Terbit *online* pada laman : http://teknosi.fti.unand.ac.id/

Jurnal Nasional Teknologi dan Sistem Informasi

| ISSN (Print) 2460-3465 | ISSN (Online) 2476-8812 |

Artikel Penelitian

Komparasi Algoritma Naïve Bayes dan Gradient Boosting untuk Prediksi Pasien Diabetes

Nova Christina Sari [a,*], *Triana Linda Larasati* [b]

[a] *Teknologi Informasi, Universitas Muhammadiyah Semarang, Kota Semarang, Indonesia*
[b] *Klinik Simpang Jawo, Kota Jambi, Indonesia*

INFORMASI ARTIKEL

Sejarah Artikel:
Diterima Redaksi: 14 Juli 2024
Revisi Akhir: 16 Agustus 2024
Diterbitkan *Online*: 31 Agustus 2024

KATA KUNCI

Diabetes,
Gradient Boosting,
Machine Learning,
Naïve Bayes

KORESPONDENSI

E-mail: novachristinasari@unimus.ac.id

A B S T R A C T

Diabetes mellitus diperkirakan semakin meningkat seiring bertambahnya usia penduduk dari 19,9%, menjadi 111,2 juta orang diusia 65-79 tahun, diprediksikan bahwa penderita diabetes akan terus meningkat hingga 578 juta orang pada tahun 2030 kemudian 700 juta ditahun 2045. *Machine learning* atau pembelajaran mesin merupakan salah satu kecerdasan buatan yang bertujuan untuk memahami atau mengenali suatu struktur suatu data dan mengonversi data tersebut kedalam suatu model. Penggunaan *Machine learning* dalam dunia kesehatan semakin pesat, semakin banyak peneliti kesehatan menggunakan algoritma *machine learning* untuk penelitiannya. Sebagian algoritma *machine learning* dapat digunakan untuk melakukan prediksi, salah satunya adalah algoritma klasifikasi untuk prediksi penyakit diabetes. Berdasarkan hasil komparasi dari beberapa algoritma yang digunakan, algoritma klasifikasi *naive bayes* dan *gradient boosting* memiliki nilai yang terbaik dari algoritma lainnya. Algoritma *gradient boosting* memiliki hasil yang tinggi terhadap nilai *accuracy* 77.09% dan *f-measure* 83.39% pada sampel *linear*. *Naive bayes* menghasilkan nilai yang terbaik terhadap pengujian sampel acak, dengan nilai *accuracy* 76.57% dan nilai *f-measure* 82.82%. Hasil pengujian sampel berlapis *(stratified)* yang memiliki nilai pada akurasi tertinggi terdapat pada algoritma *gradient boosting* dengan nilai *accuracy* 77.34% dan *f-measure* 83.39%.

1. PENDAHULUAN

Menurut International Diabetes Federation (IDF) terdapat 537 juta penderita diabetes didunia pada tahun 2021. Negara Indonesia menduduki peringkat ke lima sebagai negara dengan jumlah penduduk yang memiliki diabetes tinggi didunia pada tahun 2021. Bedasrkan jenis kelamin, IDF memperhitungkan bahwa prevalensi diabetes pada wanita ditahun 2019 9% sedangkan 9,65% terdapat pada pria. Umumnya diabetes mellitus diperkirakan semakin meningkat seiring bertambahnya usia penduduk dari 19,9%, menjadi 111,2 juta orang diusia 65-79 tahun, diprediksikan bahwa penderita diabetes akan terus meningkat hingga 578 juta orang pada tahun 2030 kemudian 700 juta ditahun 2045 [1].

Diabetes Melitus termasuk suatu penyakit yang merupakan indung atau inangnya dari segala penyakit yang ada di dalam tubuh manusia pada umumnya [2]. Diabetes Mellitus bisa mengakibatkan berbagai jenis penyakit lainnya. Komplikasi penyakit ini bisa timbul dari kepala hingga kaki, mulai dari penyakit jantung dan stroke, gagal ginjal yang menyengsarakan, hingga infeksi terutama pada kaki yang bisa berlanjut pada amputasi dan semua pada akhirnya bisa merengut nyawa [3]. Meningkatnya penderita diabetes bisa terjadi akibat dari pola hidup yang tidak sehat. Beberapa masyarakat tidak menyadari bahwa pola hidup yang tidak sehat dapat menyebabkan penyakit diabetes, pola hidup tidak sehat yang paling berpengaruh dalam meningkatnya penyakit diabetes berasal dari makanan dan kurangnya berolahraga [4].

