PERBANDINGAN AKURASI ALGORITMA *NAÏVE BAYES* DAN ALGORITMA *XGBOOST* PADA KLASIFIKASI PENYAKIT DIABETES

Muhammad Kaddafi Nasution[1], Rd. Rohmat Saedudin[2], Vandha Pradwiyasma Widartha[3]

[1,2,3] Universitas Telkom, Bandung
mkaddafinst@student.telkomuniversity.ac.id[1], **rdrohmat@telkomuniversity.ac.id[2]**,
vandhapw@telkomuniversity.ac.id[3]

Abstrak

Diabetes merupakan penyakit yang terus meningkat dan semakin tinggi kasus kematian yang memakan korban, Penyakit kronis serius ini disebabkan oleh gangguan metabolik yang terjadi karena pankreas tidak dapat menghasilkan atau memproduksi cukup Insulin (hormon yang mengatur glukosa). Menurut *Internasional of Diabetic Federation* (IDF) tingkat prevalensi global penderita diabetes terus meningkat setiap tahunnya. Diabetes merupakan salah satu penyakit paling umum dan menjadi penyebab kematian terbesar di dunia. Pendeteksian penyakit diabetes dapat dilakukan dengan teknik *data mining*. *Data Mining* merupakan suatu proses pengumpulan informasi penting dari sebuah data yang besar pada suatu keahlian yang berkaitan dengan informatika. *Data Mining* juga dapat digunakan pada penelitian yang bergerak di aspek lainnya, salah satunya pada bagian kesehatan untuk melakukan prediksi penyakit Diabetes pada suatu kelompok individu dengan metode klasifikasi. Pada penelitian ini, *dataset* yang digunakan berasal dari *Pima Indians Diabetes Databases* (PPID). Penelitian ini bertujuan untuk melakukan perbandingan performa klasifikasi dari Algoritma *Supervised Learning*, yaitu *Naïve Bayes* dan *XGBoost*. Penelitian ini juga akan melakukan penanganan *missing value* terhadap *dataset* dan membahas mengenai metode *Grid Search* sebagai optimisasi berdasarkan kinerja akurasi klasifikasi penyakit diabetes pada Algoritma *Naïve Bayes* dan *XGBoost*. Hasil akurasi dievaluasi dengan menggunakan *confusion matrix* serta menghitung nilai AUC. Sehingga dari hasil klasifikasi, didapat model klasifikasi Algoritma *Naïve Bayes* dengan nilai hasil akurasi model sebesar 79.68% dan Algoritma *XGBoost* memiliki performa yang lebih baik dengan nilai hasil akurasi yang didapat sebesar 90.10%.

Kata Kunci: *Data Mining*, klasifikasi, Diabetes, Naive Bayes, XGBoost

Abstract

Diabetes is a disease that continues to increase, and the number of deaths is increasing. This serious chronic disease is caused by metabolic disorders that occur because the pancreas cannot produce or produce enough insulin (a hormone that regulates glucose). According to the International of Diabetic Federation (IDF), the global prevalence rate of people with diabetes continues to increase every year. Diabetes is one of the most common diseases and the biggest cause of death in the world. Detection of diabetes can be done with data mining techniques. Data Mining is a process of collecting important information from big data in an expertise related to informatics. Data Mining can also be used in research that is engaged in other aspects, one of which is in the health sector to predict Diabetes in a group of individuals using the classification method. In this study, the dataset used came from the Pima Indians Diabetes Databases (PPID). This study aims to compare the classification performance of the Supervised Learning Algorithm, namely Naïve Bayes and XGBoost. This study will also handle missing values on the dataset and discuss the Grid Search method as an optimization based on the performance of diabetes classification accuracy on the Naïve Bayes Algorithm and XGBoost. Accuracy results are evaluated by using a confusion matrix and calculating the AUC value. So, from the classification results, the Naïve Bayes Algorithm classification model is obtained with a model accuracy value of 79.68% and the XGBoost Algorithm has a better performance with an accuracy value of 90.10%.

