LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818

Analisis Perbandingan Algoritma XGBoost Dan Algoritma *Random Forest* Untuk Klasifikasi Data Kesehatan Mental

Kadek Aditya Ananta Wisnu Wardana[1*], Abdul Mizwar A. Rahim[2]

[1,2]Fakultas Ilmu Komputer, Informatika, Universitas Amikom Yogyakarta, Indonesia
Email : [1*]Kadek3912@students.amikom.ac.id, [2]abdulmizwar@amikom.ac.id
(* : coressponding author)

Abstrak - Kesehatan mental merupakan sebuah perhatian utama dalam upaya meningkatkan kesejahteraan secara keseluruhan. Permasalahan kesehatan mental ini menjadi sesuatu yang tidak bisa diabaikan begitu saja dan perlu ditangani lebih lanjut. Namun, dalam dunia nyata masih banyak masyarakat yang belum menyadari hal tersebut. Oleh karena itu, penelitian ini dilakukan dengan menggunakan sebuah algoritma untuk klasifikasi kesehatan mental menggunakan perbandingan 2 algoritma yaitu *Extreme Gradient Boost* dan *Random Forest* dengan dataset yang diambil kaggle dengan nama mental health dataset.csv berdasarkan beberapa fitur yaitu Jenis Kelamin, Negara, Pekerjaan, Pengobatan, Stres yang Bertambah, Perubahan Kebiasaan, Riwayat Kesehatan Mental, Perubahan Mood, Wawancara Kesehatan Mental, dan Pilihan Perawatan. Metode penelitian dilakukan dengan pemeriksaan data dengan melakukan analisi pada dataset serta membagi data menjadi data training dan data test. Hasil penelitian pada algoritma *Extreme Gradient Boost* memiliki rata-rata nilai akurasi, presisi, recall, dan f1-score dalam 30 kali percobaan sebesar 99.82% sedangkan algoritma *Random Forest* memiliki rata-rata nilai akurasi, presisi, recall, dan f1-score dalam 30 kali percobaan sebesar 99.04%.

Kata Kunci: Kesehatan Mental, *Extreme Gradient Boost*, *Random Forest*.

Abstract *- Mental health is a primary concern in efforts to enhance overall well-being. This issue cannot be ignored and needs to be addressed further. However, in reality, many people are still unaware of its importance. Therefore, this study was conducted using an algorithm to classify mental health by comparing two algorithms, namely Extreme Gradient Boost and Random Forest, with a dataset taken from Kaggle named mental health dataset.csv. The dataset includes several features: Gender, Country, Occupation, Treatment, Increased Stress, Habit Changes, Mental Health History, Mood Changes, Mental Health Interview, and Treatment Options. The research method involved data examination through dataset analysis and splitting the data into training and test sets. The results showed that the Extreme Gradient Boost algorithm had an average accuracy, precision, recall, and f1-score of 99.82% over 30 trials, whereas the Random Forest algorithm had an average accuracy, precision, recall, and f1-score of 99.04% over 30 trials.*

Keywords: *Mental Health, Extreme Gradient Boost, Random Forest.*

1. PENDAHULUAN

Kesehatan merupakan aspek yang tak terpisahkan dari kehidupan manusia, mencakup kondisi fisik, mental, dan sosial yang memungkinkan seseorang merasa sehat secara keseluruhan. Kesehatan optimal memberikan kemampuan mengatasi tantangan sehari-hari, mencapai tujuan hidup, dan memberikan kontribusi positif kepada masyarakat. Namun, perhatian terhadap kesehatan fisik sering kali lebih diutamakan dibandingkan kesehatan mental, meski keduanya saling mempengaruhi.

Berbagai faktor mempengaruhi kesehatan secara keseluruhan, termasuk gaya hidup, lingkungan fisik, situasi ekonomi, akses terhadap pelayanan kesehatan, dan dukungan sosial. Pola makan yang tidak seimbang dan kurangnya aktivitas fisik dapat meningkatkan risiko penyakit fisik, sementara tekanan finansial dan kurangnya dukungan sosial dapat memperburuk kesehatan mental. Menurut Organisasi Kesehatan Dunia (WHO), kesehatan mental adalah bagian penting dari kesehatan secara keseluruhan. Kesehatan mental mencakup keadaan sejahtera yang memungkinkan seseorang mengatasi stres hidup dengan tepat, bekerja secara produktif, dan berpartisipasi aktif dalam kehidupan sosial(Rosta Br Sebayang et al., 2023).

