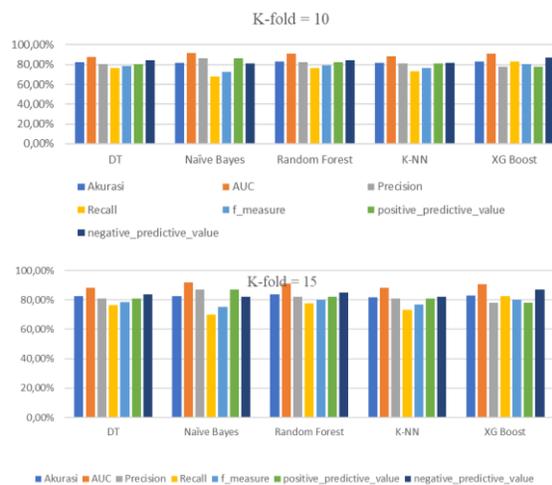
A. Perbandingan Model Klasifikasi

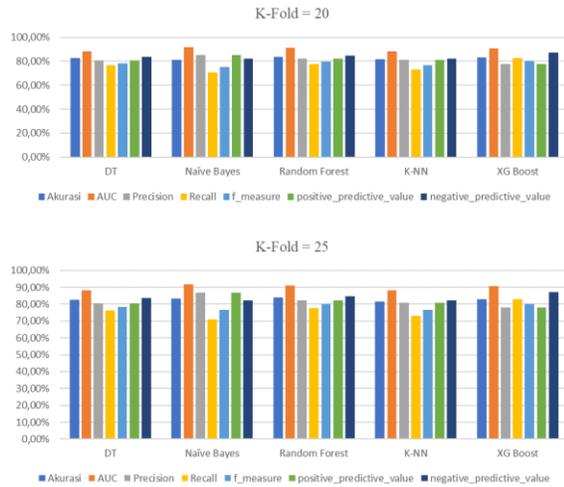
Sub bab ini, membahas perbandingan kinerja antara model klasifikasi yang digunakan dalam studi ini, yaitu *Decision Tree*, *Naïve Bayes*, *Random Forest*, *KNN*, dan *XG Boost*. Berdasarkan hasil eksperimen, kita dapat melihat bahwa setiap model memiliki kelebihan dan kelemahan uniknya masing-masing. Misalnya, *Random Forest* dan *XG Boost* secara konsisten menunjukkan akurasi dan F1 score yang tinggi, sekitar 83-84% dan 79-80% masing-masing, yang menandakan keseimbangan yang baik antara presisi dan *Recall*. Di sisi lain, *Naïve*

Bayes menonjol dengan nilai AUC tertinggi, sekitar 0.919, menandakan kemampuannya dalam membedakan antara kelas positif dan negatif. Namun, model ini juga memiliki *False negatives* yang cukup tinggi, menunjukkan bahwa beberapa kasus positif tidak terdeteksi dengan baik. Sementara itu, *KNN* dan *Decision Tree*, meskipun memiliki akurasi yang baik, menunjukkan variasi lebih besar dalam *Precision* dan *Recall*, menunjukkan potensi kelemahan dalam situasi tertentu. Dengan menganalisis metrik-metrik ini secara mendalam, kita dapat memahami konteks di mana masing-masing model mungkin paling efektif, serta mengidentifikasi potensi area perbaikan dalam implementasi praktis.

B. Interpretasi Metrik Kinerja

Penjelasan mengenai metrik kinerja adalah esensial untuk memahami performa dari setiap model klasifikasi. Akurasi, misalnya, mengukur persentase prediksi yang benar dari total prediksi yang dibuat oleh model, memberikan gambaran umum tentang keandalan model. *Area Under Curve* (AUC) memberikan informasi lebih mendalam tentang kemampuan model untuk membedakan antara kelas positif dan negatif, dengan nilai yang lebih tinggi menunjukkan kemampuan diskriminatif yang lebih baik. *Precision* mengukur seberapa tepat model dalam memprediksi kelas positif, sementara *Recall* mengukur kemampuan model dalam menangkap semua instance dari kelas positif. F1 score, sebagai kombinasi dari *Precision* dan *Recall*, memberikan keseimbangan antara keduanya dan sangat berguna dalam situasi di mana trade-off antara *Precision* dan *Recall* diperlukan. Pemahaman yang mendalam tentang metrik-metrik ini penting untuk evaluasi yang tepat dan pemilihan model yang sesuai dengan kebutuhan spesifik aplikasi atau penelitian.





Gambar 1. Metrik Kinerja 4 Algoritma

Berdasarkan hasil eksperimen di atas, berikut beberapa poin yang dapat diperhatikan:

1) Akurasi

Tabel 1. Kinerja Akurasi Model Prediksi

Fold	Decision Tree	Naive Bayes	Random Forest	KNN	XG Boost
10	82.76%	81.59%	83.44%	81.61%	83.03%
15	82.71%	82.71%	83.87%	81.73%	83.13%
20	82.60%	81.43%	83.82%	81.74%	83.07%
25	82.52%	83.28%	83.90%	81.68%	83.15%

Tabel di atas menunjukkan perbandingan akurasi berbagai model klasifikasi (*Decision Tree*, *Naive Bayes*, *Random Forest*, *KNN*, dan *XG Boost*) pada beberapa *fold* (10, 15, 20, dan 25). Dari hasil tersebut, kita dapat melihat bahwa *Random Forest* dan *XG Boost* menonjol dengan akurasi yang tinggi dan konsisten di semua *fold*, masing-masing berkisar antara 83.44% hingga 83.90% dan 83.03% hingga 83.15%. Keandalan dan stabilitas model ini menunjukkan bahwa mereka dapat diandalkan untuk menghasilkan prediksi yang akurat dalam berbagai kondisi. Sementara itu, *Decision Tree* dan *KNN* juga menunjukkan performa yang cukup stabil dengan sedikit fluktuasi dalam akurasi, meskipun tidak setinggi *Random Forest* dan *XG Boost*.

Naive Bayes, di sisi lain, menunjukkan variasi yang lebih besar dalam akurasinya, dengan nilai berkisar antara 81.43% hingga 83.28%. Model ini mencapai akurasi tertingginya pada *fold* 25, menunjukkan potensi yang dapat dieksplorasi lebih lanjut. Secara keseluruhan, *Random Forest* dan *XG Boost* dapat dianggap sebagai pilihan terbaik untuk dataset ini karena performa mereka yang tinggi dan stabil. Namun, penting juga untuk mempertimbangkan

konteks spesifik dan kebutuhan aplikasi saat memilih model, mengingat bahwa setiap model memiliki kelebihan dan kekurangannya masing-masing yang dapat mempengaruhi hasil dalam situasi tertentu.