Keywords: Data Mining, Classification, Diabetes, Naive Bayes, XGBoost

1. Pendahuluan

Diabetes merupakan penyakit yang terus meningkat dan semakin tinggi kasus kematian yang memakan korban, Penyakit kronis serius ini disebabkan karena gangguan metabolik yang terjadi karena pankreas tidak dapat menghasilkan atau memproduksi cukup Insulin (hormon yang mengatur glukosa) atau dimana ketika tubuh tidak dapat secara efektif menggunakan Insulin yang diproduksi, sehingga menghasilkan kadar gula darah tinggi atau disebut hiperglikemia [1].

Diabetes adalah penyakit tidak menular (PTM) merupakan penyebab utama kematian dan kecacatan di dunia, menjadi penyebab masalah

kesehatan masyarakat yang sangat penting dan menjadi salah satu dari empat prioritas penyakit tidak menular untuk ditindaklanjuti oleh pemerintah di seluruh dunia [2].

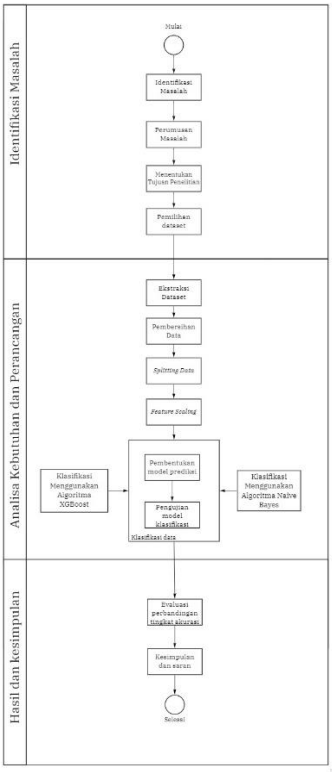
Secara global diperkirakan 463 juta orang dewasa mengidap diabetes pada tahun 2019, Menurut World Health Organization (WHO) jumlah ini meningkat drastis dibandingkan pada tahun 1980 dengan 108 juta orang dewasa mengidap diabetes dan juga pada tahun 2014 sebanyak 422 juta. Prevalensi diabetes di dunia secara global telah meningkat sejak tahun 1980. Hal ini mencerminkan peningkatan faktor risiko terkait seperti kelebihan berat badan atau obesitas. Selama beberapa dekade terakhir, prevalensi diabetes meningkat lebih cepat di negara berpenghasilan rendah dan menengah daripada di negara berpenghasilan tinggi.

Identifikasi penyakit diabetes dapat dilakukan dengan melakukan pengklasifikasian untuk membantu dalam penanggulangan penyakit diabetes, maka langkah untuk melakukan analisis dan mencari solusi yang nantinya dapat membantu untuk menurunkan angka tingkat diabetes yang semakin tinggi dengan dilakukanya klasifikasi dalam Machine learning, untuk melakukan klasifikasi dapat dilakukan dengan teknik Data Mining [3]. Data mining merupakan gabungan dari berbagai disiplin ilmu seperti statistik, matematika, pengenalan pola, gudang data (data warehouse), kecerdasan buatan, dan visualisasi data (data visualization) [4]. Saat ini banyak algoritma-algoritma yang dapat digunakan untuk melakukan metode Machine learning. *Data Mining* merupakan suatu proses pengumpulan informasi penting dari sebuah data yang besar pada suatu keahlian yang berkaitan dengan informatika. Data Mining juga dapat digunakan pada penelitian yang bergerak di aspek lainnya, salah satunya pada bagian kesehatan untuk melakukan prediksi penyakit Diabetes pada suatu kelompok individu dengan teknik klasifikasi. Teknik klasifikasi mempunyai algoritma yang beragam. Pada penelitian ini, menggunakan algoritma *Naïve Bayes* dan *XGBoost*. Naïve Bayes adalah suatu pendekatan untuk menghitung prediksi berdasarkan probabilitas dari pola klasifikasi data [5]. *XGBoost* adalah algoritma berbasis pohon yang lebih efisien dan Scalable [6].

Berdasarkan dengan permasalahan yang telah di didapat, maka pada penelitian ini akan dibangun model klasifikasi terhadap pengidap penyakit diabetes, oleh karena itu penelitian ini akan menggunakan algoritma Naïve Bayes dan algoritma XGBoost dengan tujuan untuk mengetahui sekaligus membandingkan performa akurasi dari kedua algoritma tersebut.