Permasalahan kesehatan mental tidak bisa diabaikan dan memerlukan penanganan lebih lanjut. Algoritma klasifikasi data mining dapat digunakan untuk menganalisis masalah kesehatan mental ini. Metode seperti XGBoost dan *Random Forest* efektif dalam mengidentifikasi pola dan faktor yang berkontribusi terhadap kondisi kesehatan mental. Penelitian yang dilakukan oleh Givari

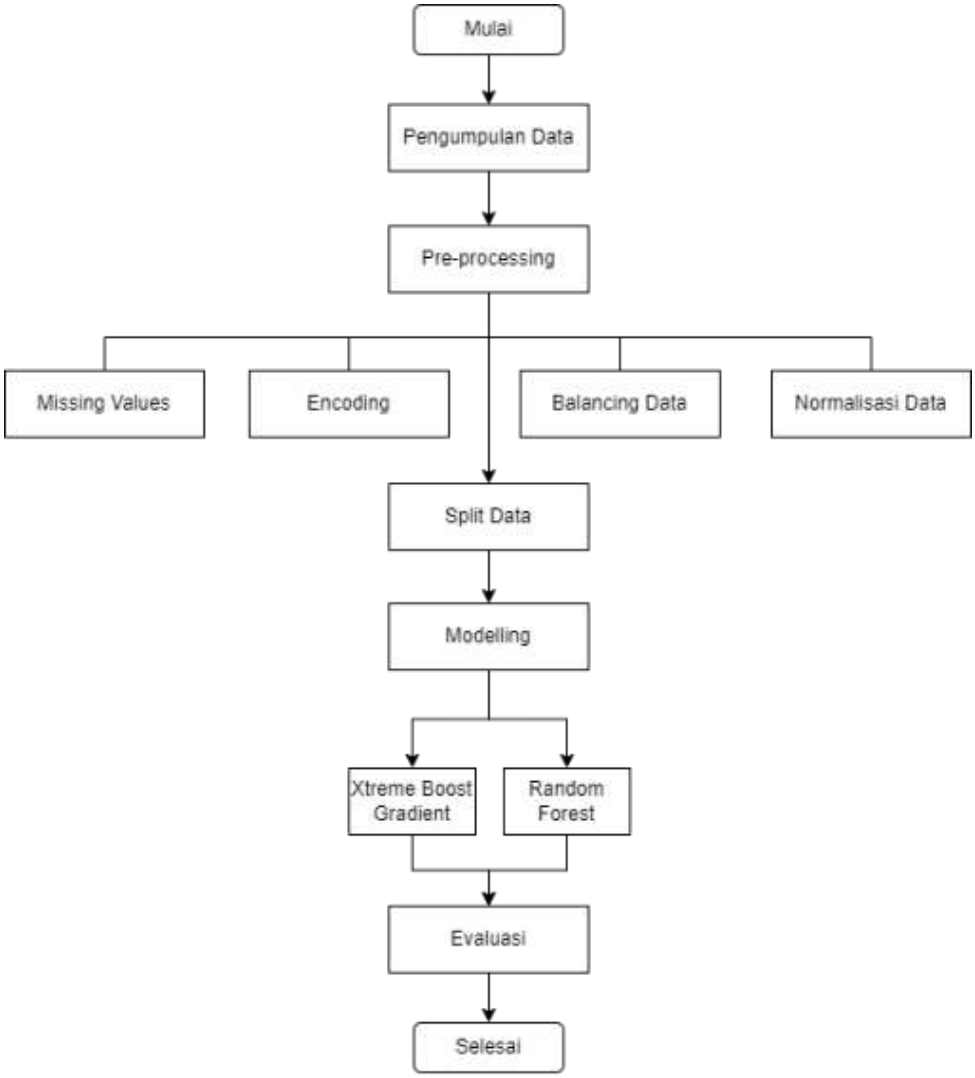
LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818

M, Sulaeman M & Umaidah Y menunjukkan bahwa XGBoost memiliki akurasi tertinggi (82%) dibandingkan dengan SVM dan *Random Forest* dalam penentuan persetujuan pengajuan kredit (Givari et al., 2022). Penelitian lain oleh Jan Melvin et al. juga menemukan bahwa XGBoost lebih akurat dibandingkan *Random Forest* dalam memberikan keputusan kredit(Jan Melvin Ayu Soraya Dachi & Pardomuan Sitompul, 2023).

Selain itu, penelitian oleh Adriansyah D dan Eka Wulansari F membandingkan performa algoritma SVM dan XGBoost dalam mendeteksi kanker payudara, dengan hasil XGBoost memiliki performa lebih baik (91,4%) dibandingkan SVM (89,8%)(Andriansyah & Eka Wulansari Fridayanthie, 2023). Berdasarkan penelitian-penelitian ini, penulis melakukan penelitian dengan judul "Analisis Perbandingan Algoritma XGBoost dan Algoritma *Random Forest* untuk Klasifikasi Data Kesehatan Mental" untuk membandingkan performa kedua algoritma tersebut terhadap data kesehatan mental. Tujuannya adalah agar para psikolog dapat lebih efektif dalam mendeteksi masalah kesehatan mental pada individu.