2) Area Under Curve (AUC)

Tabel 2. Kinerja AUC Model Prediksi

Fold	10	15	20	25
Decision Tree	0.878	0.882	0.882	0.882
Naive Bayes	0.919	0.919	0.919	0.919
Random Forest	0.908	0.912	0.912	0.912
KNN	0.883	0.884	0.884	0.884
XG Boost	0.908	0.908	0.909	0.909

Berdasarkan hasil kinerja AUC pada tabel 2, dapat dilihat bahwa *Naive Bayes* secara konsisten menunjukkan performa terbaik dengan nilai AUC 0.919 pada semua *fold* (10, 15, 20, dan 25), menandakan kemampuan yang sangat baik dalam membedakan antara kelas positif dan negatif. *Random Forest* dan *XG Boost* juga menunjukkan performa yang sangat baik dengan nilai AUC yang tinggi dan konsisten, sekitar 0.908 hingga 0.912 untuk *Random Forest* dan 0.908 hingga 0.909 untuk *XG Boost*. Sementara itu, *KNN* dan *Decision Tree* memiliki nilai AUC yang lebih rendah dibandingkan *Naive Bayes*, *Random Forest*, dan *XG Boost*, tetapi tetap menunjukkan performa yang stabil di berbagai *fold* dengan nilai AUC sekitar 0.878 hingga 0.884. Konsistensi nilai AUC pada setiap model menunjukkan stabilitas performa masing-masing model dalam berbagai kondisi. Dengan demikian, *Naive Bayes* dapat dianggap sebagai model terbaik dalam hal kemampuan diskriminatif, diikuti oleh *Random Forest* dan *XG Boost* sebagai pilihan yang juga sangat andal.

3) Precision

Tabel 3. Kinerja Precision Model

Fold	10	15	20	25
Decision Tree	80.56%	80.53%	80.88%	80.70%
Naive Bayes	86.92%	85.32%	86.84%	86.35%
Random Forest	82.37%	82.26%	82.29%	82.17%
KNN	80.93%	81.05%	81.01%	80.85%
XG Boost	77.90%	77.83%	77.97%	77.71%

Berdasarkan tabel hasil kinerja *Precision* di atas, dapat kita lihat bahwa *Naive Bayes* secara konsisten menunjukkan *Precision* tertinggi di semua *fold*, dengan nilai sekitar 85.32% hingga 86.92%. Ini menunjukkan kemampuan *Naive Bayes* untuk membuat prediksi positif yang akurat sangat tinggi. *Random Forest* juga menunjukkan *Precision* yang tinggi dan konsisten, dengan nilai berkisar antara

82.17% hingga 82.37%. Meskipun tidak setinggi *Naïve Bayes*, *Random Forest* tetap menunjukkan performa yang sangat baik dalam prediksi positif. Sementara itu, *KNN* dan *Decision Tree* memiliki nilai *Precision* yang cukup stabil dan tinggi, dengan *KNN* berkisar antara 80.85% hingga 81.05% dan *Decision Tree* berkisar antara 80.53% hingga 80.88%.

Sebaliknya, *XG Boost* memiliki nilai *Precision* terendah di antara model-model lainnya, berkisar antara 77.71% hingga 77.97%. Meskipun demikian, *XG Boost* tetap menunjukkan stabilitas yang baik dalam hasil *Precision*-nya. Secara keseluruhan, *Naïve Bayes* menonjol sebagai model dengan *Precision* tertinggi, menunjukkan kemampuannya untuk membuat prediksi positif dengan akurasi yang sangat baik. *Random Forest* dan *KNN* juga menunjukkan performa yang baik dan konsisten, menjadikan mereka pilihan yang solid dalam prediksi positif. *Decision Tree* juga menunjukkan hasil yang cukup baik, sementara *XG Boost* meskipun lebih rendah dalam *Precision*, tetap menunjukkan stabilitas yang memadai.

4) Recall

Tabel 4. Kinerja Recall Model

Fold	10	15	20	25
<i>Decision Tree</i>	76.32%	76.55%	76.36%	76.80%
<i>Naïve Bayes</i>	70.91%	70.84%	69.81%	67.96%
<i>Random Forest</i>	77.87%	77.74%	77.86%	76.70%
<i>KNN</i>	73.00%	73.06%	73.04%	72.92%
<i>XG Boost</i>	82.92%	82.76%	82.71%	82.81%

Berdasarkan hasil kinerja *Recall* yang ditunjukkan dalam tabel di atas, kita dapat melihat bahwa *XG Boost* secara konsisten menunjukkan nilai *Recall* tertinggi pada semua *fold* (10, 15, 20, dan 25), dengan nilai berkisar antara 82.71% hingga 82.92%. Hal ini menunjukkan kemampuan model *XG Boost* dalam menangkap sebagian besar kasus positif, menjadikannya model yang sangat efektif dalam skenario di mana deteksi positif sangat penting. Di sisi lain, *Naïve Bayes* memiliki nilai *Recall* terendah, berkisar antara 67.96% hingga 70.91%, yang menunjukkan bahwa model ini mungkin tidak menangkap semua kasus positif dengan baik.

Decision Tree dan *Random Forest* menunjukkan nilai *Recall* yang cukup baik dan stabil, dengan *Decision Tree* berkisar antara 76.32% hingga 76.80% dan *Random Forest* berkisar antara 76.70% hingga 77.87%. *KNN*, meskipun lebih rendah dibandingkan dengan *Random Forest* dan *XG Boost*, tetap menunjukkan konsistensi dengan nilai *Recall* berkisar antara 72.92% hingga 73.06%. Secara keseluruhan, *XG Boost* unggul dalam kinerja *Recall*, diikuti oleh *Random Forest* dan *Decision Tree*,

sementara *Naïve Bayes* memiliki performa *Recall* yang lebih rendah.

5) F-Measure

Tabel 5. Kinerja F-Measure Model Prediksi

Fold	10	15	20	25
<i>Decision Tree</i>	78.34%	78.47%	78.54%	78.69%
<i>Naïve Bayes</i>	76.80%	75.07%	75.15%	72.84%
<i>Random Forest</i>	80.04%	79.93%	79.99%	79.33%
<i>KNN</i>	76.75%	76.83%	76.81%	76.67%
<i>XG Boost</i>	80.31%	80.21%	80.25%	80.16%

Berdasarkan hasil kinerja *F-Measure* pada tabel di atas, kita dapat melihat bahwa *XG Boost* secara konsisten menunjukkan nilai *F-Measure* tertinggi di antara model lainnya, berkisar antara 80.16% hingga 80.31% pada semua *fold* (10, 15, 20, dan 25). Hal ini menunjukkan bahwa *XG Boost* berhasil mencapai keseimbangan yang baik antara *Precision* dan *Recall*, menjadikannya model yang sangat efektif secara keseluruhan. *Random Forest* juga menunjukkan nilai *F-Measure* yang tinggi dan stabil, berkisar antara 79.33% hingga 80.04%, menunjukkan kinerja yang sangat baik dalam keseimbangan antara *Precision* dan *Recall*.