2. Metode Penelitian

Sistematika penyelesaian pada penelitian ini, terdapat tiga tahapan yang akan diterapkan. Tahapan tersebut dimulai dari tahap identifikasi masalah, analisa kebutuhan dan perancangan, dan yang terakhir adalah implementasi dan hasil. Seperti pada gambar 1:



Gambar 1 Tahapan penelitian

A. Identifikasi Masalah

Pada tahap ini, peneliti mengidentifikasi masalah yang ada. Selanjutnya peneliti menentukan rumusan masalah yang ada berdasarkan masalah yang sudah teridentifikasi. Selanjutnya peneliti menentukan tujuan dari penelitian yang akan dilaksanakan berdasarkan rumusan masalah yang telah dirumuskan. Langkah terakhir pada tahapan identifikasi masalah adalah pemilihan dataset yang sesuai

dengan rumusan masalah dan tujuan dari penelitian ini.

B. Analisa Kebutuhan dan Perancangan

Pada tahap ini, dilakukan pengumpulan data serta proses ekstraksi data yang digunakan. Data tersebut akan melalui tahapan data preprocessing dengan melakukan pembersihan data untuk mengganti nilai *null* (*missing value*) dengan nilai *median* dari tiap kolom pada kelasnya masing-masing, kemudian melakukan *Splitting Data* dengan membagi data menjadi data *training* dan data *testing*. Peneliti menggunakan algoritma *Naïve Bayes* dan *XGBoost* pada klasifikasi penyakit diabetes. Kemudian data akan masuk ke dalam tahapan implementasi.

C. Implementasi dan Hasil

Pada proses implementasi algoritma *Naïve Bayes* dan *XGBoost* maka akan dilakukan proses pencarian nilai akurasi dari masing-masing algoritma, dengan melakukan pengujian pada data *training* sesuai dengan rasio yang ditentukan pada setiap algoritma. Tahap selanjutnya dengan melakukan implementasi *K-fold Cross Validation* dan implementasi *Tuning Hyperparameter* dengan penentuan parameter yang akan digunakan untuk menghasilkan nilai akurasi terbaik. Setelah mendapatkan akurasi dari masing-masing algoritma, dilakukan tahapan evaluasi perbandingan akurasi menggunakan *confusion matrix* dan mengukur nilai *Precision*, *Recall*, *f1_score* serta nilai AUC pada kedua algoritma tersebut.

3. Hasil dan Pembahasan

Penelitian ini menggunakan *dataset Pima Indians Diabetes Database* yang dapat diakses secara online melalui website atapdata.ai. Untuk isi atribut *dataset* tersebut dapat dilihat seperti pada gambar 2.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1
...	...	...	...	...	...	...	...	...	...
763	10	101	76	48	180	32.9	0.171	63	0
764	2	122	70	27	0	36.8	0.340	27	0
765	5	121	72	23	112	26.2	0.245	30	0
766	1	126	60	0	0	30.1	0.349	47	1
767	1	93	70	31	0	30.4	0.315	23	0

768 rows × 9 columns

Gambar 2 Data Pima Indians Diabetes Database

Pada gambar 2 *Dataset* tersebut terdapat variabel independen dan dependen. Variabel Independen terdiri dari 8 variabel yaitu *Pregnancies*, *Glucose*, *BloodPressure*, *SkinThickness*, *Insulin*, *BMI*, *DiabetesPedigreeFunction*, *Age* dan Variabel Dependen terdiri dari 1 variabel yang berisi kan *Outcome* (*Class*). Tabel 1(a) dan tabel 1(b) merupakan keterangan dari masing-masing Variabel Independen (X) dan Variabel Dependen (y).