2. METODE

Dalam penelitian ini terdapat alur penelitian yang dibuat dalam bentuk flowchart seperti yang ada dibawah ini:



Gambar 1. Alur Penelitian

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818

2.1 Pengumpulan Dataset

Proses pengumpulan data dilakukan melalui website Kaggle dengan nama Mental Health Dataset yang diupload oleh Bhavik Jikadara. Total jumlah dataset ini 292364 data. Dalam dataset ini terdapat 17 Fitur yaitu Timestamp, Gender, Country, Occupation, self_employed, family_history, treatment, Days_Indoors, Growing_Stress, Changes_Habits, Mental_Health_History, Mood_Swings, Coping_Struggles, Work_Interest, Social_Weakness, mental_health_interview, care_options. Source dataset yang ada bisa diakses melalui link yang ada disamping ini : https://www.kaggle.com/datasets/bhavikjikadara/mental-health-dataset

2.2. *Preprocessing*

Tahapan *pre-processing* dalam analisis data pada data mining dapat dibagi menjadi beberapa sub bagian, yaitu :

2.2.1 *Missing Values*

Missing values adalah sebuah data yang hilang atau data yang tidak tersedia dalam dataset. Missing values ini bisa berdampak pada performa model yang akan dibuat dikarenakan model akan kesusahan dalam menangani data yang hilang tersebut. Terdapat beberapa cara untuk dapat menangani missing values diantaranya : Mengisi nilai yang hilang dengan nilai rata-rata (mean), nilai tengah (median), ataupun mengisi missing values berdasarkan nilai default dari fitur lainnya. Pada penelitian ini akan menggunakan penanganan missing values berdasarkan nilai default dari fitur lainnya.

2.2.2 *Encoding*

Encoding (pengkodean) dalam data mining melibatkan berbagai metode untuk mengubah data mentah menjadi format terstruktur yang cocok untuk analisis (Doctor et al., 2024). Dalam penelitian ini proses encoding menggunakan 2 teknik yaitu Label Encoding dan One Hot Encoding . Label encoding memberikan nilai numerik unik kepada setiap kategori, yang dapat memperkenalkan sifat ordinal dan menciptakan hubungan yang tidak diinginkan antara kategori . Di sisi lain, one-hot encoding merepresentasikan setiap kategori sebagai vektor biner, di mana hanya satu bit yang aktif untuk menunjukkan kategori, sehingga menghindari masalah ordinality (Dahouda & Joe, 2021) .

2.2.3 *Balancing Data*

Balancing data adalah teknik penting dalam pembelajaran mesin untuk mengatasi masalah dataset yang tidak seimbang, di mana beberapa kelas memiliki jumlah instance yang jauh lebih sedikit daripada yang lain (Moore et al., 2023) . Pada penelitian ini teknik balancing data yang digunakan adalah SMOTE (*Synthetic Minority Over-sampling Technique*). Teknik ini menyamakan kelas minoritas agar setara dengan kelas mayoritas sehingga performa model dapat meningkat.

2.2.4 Normalisasi Data

Normalisasi data adalah langkah prapemrosesan yang krusial dalam berbagai bidang seperti pemantauan sistem reverse osmosis, eksperimen psikologis, data mining, dan pembelajaran mesin (“Data Collection and Normalization,” 2023). Proses ini melibatkan penyesuaian data ke skala yang umum untuk menghilangkan pengaruh dari satuan atau skala yang berbeda, sehingga memungkinkan perbandingan yang adil dan meningkatkan akurasi model. Penelitian ini menggunakan MinMaxScalar dengan library Scikitlearn. MinMaxScaler melakukan proses standarisasi fitur-fitur dalam data agar berada dalam rentang tertentu, biasanya antara 0 dan 1.

2.3 *Splitting Data*

Proses pembagian data dibagi menjadi 2 bagian yaitu data train dan data testing. Tujuan utama dari pembagian data adalah untuk menilai kemampuan model ketika dihadapkan dengan data yang sudah dipelajari. Dalam proses pembagian data terdapat berbagai proporsi dengan tujuan analisis yang berbeda tergantung kebutuhan. Dalam penelitian ini pembagian data train dan data testing adalah 80:20 yang mana 80% digunakan untuk train data sedangkan 20% digunakan untuk testing data.

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818

2.4 *Modelling*

Proses pembuatan model atau algoritma digunakan untuk memahami dan memprediksi perilaku atau hubungan antar variabel dalam kumpulan data. Model yang dibuat dapat digunakan untuk mengklasifikasi atau memprediksi perilaku atau hasil yang mirip dari sebuah kumpulan data antar variabel. Algoritma yang digunakan adalah *Extreme Gradient Boost* dan *Random Forest*.

2.4.1 *Extreme Gradient Boost* (XGBoost)

*Extreme Gradient Boost*ing (XGBoost) adalah algoritma pembelajaran mesin yang kuat yang telah menunjukkan akurasi dan kinerja luar biasa di berbagai domain. XGBoost bekerja dengan membangun serangkaian pohon keputusan secara berurutan, di mana setiap pohon berikutnya mengoreksi kesalahan dari yang sebelumnya (Prakash et al., 2023). Rumus XGBoost memperkenalkan istilah regularisasi dalam fungsi objektif untuk mencegah overfitting. Fungsi objektif didefinisikan pada persamaan (1) :

$$(1) \qquad O = \sum_{i=1}^{n} L(y_i, F(x_i)) + \sum_{k=1}^{t} R(f_k) + C$$

Pada persamaan (1) terdapat beberapa penjelasan sebagai berikut :

a. $L(y_i, F(x_i))$ merupakan fungsi kerugian (*loss function*) yang digunakan untuk mengukur seberapa baik model dalam memprediksi data.

b. $R(f_k) = \alpha H + \frac{1}{2} n \sum_{j=1}^{H} \omega_j^2$ merupakan istilah regularisasi yang digunakan untuk mencegah overfitting dengan penjelasan lebih lengkap sebagai berikut :

c. α mewakili kompleksitas daun.

d. H menunjukkan jumlah daun.

e. n menunjukkan parameter penalti.

f. ω_j^2 menunjukkan hasil output dari setiap simpul daun.

g. C merupakan konstanta yang dapat dihilangkan secara efektif.(Jan Melvin Ayu Soraya Dachi & Pardomuan Sitompul, 2023).