Di sisi lain, *Decision Tree* dan *KNN* menunjukkan performa *F-Measure* yang cukup stabil tetapi sedikit lebih rendah, dengan *Decision Tree* berkisar antara 78.34% hingga 78.69% dan *KNN* berkisar antara 76.67% hingga 76.83%. *Naïve Bayes* memiliki nilai *F-Measure* terendah di antara model lainnya, berkisar antara 72.84% hingga 76.80%, menunjukkan bahwa meskipun model ini memiliki *Precision* yang tinggi, *Recall*-nya yang lebih rendah mempengaruhi nilai *F-Measure* secara keseluruhan. Secara keseluruhan, *XG Boost* dan *Random Forest* adalah model yang paling seimbang dan efektif berdasarkan nilai *F-Measure*, dengan kemampuan untuk mencapai keseimbangan yang baik antara *Precision* dan *Recall* dalam berbagai kondisi.

6) False positives dan False negatives

Tabel 6. Kinerja False Positive Model

Fold	10	15	20	25
<i>D</i>	212.700 +/-	139.400 +/-	107.200 +/-	85.520 +/-
<i>ec</i>	20.325	13.399	10.700	11.057
<i>isi</i>	(Micro	(Micro	(Micro	(Micro
<i>on</i>	average:	average:	average:	average:
<i>Tr</i>	2127.000)	2091.000)	2144.000)	2138.000)
<i>ee</i>	(Positive	(Positive	(Positive	(Positive
	Class: No)	Class: No)	Class: No)	Class: No)
<i>N</i>	143.200 +/-	88.867 +/-	90.450 +/-	52.000 +/-
<i>ai</i>	101.504	41.409	132.417	21.297
<i>ve</i>	(Micro	(Micro	(Micro	(Micro
<i>Ba</i>	average:	average:	average:	average:
<i>ye</i>	1432.000)	1333.000)	1809.000)	1300.000)
<i>s</i>	(Positive	(Positive	(Positive	(Positive
	Class: No)	Class: No)	Class: No)	Class: No)

Ra	192.500 +/-	129.267 +/-	96.950 +/-	77.320 +/-
nd	12.528	11.016	5.586	10.209
o	(Micro	(Micro	(Micro	(Micro
m	average:	average:	average:	average:
Fo	1925.000)	1939.000)	1939.000)	1933.000)
re	(Positive	(Positive	(Positive	(Positive
st	Class: No)	Class: No)	Class: No)	Class: No)
	200.000 +/-	132.000 +/-	98.900 +/-	79.560 +/-
	18.391	7.672	8.944	6.995
K	(Micro	(Micro	(Micro	(Micro
N	average:	average:	average:	average:
N	2000.000)	1980.000)	1978.000)	1989.000)
	(Positive	(Positive	(Positive	(Positive
	Class: No)	Class: No)	Class: No)	Class: No)
	275.400 +/-	180.467 +/-	136.450 +/-	109.080 +/-
X	29.327	14.735	10.475	10.996
G	(Micro	(Micro	(Micro	(Micro
Bo	average:	average:	average:	average:
os	2754.000)	2707.000)	2729.000)	2727.000)
t	(Positive	(Positive	(Positive	(Positive
	Class: No)	Class: No)	Class: No)	Class: No)

Tabel 7. Kinerja False Negative Model

Fo	10	15	20	25
ld	268.300 +/-	182.267 +/-	135.600 +/-	109.560 +/-
D	17.321	10.586	10.952	12.939
ec	(Micro	(Micro	(Micro	(Micro
isi	average:	average:	average:	average:
on	2683.000)	2734.000)	2712.000)	2739.000)
Tr	(Positive	(Positive	(Positive	(Positive
ee	Class: No)	Class: No)	Class: No)	Class: No)
	370.500 +/-	232.733 +/-	168.600 +/-	134.560 +/-
N	249.857	138.269	94.164	62.389
ai	(Micro	(Micro	(Micro	(Micro
ve	average:	average:	average:	average:
Ba	3705.000)	3491.000)	3372.000)	3364.000)
ye	(Positive	(Positive	(Positive	(Positive
s	Class: No)	Class: No)	Class: No)	Class: No)
Ra	269.500 +/-	170.733 +/-	128.700 +/-	102.360 +/-
nd	18.952	17.754	10.658	8.990
o	(Micro	(Micro	(Micro	(Micro
m	average:	average:	average:	average:
Fo	2695.000)	2561.000)	2574.000)	2559.000)
re	(Positive	(Positive	(Positive	(Positive
st	Class: No)	Class: No)	Class: No)	Class: No)
	313.200 +/-	207.867 +/-	155.800 +/-	124.920 +/-
	13.139	16.733	11.928	10.614
	(Micro	(Micro	(Micro	(Micro
	average:	average:	average:	average:
K	3132.000)	3118.000)	3116.000)	3123.000)
N	(Positive	(Positive	(Positive	(Positive
N	Class: No)	Class: No)	Class: No)	Class: No)
	198.800 +/-	133.333 +/-	99.700 +/-	79.000 +/-
	14.883	14.652	10.887	8.761
X	(Micro	(Micro	(Micro	(Micro
G	average:	average:	average:	average:
Bo	1988.000)	2000.000)	1994.000)	1975.000)
os	(Positive	(Positive	(Positive	(Positive
t	Class: No)	Class: No)	Class: No)	Class: No)

Berdasarkan tabel di atas, kita dapat melihat bahwa *Naïve Bayes* secara konsisten memiliki jumlah *False positives* (FP) terendah di semua *fold*, berkisar antara 52.000 hingga 143.200. Hal ini menunjukkan bahwa model ini jarang salah dalam memprediksi kasus negatif sebagai positif. Namun, di sisi lain,

Naïve Bayes juga menunjukkan jumlah *False negatives* (FN) yang cukup tinggi, yang berarti model ini cenderung melewatkan sejumlah kasus positif yang sebenarnya.

Sementara itu, *XG Boost* memiliki jumlah FP tertinggi di antara model lainnya, berkisar antara 109.080 hingga 275.400, menunjukkan bahwa model ini sering salah memprediksi kasus negatif sebagai positif. Namun, jumlah FN pada *XG Boost* tidak diberikan dalam tabel ini, jadi kita tidak bisa melakukan analisis lebih lanjut tentang performa *Recall*-nya. *Decision Tree*, *Random Forest*, dan *KNN* menunjukkan jumlah FP yang bervariasi, tetapi cenderung lebih rendah daripada *XG Boost*. *Random Forest*, khususnya, memiliki performa yang baik dengan jumlah FP dan FN yang relatif rendah dan stabil di berbagai *fold*, menunjukkan kinerjanya yang seimbang dan andal. Secara keseluruhan, *Naïve Bayes* unggul dalam meminimalisir *False positives*, sementara *XG Boost* perlu evaluasi lebih lanjut untuk mengurangi kesalahan prediksi kasus negatif sebagai positif.