Tabel 1 (a) Variabel Independen (X)

Kolom *Dataset*	Deskripsi
Preg (*Pregnancies*)	Variabel ini menunjukkan tentang Berapa kali hamil
Plas (*Glucose*)	Variabel ini menunjukkan tentang konsentrasi glukosa plasma 2 jam dalam tes toleransi glukosa
Pres (*Blood Pressure*)	Variabel ini menunjukkan tentang Tekanan darah diastolik
Skin (*Skin Thickness*)	Variabel ini menunjukkan tentang Ketebalan lipatan kulit trisep
Insu (*Insulin*)	Variabel ini menunjukkan tentang Tingkat Insulin serum dalam 2 jam
BMI (*Body Mass Index*)	Variabel ini menunjukkan tentang indeks massa tubuh
Pedi (*Diabetes Pedigree Function*)	Variabel ini menunjukkan tentang indikator riwayat diabetes di dalam keluarga
Age	Variabel ini menunjukkan tentang umur seseorang

Tabel 2 (b) Variabel Dependen (y)

Kolom *Dataset*	Deskripsi
Outcome	Positif diabetes (1) dan negatif diabetes (0)

Pada tahap selanjutnya, peneliti melakukan tahap preprocessing data agar model dapat menerima atau menggunakan data tersebut dengan performa yang maksimal. Tahapan yang dilakukan peneliti yaitu dimulai dari *data cleansing* dengan mencari apakah ada data yang bernilai *null* (*missing value*) dari *dataset* yang digunakan. Setelah itu masuk ke tahapan mencari nilai median dari setiap kolom untuk mengantikan data yang bernilai *null*. Hasil dari

perhitungan nilai median dapat dilihat pada tabel 3.

Tabel 3 Nilai Median Tiap Kolom

Outcome	plas	pres	skin	insu	mass
0	107	70	27	102,5	30,1
1	140	74,5	32	169,5	34,3

Pada tabel 3 dapat dilihat bahwa, nilai median yang telah di dapatkan tersebut akan mengisi nilai *null* pada *dataset* dan data atribut yang memiliki *missing value* sudah berhasil diatasi. Sehingga data dapat masuk ke tahap selanjutnya.

Pada tahap selanjutnya, dilakukan proses pembuatan 2 *matrix* X dan y atau di lakukan pemisahan pada kedua *matrix* tersebut. Pada *matrix* tersebut terdapat 2 variabel yang dikelompokkan menjadi 2 *matrix*, dimana *matrix* X menggambarkan variabel bebas (independen), *matrix* y menjelaskan variabel terikat (dependen), dan Tahapan terakhir melakukan *splitting data* dengan rasio yang sudah di tetapkan.

Pengujian awal hasil akurasi yang dilakukan oleh peneliti pada Algoritma *Naïve Bayes* dan *XGBoost* yang didapatkan berdasarkan perbandingan rasio 75:25. pembagian data ditampilkan pada tabel 4 dan tabel 5.

Tabel 4 Hasil Akurasi Percobaan Awal dari *Naive Bayes*

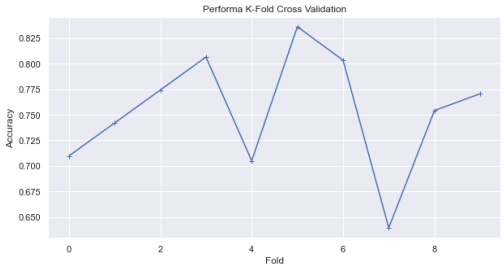
Nilai Rasio	*Data Train*	*Data Test*	Akurasi	Total Data
75:25	576	192	79.68%	768

Tabel 5 Hasil Akurasi Percobaan Awal dari *XGBoost*

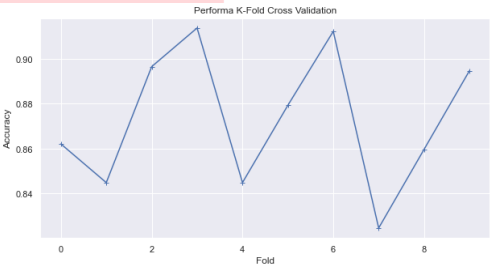
Nilai Rasio	*Data Train*	*Data Test*	Akurasi	Total Data
75:25	576	192	87.50%	768

Dapat dilihat bahwa hasil akurasi dengan rasio 75:25 pada Algoritma *Naïve Bayes* menghasilkan nilai akurasi sebesar 79.68%. Sementara hasil akurasi dari pengujian awal Algoritma *XGBoost* pada rasio 75:25 yang menghasilkan nilai akurasi sebesar 87.50%.