2.4.2 *Random Forest*

Random Forest adalah metode ansambel yang kuat yang menggunakan pohon keputusan untuk membuat prediksi. Ini melibatkan membangun beberapa pohon keputusan selama pelatihan dan mengeluarkan prediksi rata-rata semua pohon untuk masalah regresi atau suara mayoritas untuk tugas klasifikasi. Algoritma memperkenalkan keacakan dalam dua cara utama: dengan memilih subset fitur acak pada setiap pemisahan dan dengan melakukan bootstrap data pelatihan (Potyka et al., 2022). Dalam algoritma *Random Forest* dapat dijelaskan dengan beberapa tahapan berikut, yaitu :

$$(2) \qquad D_{bootstrap} = \{D_i\}_{i=1}^{m}$$

Pada persamaan (2), dilakukan proses pembagian data yang mana $D_{bootstrap}$ merupakan jumlah bootstrap sample yang dihasilkan sama dengan jumlah data asli. D_i merupakan bootstrap sample yang diambil dari dataset asli.

$$(3) \qquad T_j = T(D_j, X_j, \theta_j)$$

Pada persamaan (3), dilakukan proses pembangunan pohon menggunakan bootstrap sampel. T_j merupakan pohon Keputusan yang dibangun dari bootstrap sampel D_j , variabel predictor X_j , dan parameter acak θ_j .

$$(4)\ F(X) = \frac{1}{m} \sum_{j=1}^{m} T_j(X)$$

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818

Pada persamaan (4), dilakukan proses penggabungan pohon dengan menggabungkan prediksi dari masing-masing pohon untuk mendapatkan fungsi prediksi akhir. $F(X)$ merupakan fungsi prediksi yang dihasilkan dari penggabungan semua pohon.

$$(5)\ y = \arg max_k\ F(X)$$

Pada persamaan (5), dilakukan proses prediksi dengan menggunakan fungsi prediksi yang dihasilkan. Prediksi dilakukan dengan menemukan kelas yang memiliki nilai tertinggi dalam fungsi prediksi. y merupakan adalah kelas yang diprediksi berdasarkan voting mayoritas dari semua pohon(Mahmuda, 2024).

2.5 Evaluasi

Model machine learning berupa klasifikasi digunakan untuk dapat membagi sebuah nilai kedalam beberapa kategori, dan performa model dapat dievaluasi menggunakan *confution metrix*. *Confution metrix* menghitung tingkat akurasi, recall, presisi, dan f1-score dari sebuah performa model. Semakin tinggi nilai *confution matrix* makan semakin baik performa model dalam melakukan proses klasifikasi. Rumus *confution matrix* bisa dilihat sebagai berikut :

Tabel 1. *Confution Matrix*

		True Class	
		Positive	*Negative*
Predicted Class	*Positive*	*True Positive (TP)*	*False Positive (FN)*
	Negative	*False Negative (FN)*	*True Negative (TN)*

Penjelasan :

a. Akurasi (Accuracy) = (TP + TN) / (TP+TN+FP+FN)

b. Presisi (Precision) = TP / (TP+FP)

c. Recall = TP / (TP+FN)

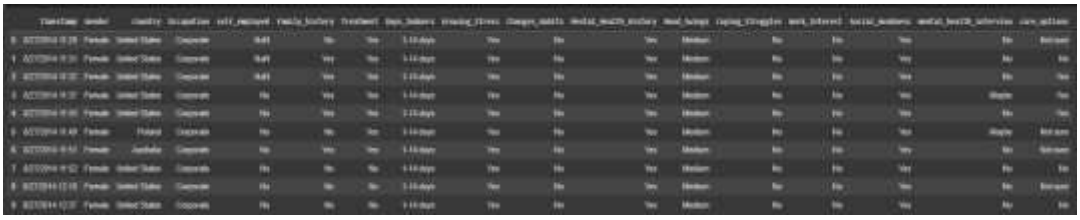
d. F1-Score = 2 x ((Presisi x Recall) / (Presisi + Recall))

3. ANALISA DAN PEMBAHASAN

Proses penelitian ini menggunakan bahasa python. Pada proses analisis data menggunakan library numpy, pandas, sklearn, matplotlib, imlearn, xgboost dan seaborn. Pandas dan numpy sebagai data analysis tools. Matplotlib dan seaborn digunakan untuk visualisasi data. Imlearn digunakan untuk proses balancing data. Kemudian yang terakhir, sklearn dan xgboost digunakan untuk machine learning. Source code pada penelitian ini dapat diakses pada website github dengan link berikut : https://github.com/AdityaAnanta123/MentalHealthAnalysis.git

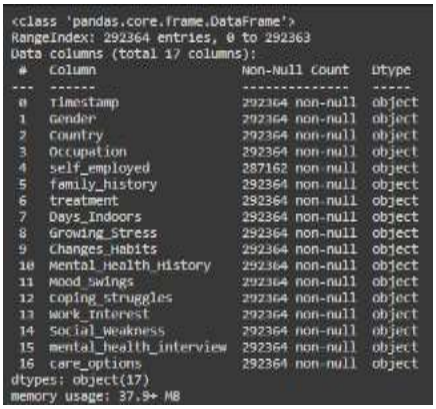
3.1 Pemanggilan dan Analisis Dataset

Proses pemanggilan dan menampilkan dataset menggunakan perintah read dan print dari library pandas. Hasil perintah tersebut terdapat pada gambar 2.