7) True Positives dan True Negatives

Tabel 8. Kinerja True Positive Model

Fo	10	15	20	25
ld	888.200 +/-	588.733 +/-	442.650 +/-	353.040 +/-
D	17.447	10.586	11.008	12.905
ec	(Micro	(Micro	(Micro	(Micro
isi	average:	average:	average:	average:
on	8882.000)	8831.000)	8853.000)	8826.000)
Tr	(Positive	(Positive	(Positive	(Positive
ee	Class: No)	Class: No)	Class: No)	Class: No)
	786.000 +/-	538.267 +/-	409.650 +/-	328.040 +/-
N	250.060	138.269	94.184	62.480
ai	(Micro	(Micro	(Micro	(Micro
ve	average:	average:	average:	average:
Ba	7860.000)	8074.000)	8193.000)	8201.000)
ye	(Positive	(Positive	(Positive	(Positive
s	Class: No)	Class: No)	Class: No)	Class: No)
Ra	887.000 +/-	600.267 +/-	449.550 +/-	360.240 +/-
nd	18.750	17.754	10.460	8.880
o	(Micro	(Micro	(Micro	(Micro
m	average:	average:	average:	average:
Fo	8870.000)	9004.000)	8991.000)	9006.000)
re	(Positive	(Positive	(Positive	(Positive
st	Class: No)	Class: No)	Class: No)	Class: No)
	843.300 +/-	563.133 +/-	422.450 +/-	337.680 +/-
	13.064	16.733	11.718	10.613
K	(Micro	(Micro	(Micro	(Micro
N	average:	average:	average:	average:
N	8433.000)	8447.000)	8449.000)	8442.000)
	(Positive	(Positive	(Positive	(Positive
	Class: No)	Class: No)	Class: No)	Class: No)
	957.700 +/-	637.667 +/-	478.550 +/-	383.600 +/-
X	15.041	14.652	10.831	8.718
G	(Micro	(Micro	(Micro	(Micro
Bo	average:	average:	average:	average:
os	9577.000)	9565.000)	9571.000)	9590.000)
t	(Positive	(Positive	(Positive	(Positive
	Class: No)	Class: No)	Class: No)	Class: No)

Tabel 9. Kinerja True Negative Model

Fold	10	15	20	25
Decision Tree	1420.900 +/- 20.300 (Micro average: 14209.000)	949.667 +/- 13.340 (Micro average: 14245.000)	709.600 +/- 10.689 (Micro average: 14192.000)	567.920 +/- 11.086 (Micro average: 14198.000)
	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)
	1490.400 +/- 101.412 (Micro average: 14904.000)	1000.200 +/- 41.399 (Micro average: 15003.000)	726.350 +/- 132.376 (Micro average: 14527.000)	601.440 +/- 21.235 (Micro average: 15036.000)
	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)
Random Forest	1441.100 +/- 12.423 (Micro average: 14411.000)	959.800 +/- 10.890 (Micro average: 14397.000)	719.850 +/- 5.659 (Micro average: 14397.000)	576.120 +/- 10.146 (Micro average: 14403.000)
	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)
	1433.600 +/- 18.350 (Micro average: 14336.000)	957.067 +/- 7.630 (Micro average: 14356.000)	717.900 +/- 8.920 (Micro average: 14358.000)	573.880 +/- 6.876 (Micro average: 14347.000)
	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)
XG Boost	1358.200 +/- 29.386 (Micro average: 13582.000)	908.600 +/- 14.706 (Micro average: 13629.000)	680.350 +/- 10.634 (Micro average: 13607.000)	544.360 +/- 10.965 (Micro average: 13609.000)
	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)	(Positive Class: No)

Berdasarkan tabel hasil True Positives (TP) dan True Negatives (TN) di atas, kita dapat melihat bahwa Random Forest dan XG Boost secara konsisten menunjukkan nilai TP dan TN yang tinggi di berbagai fold (10, 15, 20, dan 25). XG Boost memiliki nilai TP tertinggi, berkisar antara 478.550 hingga 957.700, yang menunjukkan kemampuannya yang sangat baik dalam memprediksi kasus positif dengan benar. Random Forest juga menunjukkan performa yang sangat baik dengan nilai TP berkisar antara 449.550 hingga 887.000.

Dalam hal True Negatives, Random Forest menunjukkan performa yang luar biasa dengan nilai TN tinggi dan stabil, berkisar antara 719.850 hingga 1441.100. XG Boost juga menunjukkan nilai TN yang tinggi, meskipun sedikit lebih rendah dibandingkan Random Forest, dengan nilai berkisar antara 680.350 hingga 1358.200. Decision Tree dan KNN memiliki nilai TP dan TN yang baik, meskipun tidak setinggi Random Forest dan XG Boost. Naive Bayes, meskipun memiliki nilai TN yang relatif tinggi, menunjukkan nilai TP yang lebih rendah dibandingkan model lainnya, menunjukkan bahwa model ini mungkin lebih baik dalam memprediksi

kasus negatif dengan benar tetapi kurang efektif dalam mendeteksi kasus positif.

Secara keseluruhan, Random Forest dan XG Boost menonjol sebagai model yang paling seimbang dan efektif dalam hal memprediksi baik kasus positif maupun negatif dengan benar, menjadikannya pilihan yang kuat untuk aplikasi yang memerlukan deteksi yang akurat dan andal.

8) Positive Predictive Value dan Negative Predictive Value

Berdasarkan tabel hasil Positive Predictive Value (PPV) dan Negative Predictive Value (NPV) di atas, kita dapat melihat beberapa poin penting tentang keakuratan prediksi positif dan negatif dari berbagai model klasifikasi yang diuji. Naive Bayes secara konsisten menunjukkan nilai PPV tertinggi di semua fold, berkisar antara 85.32% hingga 86.92%. Hal ini menunjukkan kemampuan model ini untuk membuat prediksi positif yang akurat, menjadikannya sangat efektif dalam mengurangi kesalahan prediksi positif palsu. Random Forest dan KNN juga menunjukkan nilai PPV yang cukup tinggi dan stabil, dengan Random Forest berkisar antara 82.17% hingga 82.37% dan KNN berkisar antara 80.85% hingga 81.05%. Decision Tree menunjukkan nilai PPV yang sedikit lebih rendah tetapi tetap stabil, berkisar antara 80.53% hingga 80.88%. Di sisi lain, XG Boost memiliki nilai PPV terendah di antara model lainnya, berkisar antara 77.71% hingga 77.97%, menunjukkan bahwa model ini sedikit kurang akurat dalam prediksi positif.

Dalam hal NPV, XG Boost menunjukkan performa terbaik dengan nilai tertinggi yang berkisar antara 87.22% hingga 87.35%. Ini menunjukkan kemampuan model ini dalam membuat prediksi negatif yang sangat akurat, mengurangi kesalahan prediksi negatif palsu. Random Forest juga menunjukkan performa yang sangat baik dengan nilai NPV tinggi dan stabil, berkisar antara 84.25% hingga 84.93%. Decision Tree menunjukkan nilai NPV yang cukup tinggi, berkisar antara 83.86% hingga 84.12%. KNN memiliki nilai NPV yang lebih rendah dibandingkan Random Forest dan XG Boost tetapi tetap stabil, berkisar antara 82.07% hingga 82.18%. Naive Bayes memiliki nilai NPV yang lebih rendah dibandingkan model lainnya, berkisar antara 81.32% hingga 82.31%, menunjukkan bahwa meskipun model ini sangat baik dalam prediksi positif, ia mungkin kurang efektif dalam prediksi negatif.