Dalam memvalidasi hasil pengujian klasifikasi terhadap *dataset*, dimana akan diuji menggunakan metode *K-Fold Cross Validation* dengan jumlah *fold* yang ditentukan sebanyak 10-*fold*. Hasil dari masing-masing *fold* akan dihitung nilai rata-rata sebagai hasil akurasi dari pengujian. Grafik performa *K-Fold Cross Validation* untuk pengujian terhadap kedua algoritma tersebut ini dapat dilihat pada gambar 3 dan gambar 4.



Gambar 3 Grafik *K-fold Cross Validation* dari *Naïve Bayes*



Gambar 4 Grafik *K-fold Cross Validation* dari *XGBoost*

Berdasarkan hasil pengujian *K-Fold Cross Validation* tersebut dapat dilihat hasil akurasi yang didapatkan dari 10-*fold* yang digunakan serta grafik nilai pengujian tiap *fold* nya. Untuk hasil akurasi 10-*fold* dalam *K-Fold Cross Validation* diperoleh nilai rata-rata skor dari Algoritma *Naïve Bayes* sebesar 76.23% dan untuk Algoritma *XGBoost* sebesar 87.33%. Untuk mendapatkan peningkatan kinerja akurasi, peneliti menerapkan *Tuning hyperparameter* dari model yang sudah dibuat dengan menggunakan metode *Grid Search Cross Validation*. Pada tahap ini dilakukan analisis untuk mendapatkan pola sekuensial yang akan diuji. Pola dalam bentuk grid memungkinkan perumusan *Hyperparameter* yang tepat untuk tingkat akurasi yang sesuai. Pada Algoritma *Naïve Bayes*, *Hyperparameter* yang digunakan pada metode *Grid Search* CV adalah var_smoothing dengan default = 1e-09.

Hyperparameter yang akan dikonfigurasi untuk *XGBoost* adalah kedalaman pohon (*max_depth*), Jumlah minimum berat instance yang dibutuhkan (*min_child_weight*), rasio subsample rasio *subsampel* untuk proses pelatihan (*subsample*) dan rasio *subsampel* kolom saat membangun setiap pohon

(colsample_by*tree*). *Hyperparameter* lain yang dapat disesuaikan antara lain jumlah putaran boosting (*n_estimators*), nilai *regularisation* (*gamma*), dan mengatur seberapa influence dari setiap *decision tree* terhadap total prediksi (*learning rate*). Konfigurasi *Hyperparameter* yang digunakan ditampilkan pada tabel. V.6 [7]. dan hasil kedua algoritma dari pengujian *Tuning Hyperparameter* dengan menggunakan *Grid Search* CV ditampilkan tabel 7 dan tabel 8.

Tabel 6 *XGBoost Hyperparameter configuration*

No	*Hyperparameter*	Nilai
1	*n_estimators*	100
2	*eta*	0.1
3	*gamma*	0
4	*max_depth*	3
5	*min_child_weight*	5
6	*subsample*	0.7
7	*colsample_bytree*	1
8	*boost_type*	gb*tree*

Tabel 7 Akurasi setelah *Tuning Hyperparameter* dari *Naïve Bayes*

Nilai Rasio	*Data Train*	*Data Test*	Akurasi	Total Data
75:25	576	192	79.68%	768

Tabel 8 Akurasi setelah *Tuning Hyperparameter* dari *XGBoost*

Nilai Rasio	*Data Train*	*Data Test*	Akurasi	Total Data
75:25	576	192	90.10%	768

Pada tabel 7 dan tabel 8 dapat dilihat bahwa hasil akurasi dari Algoritma *Naïve Bayes* terdapat pada rasio 75:25 dengan menerapkan *tunning hyperparameter* menghasilkan nilai akurasi sebesar 79.68%. Sementara hasil akurasi dari Algoritma *XGBoost* dengan menerapkan *tunning hyperparameter* terdapat pada rasio 75:25 yang menghasilkan nilai akurasi sebesar 90.10%.

Berdasarkan hasil akurasi dari pengujian model yang dibuat, maka selanjutnya akan mengukur performa model menggunakan *Confusion Matrix* dengan hasil yang dapat dilihat pada tabel 9 dan tabel 10.