Gambar 2. Dataset Mentah Health.csv

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 292364 entries, 0 to 292363
Data columns (total 17 columns):
 #   Column                  Non-Null Count   Dtype
---  ------                  --------------   -----
 0   Timestamp               292364 non-null  object
 1   Gender                  292364 non-null  object
 2   Country                 292364 non-null  object
 3   Occupation              292364 non-null  object
 4   self_employed           287162 non-null  object
 5   family_history          292364 non-null  object
 6   treatment               292364 non-null  object
 7   Days_Indoors            292364 non-null  object
 8   Growing_Stress          292364 non-null  object
 9   Changes_Habits          292364 non-null  object
 10  Mental_Health_History   292364 non-null  object
 11  Mood_Swings             292364 non-null  object
 12  Coping_Struggles        292364 non-null  object
 13  Work_Interest           292364 non-null  object
 14  Social_Weakness         292364 non-null  object
 15  mental_health_interview 292364 non-null  object
 16  care_options            292364 non-null  object
dtypes: object(17)
memory usage: 37.9+ MB
```

Gambar 3. Fitur Dataset

Gambar 3 menjelaskan bahwa dataset memiliki jumlah 292364 baris dan 17 kolom yang memiliki tipe objek secara keseluruhan. Dalam dataset mental health 17 variabel yang merupakan tipe data objek diubah menjadi numerik menggunakan encoding.

3.2 Penanganan Missing Values

Pada tahap ini terdapat missing value dalam dataset mental health yang berjumlah 5202 data pada variabel self_employed. Penanganan missing values dilakukan dengan cara mengisi nilai default sesuai dengan nilai target yang sama. Hasil dari pengecekan sebelum dan sesudah penanganan missing value bisa dilihat pada gambar 4 dan gambar 5.