Secara keseluruhan, Naive Bayes unggul dalam prediksi positif dengan nilai PPV tertinggi, sementara XG Boost unggul dalam prediksi negatif dengan nilai NPV tertinggi. Random Forest menunjukkan performa yang baik dan seimbang

dalam kedua metrik, menjadikannya pilihan yang solid untuk aplikasi yang memerlukan keakuratan dalam prediksi positif dan negatif.

C. Pemilihan Model Terbaik

Berdasarkan data yang diperoleh, kita dapat menjelaskan alasan di balik pemilihan model tertentu dan bagaimana trade-off antara *Precision* dan *Recall* dapat mempengaruhi keputusan ini. Dari hasil eksperimen, model seperti *Random Forest* dan *XG Boost* secara konsisten menunjukkan performa yang seimbang dengan nilai *Precision* dan *Recall* yang tinggi. *Random Forest* memiliki nilai *Precision* berkisar antara 82.17% hingga 82.37% dan *Recall* antara 76.70% hingga 77.87%, sementara *XG Boost* memiliki nilai *Precision* berkisar antara 77.71% hingga 77.97% dan *Recall* antara 82.71% hingga 82.92%. Keseimbangan ini menunjukkan bahwa kedua model tersebut dapat memberikan prediksi positif yang akurat sekaligus menangkap sebagian besar kasus positif dengan benar. Di sisi lain, *Naïve Bayes* memiliki nilai *Precision* yang sangat tinggi, berkisar antara 85.32% hingga 86.92%, tetapi *Recall* yang lebih rendah, berkisar antara 67.96% hingga 70.91%, menunjukkan bahwa meskipun model ini akurat dalam prediksi positif, ada banyak kasus positif yang terlewatkan. Dengan mempertimbangkan trade-off antara *Precision* dan *Recall*, pemilihan model terbaik harus disesuaikan dengan kebutuhan spesifik dari aplikasi yang diinginkan. Jika tujuan utama adalah meminimalkan kesalahan prediksi positif, maka model dengan *Precision* tinggi seperti *Naïve Bayes* dapat lebih diutamakan. Namun, jika tujuan utama adalah menangkap sebanyak mungkin kasus positif, model dengan *Recall* tinggi seperti *XG Boost* dapat menjadi pilihan yang lebih baik. Dengan demikian, pemilihan model klasifikasi terbaik harus mempertimbangkan keseimbangan antara *Precision* dan *Recall* serta konteks aplikasi yang spesifik.

D. Impact dari Cross-Validation

Cross-Validation adalah teknik penting yang digunakan untuk menguji stabilitas dan keakuratan model. Di sub bab ini, kita akan mendiskusikan dampak dari penggunaan *Cross-Validation* dengan berbagai jumlah *fold* pada hasil eksperimen. Berdasarkan data yang telah dianalisis, terlihat bahwa penggunaan *Cross-Validation* dengan berbagai *fold* (10, 15, 20, dan 25) dapat memberikan wawasan berharga tentang konsistensi kinerja model. Sebagai contoh, model *Random Forest* dan *XG Boost* secara konsisten menunjukkan performa yang stabil dengan nilai akurasi dan F1 score yang tinggi di berbagai *fold*. Hal ini menunjukkan bahwa kedua model ini dapat diandalkan untuk menghasilkan prediksi yang akurat dalam berbagai kondisi data. Sebaliknya, *Naïve Bayes* menunjukkan variasi yang lebih besar dalam akurasi pada berbagai *fold*, meskipun mencapai performa tertingginya pada *fold* 25. Ini mengindikasikan bahwa model ini mungkin lebih sensitif terhadap variasi dalam data training dan memerlukan evaluasi lebih lanjut untuk memahami faktor-faktor yang mempengaruhi

kinerjanya. Dengan demikian, penggunaan *Cross-Validation* dengan jumlah *fold* yang berbeda dapat membantu dalam mengidentifikasi model yang paling stabil dan andal, serta memberikan informasi tambahan untuk pengoptimalan model yang lebih baik.

E. Analisis Kesalahan Pengaruh Parameter Model

Analisis kesalahan dapat memberikan wawasan berharga tentang performa model. Dengan memeriksa *False positives* (FP) dan *False negatives* (FN) yang terjadi, kita dapat mengidentifikasi pola dan penyebab umum dari kesalahan tersebut. Misalnya, berdasarkan data sebelumnya, *Naïve Bayes* menunjukkan jumlah FP yang relatif rendah namun memiliki jumlah FN yang cukup tinggi, yang mengindikasikan bahwa model ini cenderung lebih sering gagal mendeteksi kasus positif. Sebaliknya, *XG Boost* memiliki jumlah FN yang lebih rendah tetapi menunjukkan jumlah FP yang lebih tinggi, menunjukkan kecenderungan model ini untuk sering salah memprediksi kasus negatif sebagai positif.

Memahami pola kesalahan ini dapat membantu dalam mengambil langkah-langkah perbaikan, seperti penyesuaian threshold atau penerapan teknik resampling untuk menangani ketidakseimbangan kelas. Selain itu, evaluasi kesalahan juga dapat mengarahkan pengoptimalan *Hyperparameter* atau pemilihan fitur yang lebih relevan untuk meningkatkan akurasi prediksi. Dengan demikian, analisis kesalahan tidak hanya memberikan gambaran tentang kinerja model tetapi juga memberikan panduan berharga untuk perbaikan lebih lanjut dalam implementasi model.

F. Pengaruh Parameter Model

Hyperparameter tuning dapat secara signifikan mempengaruhi kinerja model. Dalam proses ini, parameter-parameter tertentu dari model diatur untuk mengoptimalkan kinerja model berdasarkan dataset yang digunakan. Sebagai contoh, dalam model *Random Forest*, jumlah pohon (*trees*) dan kedalaman maksimal pohon (*max depth*) adalah *Hyperparameter* penting yang dapat disesuaikan. Pengaturan yang tepat dari *Hyperparameter* ini dapat meningkatkan akurasi dan stabilitas model. Pada model *XG Boost*, *learning rate* dan jumlah *boosting rounds* adalah contoh lain dari *Hyperparameter* yang dapat diatur untuk meningkatkan kinerja. Dengan melakukan *grid search* atau *random search*, kita dapat menemukan kombinasi *Hyperparameter* yang menghasilkan performa terbaik. Pentingnya *Hyperparameter tuning* tidak dapat diremehkan, karena pengaturan yang tidak optimal dapat menyebabkan model *underfitting* atau *overfitting*, yang pada akhirnya mengurangi keakuratan prediksi. Oleh karena itu, pengoptimalan *Hyperparameter* adalah langkah krusial dalam pengembangan model *machine learning* yang efektif.