Tabel 9 *Confusion matrix* dari *Naïve Bayes*

Klasifikasi	*Predicted Healthy* (*False*)	*Predicted Diabetic* (*True*)
Actual Healthy (*False*)	TN = 103	FP = 20
Actual Diabetic (*True*)	FN = 19	TP = 50

Tabel 10 *Confusion matrix* dari *XGBoost*

Klasifikasi	*Predicted Healthy* (*False*)	*Predicted Diabetic* (*True*)
Actual Healthy (*False*)	TN = 115	FP = 8
Actual Diabetic (*True*)	FN = 11	TP = 58

a) Didapatkan hasil pengujian algoritma *Naïve Bayes* menghasilkan nilai *True Negative* (TN) dengan jumlah 103, nilai *False Negative* (FN) dengan jumlah 19, nilai *False Positive* (FP) dengan jumlah 20, dan nilai *True Positive* (TP) dengan jumlah 50.
b) Didapatkan hasil pengujian algoritma *XGBoost* menghasilkan nilai *True Negative* (TN) dengan jumlah 115, nilai *False Negative* (FN) dengan jumlah 11, nilai *False Positive* (FP) dengan jumlah 8, dan nilai *True Positive* (TP) dengan jumlah 58.

Berdasarkan hasil yang di dapat dari *Confusion Matrix* terhadap kedua Algoritma tersebut. peneliti menghitung nilai akurasi dengan persamaan berikut [8]:

$$accuracy = \frac{TN + TP}{TN + FN + FP + TP}$$

Untuk mengevaluasi pengklasifikasian dengan mengukur nilai *Precision* dan *Recall*. *Precision* adalah parameter ketepatan dari proses klasifikasi atau proporsi klasifikasi *Positive* dari hasil prediksi yang benar terhadap keseluruhan hasil prediksi yang bernilai *Positive*. Persamaan dari nilai *precision* adalah sebagai berikut [8]:

$$precision = \frac{TP}{TP + FP}$$

Sementara *recall* merupakan ukuran pada *completeness* pada proses klasifikasi *positive* dari hasil prediksi yang benar (*true*) terhadap keseluruhan kelas aktual bernilai *positive*. Persamaan dari nilai *recall* adalah sebagai berikut:

$$recall = \frac{TP}{TP + FN}$$

Terdapat kombinasi antara nilai precision dan recall yang digunakan untuk mendapatkan nilai dari *F1-Score* dengan menggunakan persamaan dari F1-Score [8]. Persamaan dari F1-Score adalah sebagai berikut:

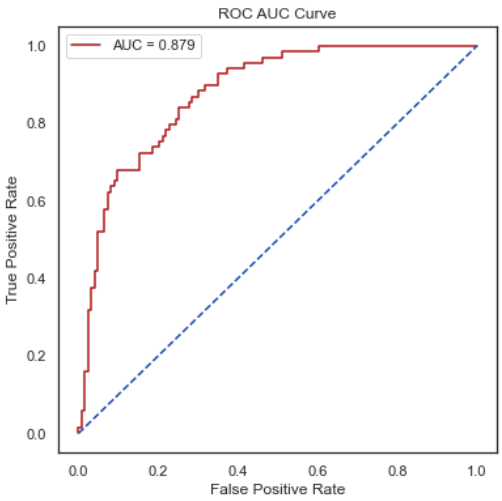
$$F1 = 2.\frac{Presisi * Recall}{Presisi + Recall}$$

Untuk hasil dari perhitungan *Confusion Matrix* dari Algoritma *Naïve Bayes* dan *XGBoost* ditampilkan pada tabel 11.