```
Timestamp                  0
Gender                     0
Country                    0
Occupation                 0
self_employed           5202
family_history             0
treatment                  0
Days_Indoors               0
Growing_Stress             0
Changes_Habits             0
Mental_Health_History      0
Mood_Swings                0
Coping_Struggles           0
Work_Interest              0
Social_Weakness            0
mental_health_interview    0
care_options               0
dtype: int64
```

Gambar 4. Missing Values

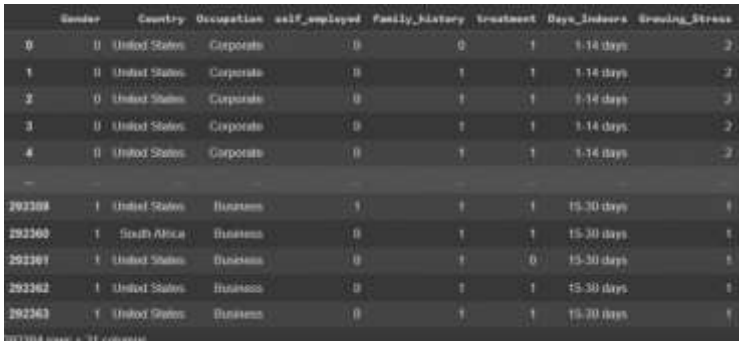
```
Timestamp                  0
Gender                     0
Country                    0
Occupation                 0
self_employed              0
family_history             0
treatment                  0
Days_Indoors               0
Growing_Stress             0
Changes_Habits             0
Mental_Health_History      0
Mood_Swings                0
Coping_Struggles           0
Work_Interest              0
Social_Weakness            0
mental_health_interview    0
care_options               0
dtype: int64
```

Gambar 5. Penanganan Missing Values

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818

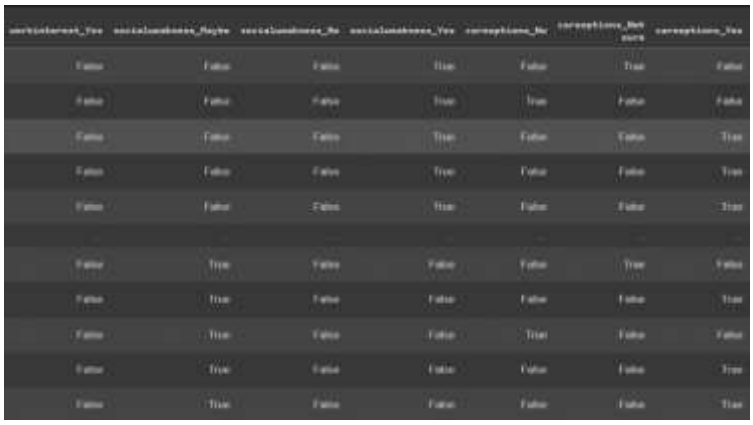
3.3 *Encoding*

Pada tahap encoding digunakan label encoding untuk fitur kategori yang sederhana seperti Gender, self_employed, family_history, treatment, Growing_Stress, Coping_Struggles, dan mental_health_interview sedangkan one hot encoding untuk fitur kategori yang lebih kompleks seperti Country, Occupation, Days_Indoors, Changes_Habits, Mental_Health_History, Work_Interest, Social_Weakness, care_options, dan Mood_Swings. Hasil pengolahannya terdapat pada gambar 6 dan gambar 7.



	Gender	Country	Occupation	self_employed	family_history	treatment	Days_Indoors	Growing_Stress
0	0	United States	Corporate	0	0	1	1-14 days	2
1	0	United States	Corporate	0	1	1	1-14 days	2
2	0	United States	Corporate	0	1	1	1-14 days	2
3	0	United States	Corporate	0	1	1	1-14 days	2
4	0	United States	Corporate	0	1	1	1-14 days	2
...								
292359	1	United States	Business	1	1	1	15-30 days	1
292360	1	South Africa	Business	0	1	1	15-30 days	1
292361	1	United States	Business	0	1	0	15-30 days	1
292362	1	United States	Business	0	1	1	15-30 days	1
292363	1	United States	Business	0	1	1	15-30 days	1

Gambar 6. *Label Encoding*

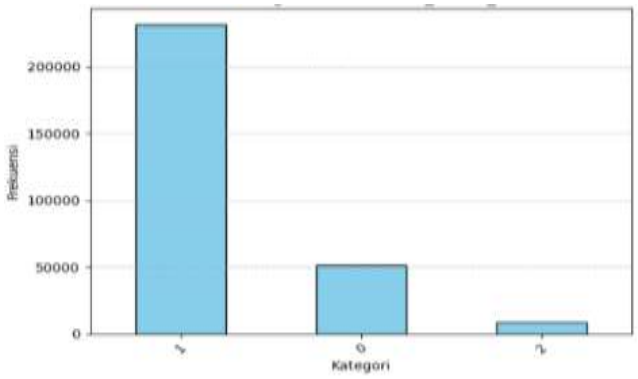


False	False	False	True	False	True	False
False	False	False	True	True	False	False
False	False	False	True	False	False	True
False	False	False	True	False	False	True
False	False	False	True	False	False	True
...						
False	True	False	False	False	True	False
False	True	False	False	False	False	True
False	True	False	False	True	False	False
False	True	False	False	False	False	True
False	True	False	False	False	False	True

Gambar 7. *One Hot Encoding*

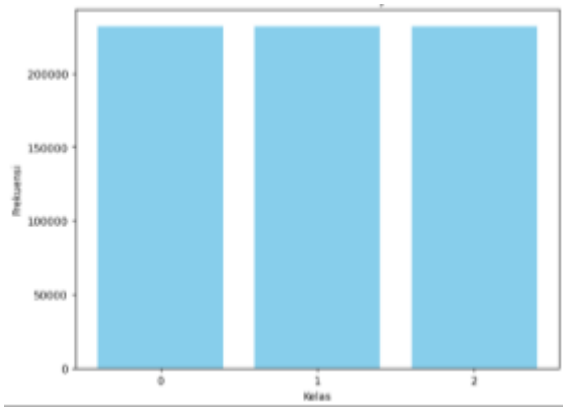
3.4 *Balancing Data*

Tahapan balancing data merupakan tahapan yang sangat penting karena bisa mempengaruhi performa model dalam memprediksi sebuah hubungan antar variabel. Tahapan balancing data menggunakan teknik SMOTE (Synthetic Minority Over-sampling Technique). Hasil balancing data dapat dilihat pada gambar 8 dan gambar 9.



Gambar 8. Sebelum *Balancing Data*

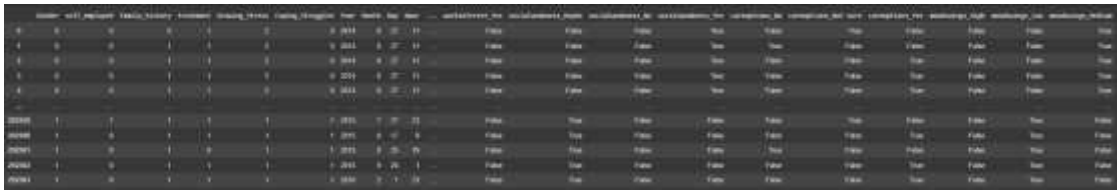
LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818



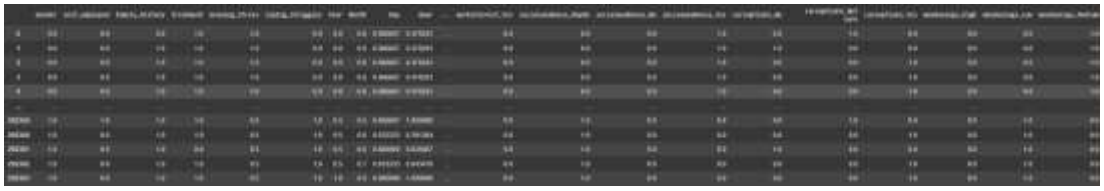
Gambar 9. Sesudah *Balancing Data*

3.5 Normalisasi Data

Pada tahapan normalisasi data menggunakan MinMaxScalar untuk mengubah nilai variabel ke rentang 0 hingga 1. Hasil Normalisasi data dapat dilihat pada gambar 10 dan gambar 11.



Gambar 10. Sebelum Normalisasi Data



Gambar 11. Sesudah Normalisasi Data

3.6 Split Data

Tahapan ini dilakukan dengan splitting data menjadi 2 variabel yaitu X dan y. Variabel X merupakan semua fitur selain mental_health_interview sedangkan variabel y merupakan fitur mental_health_interview. Test size yang digunakan pada penelitian ini 80% untuk data train dan 20% data test. Random_state yang digunakan untuk mengatur data secara random pada dataset ini yaitu 42. Pada gambar 10 merupakan hasil split data antara data train dan data test.