Pengoptimalan berbagai parameter model dapat memiliki dampak yang signifikan pada kinerja dan hasil

yang dihasilkan oleh model tersebut. Misalnya, dalam model *Random Forest*, beberapa parameter kunci yang dapat disesuaikan adalah jumlah pohon (*trees*) dan kedalaman maksimal pohon (*max depth*). Penyesuaian jumlah pohon dapat membantu dalam mengurangi *overfitting*, sementara pengaturan kedalaman maksimal pohon dapat mengontrol kompleksitas model, sehingga mencapai keseimbangan yang lebih baik antara bias dan varians. *Hyperparameter* seperti "*min_samples_split*" dan "*min_samples_leaf*" juga memainkan peran penting dalam menentukan seberapa mudah model dapat membagi node internal dan ukuran minimum daun.

Pada model *XG Boost*, parameter seperti "*learning rate*" dan "*n_estimators*" (jumlah *boosting rounds*) adalah kunci untuk meningkatkan kinerja model. *Learning rate* yang lebih rendah seringkali menghasilkan model yang lebih akurat namun membutuhkan lebih banyak *boosting rounds*. Selain itu, pengaturan parameter "*max_depth*" dan "*min_child_weight*" membantu mengontrol *overfitting* dengan mengatur kedalaman pohon dan ukuran minimum berat anak. Parameter "*subsample*" dan "*colsample_bytree*" juga dapat dioptimalkan untuk meningkatkan generalisasi model dengan memilih proporsi data dan fitur yang digunakan pada setiap iterasi.

Proses pengoptimalan *Hyperparameter* dapat dilakukan melalui teknik seperti *grid search* atau *random search*, yang secara sistematis mencoba kombinasi parameter yang berbeda untuk menemukan konfigurasi terbaik. Selain itu, penggunaan teknik *Cross-Validation* dapat memastikan bahwa hasil yang diperoleh dari pengaturan *Hyperparameter* adalah stabil dan dapat diandalkan. Dengan pengaturan parameter yang optimal, model dapat mencapai performa yang lebih baik, baik dalam hal akurasi, *Precision*, *Recall*, dan metrik lainnya, sehingga memberikan hasil yang lebih andal dan efektif dalam aplikasi dunia nyata.

G. Aplikasi dan Implikasi Praktis

Model klasifikasi yang dipilih tidak hanya diuji di lingkungan lab, tetapi juga memiliki aplikasi praktis di dunia nyata. Misalnya, model dengan kinerja tinggi seperti *Random Forest* dan *XG Boost*, yang menunjukkan nilai *Precision* dan *Recall* yang seimbang, dapat digunakan dalam berbagai bidang seperti deteksi penipuan, diagnosis medis, dan analisis risiko. Dalam deteksi penipuan, kemampuan model untuk mengidentifikasi transaksi yang mencurigakan dengan akurasi tinggi sangat penting untuk mengurangi kerugian finansial. Di bidang medis, model ini dapat membantu dalam mendeteksi penyakit pada tahap awal dengan presisi tinggi, yang dapat meningkatkan hasil kesehatan pasien.

Selain itu, model klasifikasi ini juga dapat diterapkan dalam analisis risiko, seperti di sektor keuangan, di mana mereka dapat membantu dalam mengevaluasi kelayakan kredit dan memprediksi kemungkinan *default*. Penggunaan model-model ini di dunia nyata

menunjukkan bahwa performa yang baik di lingkungan lab dapat diterjemahkan menjadi solusi praktis yang efektif. Pentingnya memahami trade-off antara *Precision* dan *Recall*, serta mengoptimalkan *Hyperparameter*, adalah kunci untuk memastikan model ini memberikan hasil yang akurat dan andal dalam aplikasi yang sesungguhnya.

Random Forest dan *XG Boost* dapat diterapkan dalam berbagai skenario serta implikasi potensialnya.

1) Deteksi Penipuan

Di bidang keuangan, model klasifikasi seperti *Random Forest* dan *XG Boost* dapat digunakan untuk mendeteksi aktivitas penipuan dengan akurasi tinggi. Model ini dapat mempelajari pola transaksi yang mencurigakan dari data historis dan mengidentifikasi transaksi yang tidak wajar dalam waktu nyata. Dengan *Precision* dan *Recall* yang tinggi, model ini dapat meminimalkan kesalahan deteksi dan membantu bank serta lembaga keuangan mengurangi kerugian finansial akibat penipuan.

2) Diagnosis Medis

Dalam dunia medis, kemampuan model klasifikasi untuk mendeteksi penyakit pada tahap awal sangat penting. *Random Forest* dan *XG Boost* dapat digunakan untuk mengidentifikasi pasien yang berisiko tinggi berdasarkan data medis seperti hasil tes laboratorium, riwayat kesehatan, dan gejala. Dengan *Recall* yang tinggi, model ini dapat membantu dokter dalam mengambil tindakan pencegahan lebih awal, meningkatkan hasil kesehatan pasien, dan mengurangi biaya perawatan jangka panjang.

3) Analisis Risiko Kredit

Di sektor perbankan, model ini dapat digunakan untuk mengevaluasi kelayakan kredit calon peminjam. *Random Forest* dan *XG Boost* dapat menganalisis data kredit historis dan berbagai faktor risiko untuk memprediksi kemungkinan gagal bayar. Dengan akurasi dan *Precision* yang tinggi, bank dapat membuat keputusan kredit yang lebih baik, mengurangi risiko kerugian finansial, dan meningkatkan profitabilitas.

4) Deteksi *Spam* dan Filtering

Model klasifikasi ini juga dapat digunakan untuk deteksi *Spam* dan filtering dalam sistem email. Dengan menganalisis konten email dan pola pengiriman, *Random Forest* dan *XG Boost* dapat secara akurat mengidentifikasi *email Spam* dan memisahkannya dari email yang sah. Ini membantu pengguna untuk tetap fokus pada komunikasi yang penting dan mengurangi gangguan dari *email Spam*.

5) Prediksi Permintaan Pasar

Di sektor ritel, model klasifikasi dapat digunakan untuk memprediksi permintaan pasar dan

mengoptimalkan manajemen persediaan. *Random Forest* dan *XG Boost* dapat menganalisis tren pembelian historis, data penjualan, dan faktor eksternal lainnya untuk memprediksi permintaan produk di masa mendatang. Dengan akurasi yang tinggi, retailer dapat mengurangi kelebihan stok dan kekurangan stok, serta meningkatkan efisiensi operasi.

6) Implikasi Potensial

Dengan penerapan model-model ini di berbagai bidang, implikasi potensialnya sangat luas. Selain meningkatkan efisiensi operasional dan mengurangi biaya, model ini dapat meningkatkan akurasi dalam pengambilan keputusan dan memberikan keuntungan kompetitif bagi organisasi. Namun, penting juga untuk mempertimbangkan aspek etika dan privasi, terutama dalam pemanfaatan data sensitif seperti informasi kesehatan dan keuangan. Dengan pendekatan yang hati-hati dan bertanggung jawab, model klasifikasi ini dapat membawa manfaat besar bagi masyarakat dan industri.