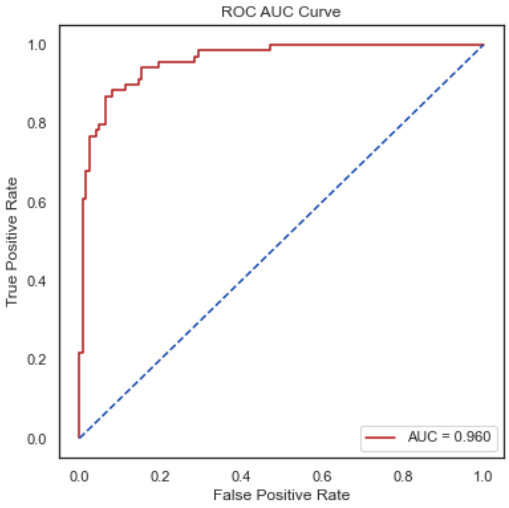
Tabel 11 Perbandingan Hasil *Confusion Matrix*

Hasil Akurasi dengan rasio 75:25	Algoritma	
	Naïve Bayes	*XGBoost*
Akurasi *Tuning Hyperparameter*	79.68%	90.10%
Precision	71.42%	87.87%
Recall	72.46%	84.05%
F1-*Score*	71.94%	85.92%

Setelah melakukan pengukuran pada *Confusion Matrix*, tahapan selanjutnya adalah membuat ROC *Curve* berdasarkan antara nilai *False Positive* dengan *True Positive*. ROC adalah grafik yang menjadikan hasil dari *False Positive* sebagai garis horizontal dan hasil dari *True Positive* untuk mengukur perbedaan performansi algoritma klasifikasi yang digunakan, dan ROC biasanya digunakan untuk mengekspresikan *Confusion Matrix*. Pada ROC *Curve* juga dapat dilakukan perhitungan skor *Area Under the Curve* (AUC). Untuk hasil grafik dari ROC dan hasil skor AUC dari Algoritma *Naïve Bayes* dan Algoritma *XGBoost* dapat dilihat pada gambar 5 dan gambar 6.



Gambar 5 Kurva ROC dari *Naïve Bayes*



Gambar 6 Kurva ROC dari *XGBoost*

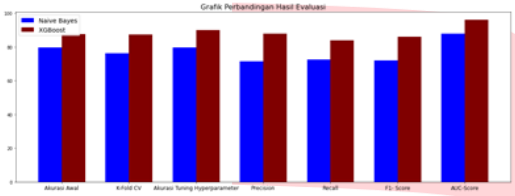
Pada gambar 5 menunjukan Kurva ROC yang dibentuk dari Algoritma *Naïve Bayes* menghasilkan nilai AUC sebesar 0.861, berdasarkan tabel klasifikasi performa nilai yang di hasilkan adalah baik karena berada di atas 80% [9][10].

Sedangkan pada gambar 6 menjelaskan Kurva ROC yang dibentuk dari Algoritma *XGBoost* menghasilkan nilai AUC sebesar 0.976, berdasarkan tabel klasifikasi performa nilai yang dihasilkan adalah sangat baik karena berada di atas 90%[9][10].

Dari hasil yang didapat terhadap pengujian akurasi dan evaluasi pada model Algoritma *Naïve Bayes* dan *XGBoost*, peneliti melakukan perbandingan performa dari kedua algoritma tersebut dan hasil yang didapatkan dapat dilihat pada tabel 12 serta grafik perbandingan hasil evaluasi dalam bentuk *Bar Chart* pada Gambar 9.

Tabel 12 Perbandingan Hasil Evaluasi

Hasil Akurasi dengan rasio 75:25	Algoritma	
	Naïve Bayes	*XGBoost*
Akurasi Awal	79.68%	87.50%
K-Fold CV	76.23%	87.33%
Akurasi *Tuning Hyperparameter*	79.68%	90.10%
Precision	71.42%	87.87%
Recall	72.46%	84.05%
F1-*Score*	71.94%	85.92%
AUC *Score*	0.879	0.960