```
Jumlah seluruh dataset: 696498
Jumlah dataset setelah splitting data untuk training: 557198
Jumlah dataset setelah splitting data untuk testing: 139300
```

Gambar 12. Hasil Split Data

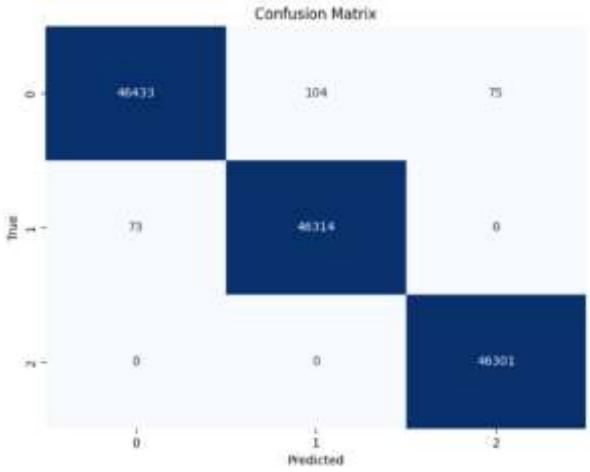
3.7 *Modelling*

Penelitian ini menggunakan 2 Algoritma model machine learning yang berbeda yaitu *Extreme Gradient Boost* (XGBoost) dan *Random Forest*. Setiap model yang diuji menggunakan parameter default yang telah disediakan oleh library sklearn.

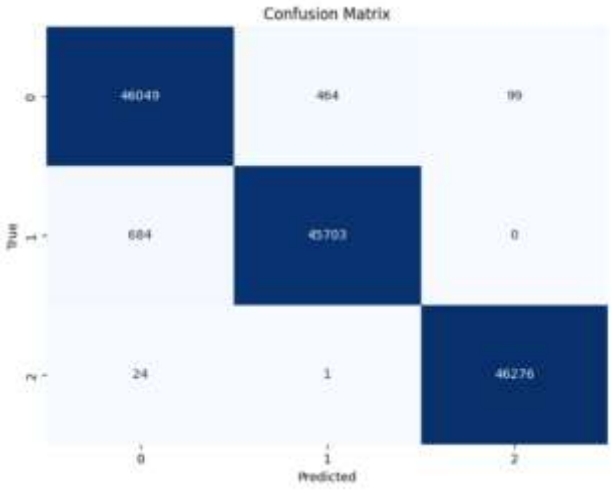
LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818

3.8 Evaluasi

Tahap evaluasi merupakan tahap yang dilakukan untuk menilai ketepatan model dalam memprediksi data sesuai dengan data nyata. Proses tersebut membutuhkan matriks yang disebut *Confution Matrix* untuk menentukan kinerja model.



Gambar 13. *Confution Matrix* XGBoost



Gambar 14. *Confution Matrix Random Forest*

Dari gambar 11 dan 12, terlihat bahwa model algoritma XGBoost memiliki kesalahan yang lebih sedikit dibandingkan dengan *Random Forest* yang memiliki kesalahan lebih banyak. Dengan adanya confution matrix dapat dilihat model terbaik dalam klasifikasi sebuah kategori dalam hal ini merupakah kesehatan mental.

Tabel 2. Performa Algoritma XGBoost dengan 30 percobaan

Percobaan ke	Accuracy	Precision	Recall	F1-Score
1	99.81%	99.81%	99.81%	99.81%
2	99.84%	99.84%	99.84%	99.84%
3	99.83%	99.83%	99.83%	99.83%
4	99.81%	99.81%	99.81%	99.81%

LOGIC : Jurnal Ilmu Komputer dan Pendidikan
Volume 2, No. 5, Agustus Tahun 2024
ISSN 2985-4172 (media online)
Hal 808-818

5	99.80%	99.80%	99.80%	99.80%
...	...	...	...	...
26	99.78%	99.78%	99.78%	99.78%
27	99.81%	99.81%	99.81%	99.81%
28	99.82%	99.82%	99.82%	99.82%
29	99.72%	99.72%	99.72%	99.72%
30	99.80%	99.80%	99.80%	99.80%
Rata-rata	99.82%	99.82%	99.82%	99.82%

Pada Algoritma XGBoost dilakukan percobaan 30 kali untuk memvalidasi performa dengan mengambil rata-rata dari percobaan yang telah dilakukan. Hasil yang didapatkan dari rata-rata percobaan tersebut adalah 99.82% untuk akurasi, presisi, recall, dan f1-score.

Tabel 3. Performa Algoritma *Random Forest* dengan 30 percobaan

Percobaan ke	Accuracy	Precision	Recall	F1-Score
1	98.93%	98.93%	98.93%	98.93%
2	99.03%	99.03%	99.03%	99.03%
3	99.09%	99.09%	99.09%	99.09%
4	98.98%	98.98%	98.98%	98.98%
5	99.06%	99.06%	99.06%	99.06%
...	...	...	...	...
26	99.05%	99.05%	99.05%	99.05%
27	99.12%	99.12%	99.12%	99.12%
28	98.92%	98.92%	98.92%	98.92%
29	99.00%	99.00%	99.00%	99.00%
30	99.02%	99.02%	99.02%	99.02%
Rata-rata	99.04%	99.04%	99.04%	99.04%

Selain pada Algoritma XGBoost, dilakukan percobaan yang sama kepada Algoritma *Random Forest* untuk mendapatkan rata-rata dari keseluruhan percobaan. Hasil yang didapatkan dari rata-rata percobaan tersebut adalah 99.04% untuk akurasi, presisi, recall dan f1-score.