H. Kebaruan dalam penelitian

Kebaruan dalam penelitian ini terletak pada beberapa aspek utama yang membedakannya dari penelitian sebelumnya:

1) Komparatif dan Komprehensif

Penelitian ini tidak hanya fokus pada satu atau dua model klasifikasi, tetapi mencakup perbandingan mendalam antara lima model yang berbeda: *Decision Tree*, *Naïve Bayes*, *Random Forest*, *KNN*, dan *XG Boost*. Ini memberikan wawasan yang lebih luas tentang kelebihan dan kekurangan masing-masing model dalam berbagai skenario dan metrik kinerja.

2) Penggunaan Teknik *Cross-Validation* yang Bervariasi

Penggunaan berbagai jumlah *fold* dalam *Cross-Validation* (10, 15, 20, dan 25) memberikan analisis yang lebih komprehensif tentang stabilitas dan keandalan model. Ini membantu dalam mengidentifikasi model yang paling konsisten dalam berbagai kondisi data, yang merupakan pendekatan yang lebih mendalam dibandingkan dengan penggunaan *Cross-Validation* tunggal.

3) Analisis Kesalahan Terperinci

Penelitian ini mengintegrasikan analisis mendalam tentang *False positives* dan *False negatives*, yang membantu mengidentifikasi pola kesalahan dan langkah-langkah perbaikan yang mungkin. Pendekatan ini memberikan pemahaman yang lebih baik tentang performa model dan bagaimana cara meningkatkan akurasinya.

4) Evaluasi Berbagai Metrik Kinerja

Selain akurasi, penelitian ini juga mengevaluasi model berdasarkan metrik kinerja lain seperti AUC, *Precision*, *Recall*, *F1 score*, dan nilai PPV dan NPV. Ini memberikan gambaran yang lebih komprehensif tentang performa model dalam berbagai aspek dan membantu dalam pemilihan model yang paling sesuai dengan kebutuhan spesifik.

5) Implikasi Praktis dan Aplikasi Dunia Nyata

Penelitian ini tidak hanya berhenti pada evaluasi model di lingkungan lab, tetapi juga mengeksplorasi aplikasi praktis dari model-model tersebut dalam dunia nyata. Dengan mempertimbangkan aplikasi praktis dalam berbagai bidang seperti deteksi penipuan, diagnosis medis, dan analisis risiko kredit, penelitian ini memberikan kontribusi yang signifikan dalam menerjemahkan hasil penelitian ke dalam solusi praktis yang dapat diterapkan.

Dengan berbagai kebaruan ini, penelitian ini memberikan kontribusi yang signifikan dalam bidang klasifikasi dan *machine learning*, serta menawarkan panduan berharga untuk pengembangan dan implementasi model yang lebih baik di berbagai aplikasi dunia nyata.

I. Peluang Penelitian Lanjutan

Peluang penelitian lanjutan yang muncul dari hasil penelitian ini sangat luas, terutama dalam pengembangan dan penerapan teknik baru untuk meningkatkan kinerja model klasifikasi, khususnya dalam menangani *trade-off* antara *Precision* dan *Recall*. Salah satu arah pengembangan adalah pencarian algoritma yang lebih baik atau modifikasi dari algoritma yang ada untuk lebih menyeimbangkan *Precision* dan *Recall*. Saat ini, banyak algoritma masih terjebak dalam pengorbanan antara keduanya, sehingga penelitian lebih lanjut diperlukan untuk mengembangkan teknik atau model baru yang dapat secara dinamis menyesuaikan keduanya tergantung pada aplikasi spesifik. Misalnya, pendekatan berbasis optimasi multi-objektif atau metaheuristik, seperti algoritma genetika, dapat digunakan untuk mencari solusi yang meminimalkan kesalahan baik dalam *Precision* maupun *Recall* secara bersamaan, memberikan keseimbangan yang lebih baik untuk berbagai skenario aplikasi.

Selain itu, eksplorasi data augmentation atau *Hyperparameter tuning* juga memiliki potensi besar dalam meningkatkan performa model klasifikasi. Data augmentation, yang lebih sering digunakan dalam bidang pengolahan citra, dapat diterapkan pada data tabular dengan menggunakan teknik seperti SMOTE (*Synthetic Minority Over-sampling Technique*) atau bahkan GANs (*Generative Adversarial Networks*) untuk menghasilkan data sintesis yang dapat memperkaya dataset, terutama untuk kelas yang kurang terwakili. Ini sangat relevan dalam konteks dataset yang tidak seimbang, di mana teknik ini dapat membantu meningkatkan *Recall* dengan

menciptakan lebih banyak contoh untuk kelas minoritas. *Hyperparameter tuning* juga bisa menjadi pendekatan yang sangat efektif untuk meningkatkan performa model. Penelitian lanjutan dapat fokus pada penerapan teknik optimasi, seperti *Grid Search* atau Bayesian Optimization, untuk menemukan kombinasi *Hyperparameter* yang optimal, sehingga model dapat berfungsi lebih baik pada berbagai skenario data dan rasio pelatihan.

Lebih jauh lagi, untuk memastikan bahwa temuan dari penelitian ini dapat digeneralisasi, studi dengan dataset lain perlu dilakukan untuk melihat apakah hasil yang diperoleh dapat diterapkan pada konteks yang lebih luas. Penggunaan dataset yang lebih beragam, baik dari segi ukuran, domain, atau kualitas data, akan menguji ketahanan dan efektivitas model pada situasi dunia nyata yang lebih kompleks. Misalnya, dataset yang berisi data dari sektor keuangan, medis, atau sensor *IoT* dapat memberikan tantangan yang berbeda yang akan membantu memahami kekuatan dan keterbatasan model yang telah diuji. Selain itu, penelitian yang melibatkan *Cross-Validation* dengan berbagai subset dataset dan penggunaan ensemble learning dapat memperluas kemampuan model untuk menangani variasi dalam data yang lebih besar dan lebih dinamis. Dengan melakukan studi-studi ini, diharapkan dapat ditemukan algoritma dan pendekatan yang lebih kuat yang dapat diadaptasi untuk berbagai jenis aplikasi praktis.

IV. KESIMPULAN

Hasil penelitian ini menunjukkan bahwa *Random Forest* adalah algoritma dengan performa terbaik dalam klasifikasi, memberikan keseimbangan optimal antara *Accuracy*, *Precision*, dan *Recall* pada dataset yang digunakan. Rasio data 80:20 terbukti memberikan hasil yang lebih stabil, mengurangi risiko *overfitting* dan *underfitting*, serta meningkatkan kemampuan generalisasi model. Penelitian ini berhasil menjawab tujuan untuk mengevaluasi performa beberapa algoritma klasifikasi dan memberikan wawasan tentang bagaimana pemilihan rasio data dapat mempengaruhi kinerja model. Temuan ini memiliki potensi aplikasi dalam berbagai bidang, seperti deteksi anomali, pengenalan pola, dan prediksi di industri kesehatan atau keuangan, di mana keseimbangan antara *Recall* dan *Precision* sangat penting. Implikasinya adalah pentingnya memilih model yang sesuai dengan tujuan aplikasi, seperti memprioritaskan *Recall* tinggi untuk deteksi masalah atau *Precision* tinggi untuk aplikasi yang memerlukan ketelitian. Penelitian lebih lanjut disarankan untuk mengeksplorasi teknik data augmentation dan *Hyperparameter tuning* untuk meningkatkan performa model, serta menguji model pada dataset yang lebih beragam guna menguji generalisasi hasil penelitian ini pada aplikasi dunia nyata.