Gambar 7 Grafik Perbandingan Akurasi

Berdasarkan hasil perbandingan terhadap kedua Algoritma, dapat dijelaskan bahwa hasil Algoritma *Naïve Bayes* menghasilkan nilai akurasi setelah *Tuning* dengan rasio *splitting data* 75:25 sebesar 79.68% dengan hasil *Precision* bernilai 71.42%, *Recall* bernilai 72.46%, *F1-Score* bernilai 71.94%, dan serta skor AUC sebesar 0.879. Algoritma *XGBoost* menghasilkan nilai akurasi setelah *Tuning* dengan rasio *splitting data* 75:25 sebesar 90.10% dengan hasil *Precision* bernilai 87.87%, *Recall* bernilai 84.05%, *F1-Score* bernilai 85.92%, dan skor AUC sebesar 0.960. Dengan hasil yang telah didapatkan tersebut, pada klasifikasi terhadap *dataset* penyakit diabetes dapat disimpulkan bahwa Algoritma *XGBoost* menghasilkan nilai akurasi dan performa yang lebih baik dibandingkan Algoritma *Naïve Bayes*. Hal tersebut dapat dilihat dari hasil akurasi terbaik dari Algoritma *XGBoost* sebesar 90.10% setelah dilakukan proses *tunning hyperparameter* dengan adanya peningkatan akurasi terhadapa Algoritma XGBoost. sedangkan Algoritma *Naïve Bayes* memiliki nilai akurasi terbaik sebesar 79.68%.

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan dapat disimpulkan hasil perbandingan akurasi Algoritma *Naïve Bayes* dan Algoritma *XGBoost* untuk klasifikasi penyakit diabetes, pada Algoritma *Naïve Bayes* menghasilkan nilai akurasi sebesar 79.68% dan pada Algoritma *XGBoost* dengan nilai yang didapatkan sebesar 90.10%. Sementara pada perhitungan nilai AUC diperoleh nilai dari Algoritma *Naïve Bayes* sebesar 0.879 dan Algoritma XGBoots sebesar 0.960. Hasil dari kedua Algoritma tersebut menunjukan bahwa Algoritma *XGBoost* menghasilkan nilai akurasi dan performa yang lebih tinggi dibandingkan Algoritma *Naïve Bayes* dalam melakukan klasifikasi terhadap data penyakit diabetes.

Referensi

[1] A. Yuniastuti, R. Susanti, and R. S. Iswari, "Efek Infusa Umbi Garut (Marantha arundinaceae L) Terhadap Kadar Glukosa dan Insulin Plasma Tikus yang Diinduksi Streptozotocyn," *J. Mipa*, vol. 41, no. 1, pp. 34–39, 2018.

[2] World Health Organization, "Classification of diabetes mellitus". Internet: https://apps.who.int/iris/handle/10665/325182, Oct. 25, 2019 [June. 29, 2021]

[3] Isbandiyo, "Artikel Ilmiah Penerapan Sequential Methods untuk Handling Missing Value pada Algoritma C4.55 dan Naive Bayes untuk memprediksi penyakit Diabetes Mellitus", *Tesis*. 2016

[4] Han & Kamber, "Data Mining: Concepts and Techniques", *2nd edn. Elsevier*, 2018.

[5] A. Ridwan, "Penerapan Algoritma Naïve Bayes Untuk Klasifikasi Penyakit Diabetes Mellitus," *J. SISKOM-KB (Sistem Komput. dan Kecerdasan Buatan)*, vol. 4, no. 1, pp. 15–21, 2020, doi: 10.47970/siskom-kb.v4i1.169.

[6] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, vol. 13-17-Augu, pp. 785–794, 2016, doi: 10.1145/2939672.2939785.

[7] I. Maalik, W. A. Kusuma, and S. Wahjuni, "Comparation Analysis of Ensemble Technique With Boosting(Xgboost) and Bagging(Randomforest) for Classify Splice Junction Dna Sequence Category," *J. Penelit. Pos dan Inform.*, vol. 9, no. 1, pp. 27–36, 2019, doi: 10.17933/jppi.2019.090.

[8] I. M. A. Agastya, "Pengaruh Stemmer Bahasa Indonesia Terhadap Peforma Analisis Sentimen Terjemahan Ulasan Film", *J. Tekno Kompak*, vol. 12, no. 1, p. 18, 2018, doi: 10.33365/jtk.v12i1.70.

[9] A. A. Abdillah and Suwarno, "Diagnosis of

diabetes using support vector machines with radial basis function kernels," *Int. J. Technol.*, vol. 7, no. 5, pp. 849–858, 2016, doi: 10.14716/ijtech.v7i5.1370.

[10] Gorunescu, F, "Data Mining: Concepts, models and techniques", *Springer Science & Business Media*, vol. 12, 2011.