4. KESIMPULAN

Dari kedua model algoritma yang telah dikembangkan dapat dilihat bahwa dari hasil perbandingan klasifikasi kedua algoritma yaitu XGBoost dan *Random Forest* dapat disimpulkan bahwa XGBoost memiliki akurasi terbesar dengan tingkat kesalahan yang begitu sedikit sedangkan *Random Forest* memiliki akurasi lebih kecil dengan tingkat kesalahan yang lebih banyak.

REFERENCES

Andriansyah, D.-, & Eka Wulansari Fridayanthie. (2023). Optimization of Support Vector Machine and XGBoost Methods Using Feature Selection to Improve Classification Performance. JOURNAL OF

INFORMATICS AND TELECOMMUNICATION ENGINEERING, 6(2), 484–493. https://doi.org/10.31289/jite.v6i2.8373

Dahouda, M. K., & Joe, I. (2021). A Deep-Learned Embedding Technique for Categorical Features Encoding. IEEE Access, 9, 114381–114391. https://doi.org/10.1109/ACCESS.2021.3104357

Data Collection and Normalization. (2023). In Reverse Osmosis 3rd Edition (pp. 419–429). Wiley. https://doi.org/10.1002/9781119725183.ch12

Doctor, K., Mao, T., & Mhaskar, H. (2024). Encoding of data sets and algorithms. Applied Numerical Mathematics, 200, 209–235. https://doi.org/10.1016/j.apnum.2023.07.013

Givari, M. R., Sulaeman, M. R., & Umaidah, Y. (2022). Perbandingan Algoritma SVM, *Random Forest* Dan XGBoost Untuk Penentuan Persetujuan Pengajuan Kredit. NUANSA INFORMATIKA, 16(1), 141–149. https://doi.org/10.25134/nuansa.v16i1.5406

Jan Melvin Ayu Soraya Dachi, & Pardomuan Sitompul. (2023). Analisis Perbandingan Algoritma XGBoost dan Algoritma *Random Forest* Ensemble Learning pada Klasifikasi Keputusan Kredit. JURNAL RISET RUMPUN MATEMATIKA DAN ILMU PENGETAHUAN ALAM, 2(2), 87–103. https://doi.org/10.55606/jurrimipa.v2i2.1470

Mahmuda, S. (2024). Implementasi Metode *Random Forest* pada Kategori Konten Kanal Youtube. JURNAL JENDELA MATEMATIKA, 2(01), 21–31. https://doi.org/10.57008/jjm.v2i01.633

Moore, R. C., Ellis, D. P. W., Fonseca, E., Hershey, S., Jansen, A., & Plakal, M. (2023). Dataset Balancing Can Hurt Model Performance. ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 1–5. https://doi.org/10.1109/ICASSP49357.2023.10095255

Potyka, N., Yin, X., & Toni, F. (2022). Explaining *Random Forest*s using Bipolar Argumentation and Markov Networks (Technical Report). http://arxiv.org/abs/2211.11699

Prakash, A., Thangaraj, J., Roy, S., Srivastav, S., & Mishra, J. K. (2023). Model-Aware XGBoost Method Towards Optimum Performance of Flexible Distributed Raman Amplifier. IEEE Photonics Journal, 15(4), 1–10. https://doi.org/10.1109/JPHOT.2023.3286272

Rosta Br Sebayang, E., Herry Chrisnanto, Y., Jenderal Achmad Yani Cimahi, U., Terusan Jend Sudirman, J., Selatan, C., & Barat, J. (2023). Klasifikasi Data Kesehatan Mental di Industri Teknologi Menggunakan Algoritma *Random Forest*. IJESPG Journal, 1(3). http://ijespgjournal.org.