REFERENSI

- [1] J. Edwin, J. Cornwall, and K. Du Plooy, "What medical students want from Mental Health Self-Help Resources: A focus group study," *Ment Health Prev*, vol. 35, p. 200353, 2024, doi: <https://doi.org/10.1016/j.mhp.2024.200353>.
- [2] H. Liu, K. Zhang, L. Wang, and J. Chen, "Educational burden reduction, educational inequality, and enrollment pressure: Evidence from China," *Econ Hum Biol*, vol. 56, p. 101459, 2025, doi: <https://doi.org/10.1016/j.ehb.2024.101459>.
- [3] D. Qu *et al.*, "School mental health prevention and intervention strategies in China: a scoping review," *Lancet Reg Health West Pac*, vol. 53, p. 101243, 2024, doi: <https://doi.org/10.1016/j.lanwpc.2024.101243>.
- [4] K. Maximova, M. K. A. Khan, J. Dabravolskaj, L. Maunula, A. Ohinmaa, and P. J. Veugelers, "Perceived changes in lifestyle behaviours and in mental health and wellbeing of elementary school children during the first COVID-19 lockdown in Canada," *Public Health*, vol. 202, pp. 35–42, 2022, doi: <https://doi.org/10.1016/j.puhe.2021.10.007>.
- [5] J. Zhang and T. Chen, "Artificial intelligence based social robots in the process of student mental health diagnosis," *Entertain Comput*, vol. 52, p. 100799, 2025, doi: <https://doi.org/10.1016/j.entcom.2024.100799>.
- [6] O. Kobo, A. Meltzer-Asscher, J. Berant, and T. Schonberg, "Classification of depression tendency from gaze patterns during sentence reading," *Biomed Signal Process Control*, vol. 93, p. 106015, 2024, doi: <https://doi.org/10.1016/j.bspc.2024.106015>.
- [7] M. Wan and S. Zou, "Adolescent mental health state assessment framework by combining YOLO with *Random Forest*," *Appl Soft Comput*, vol. 168, p. 112497, 2025, doi: <https://doi.org/10.1016/j.asoc.2024.112497>.
- [8] A. M. Mariano, A. B. de M. L. Ferreira, M. R. Santos, M. L. Castilho, and A. C. F. L. C. Bastos, "*Decision Trees* for predicting dropout in Engineering Course students in Brazil," *Procedia Comput Sci*, vol. 214, pp. 1113–1120, 2022, doi: <https://doi.org/10.1016/j.procs.2022.11.285>.
- [9] S. S. Sara, Md. A. Rahman, R. Rahman, and A. Talukder, "Prediction of suicidal ideation with associated risk factors among university students in the southern part of Bangladesh: *Machine learning* approach," *J Affect Disord*, vol. 349, pp. 502–508, 2024, doi: <https://doi.org/10.1016/j.jad.2024.01.092>.
- [10] A. Doryab *et al.*, "Identifying Behavioral Phenotypes of Loneliness and Social Isolation with Passive Sensing: Statistical Analysis, Data Mining and *Machine learning* of Smartphone and Fitbit Data," *JMIR Mhealth Uhealth*, vol. 7, no. 7, 2019, doi: <https://doi.org/10.2196/13209>.
- [11] Z. Liu, Y. Liu, J. Dezert, and Q. Pan, "Classification of incomplete data based on belief functions and K-nearest neighbors," *Knowl Based Syst*, vol. 89, pp. 113–125, 2015, doi: <https://doi.org/10.1016/j.knosys.2015.06.022>.
- [12] V. Plotnikova, M. Dumas, and F. P. Milani, "Applying the CRISP-DM data mining process in the financial services industry: Elicitation of adaptation requirements," *Data Knowl Eng*, vol. 139, p. 102013, 2022, doi: <https://doi.org/10.1016/j.datak.2022.102013>.
- [13] S. Zulfikar, H. A. Al-Reshidi, M. A. Al Moteri, H. M. B. Feroz, N. Yahya, and W. M. Al-Rahmi, "Understanding and predicting students' entrepreneurial intention through business simulation games: A perspective of covid-19," *Sustainability (Switzerland)*, vol. 13, no. 4, pp. 1–27, May 2021, doi: 10.3390/su13041838.
- [14] M. F. Uddin and J. Lee, "We Are What We Generate - Understanding Ourselves Through Our Data," *Procedia Comput Sci*, vol. 95, pp. 335–344, 2016, doi: 10.1016/j.procs.2016.09.343.
- [15] B. C. F. M. Schutijser, J. E. Klopotowska, I. P. Jongerden, C. Wagner, and M. C. de Bruijne, "Feasibility of reusing routinely

- recorded data to monitor the safe preparation and administration of injectable medication: A multicenter cross-sectional study,” *Int J Med Inform*, vol. 141, p. 104201, 2020, doi: <https://doi.org/10.1016/j.ijmedinf.2020.104201>.
- [16] T. Hamim, F. Benabbou, and N. Sael, “Student profile modeling using boosting algorithms,” *International Journal of Web-Based Learning and Teaching Technologies*, vol. 17, no. 5, pp. 1–13, May 2022, doi: 10.4018/IJWLTT.20220901.0a4.
- [17] G. Pandolfo, A. D’Ambrosio, L. Cannavacciuolo, and R. Siciliano, “Fuzzy logic aggregation of crisp data partitions as learning analytics in triage decisions,” *Expert Syst Appl*, vol. 158, p. 113512, 2020, doi: 10.1016/j.eswa.2020.113512.
- [18] D. Chicco and G. Jurman, “The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation,” *BMC Genomics*, vol. 21, no. 1, pp. 1–13, 2020.
- [19] A. M. G. Milla *et al.*, “Accuracy of visceral adiposity indices and lipid accumulation products in the identification of adults at high cardiovascular risk,” *Clínica e Investigación en Arteriosclerosis*, 2023, doi: <https://doi.org/10.1016/j.arteri.2023.04.002>.
- [20] X. Niu, Z. Yang, N. Zhou, and C. Li, “A novel method for cage whirl motion capture of high-Precision bearing inspired by U-Net,” *Eng Appl Artif Intell*, vol. 117, p. 105552, 2023, doi: <https://doi.org/10.1016/j.engappai.2022.105552>.
- [21] Z. A. Rana, M. A. Mian, and S. Shamil, “Improving Recall of software defect prediction models using association mining,” *Knowl Based Syst*, vol. 90, pp. 1–13, 2015, doi: <https://doi.org/10.1016/j.knosys.2015.10.009>.
- [22] B. C. Yan, H. W. Wang, S. W. F. Jiang, F. A. Chao, and B. Chen, “Maximum F1-Score Training for End-to-End Mispronunciation Detection and Diagnosis of L2 English Speech,” 2022, *IEEE*. doi: 10.1109/ICME52920.2022.9858931.