Machine learning atau pembelajaran mesin merupakan salah satu kecerdasan buatan yang bertujuan untuk memahami atau

https://doi.org/10.25077/TEKNOSI.v10i2.2024.118-125

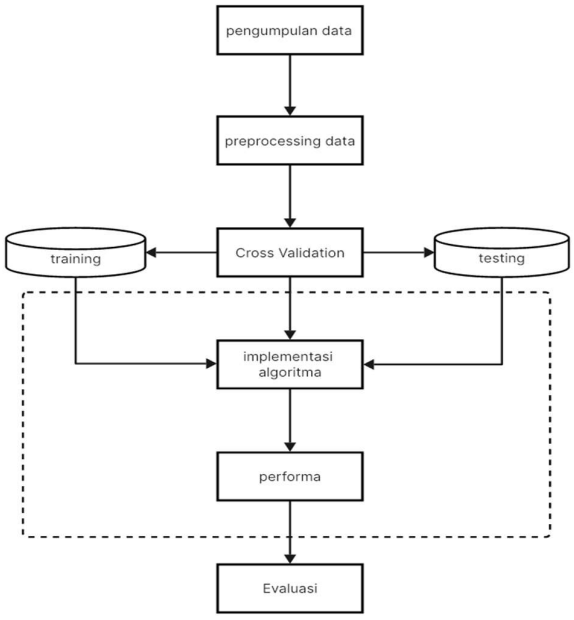
mengenali suatu struktur suatu data dan mengonversi data tersebut kedalam suatu model [5]. Penggunaan *Machine learning* dalam dunia kesehatan semakin pesat, semakin banyak peneliti kesehatan menggunakan algoritma *machine learning* untuk penelitiannya[6]. Sebagian algoritma *machine learning* dapat digunakan untuk melakukan prediksi, salah satunya adalah algoritma klasifikasi untuk prediksi penyakit diabetes [7].

Prediksi dini dan keakuratan terhadap kemungkinan perkembangan diabetes sangat penting untuk intervensi yang lebih efektif dan pengelolaan penyakit ini. Penerapan algoritma *machine learning* telah digunakan untuk penelitian, terdapat beberapa tantangan dalam menentukan algoritma yang terbaik untuk memberikan prediksi paling akurat dan andal di berbagai kondisi data. Penelitian ini bertujuan untuk mengatasi kesenjangan dengan mengidentifikasi dan membandingkan kinerja beberapa algoritma *machine learning* terutama pada algoritma *Naïve Bayes* dan *Gradient Boosting* dalam memprediksi diabetes, dengan menggunakan berbagai *sampling data*. Tujuan yang difokuskan dalam penelitian ini adalah untuk menentukan algoritma yang paling efektif dalam memberikan prediksi yang akurat dan handal terhadap pasien diabetes, dan menganalisis kinerja algoritma dengan menggunakan *sampling data* yang berbeda. Terdapat 3 penelitian terdahulu tentang prediksi diabetes menggunakan machine learning. Penelitian terdahulu tentang prediksi diabetes menggunakan machine learning [8], menghasilkan nilai pada algoritma *Decision Tree* (C4.5) dengan nilai akurasi 69%, *precision* 67%, *recall* 69%, *F1-Score* 64% dan pada algoritma *naïve bayes* menghasilkan nilai akurasi 78%, *precision* 77%, *recall* 78% dan F1-*Score* 77%. Penelitian dengan algoritma random forest [9] menghasilkan akurasi sebesar 99.3%, *recall* sebesar 99.5%, presisi sebesar 99.1%, dan F1-score sebesar 99%. Penelitian yang menggunakan algoritma gradient boosting [10] menghasilkan nilai akurasi 81%, *precision* 67%, *recall* 83%, F1-*Score* 74%. Ketiga penelitian tersebut menggunakan cross validation tetapi data yang digunakan tidak bervariasi. Data yang digunakan hanya menggunakan satu hingga dua sampel.

Penilitian yang akan dilakukan menggunakan algoritma kalsifikasi *machine learning* yaitu, *decision tree*, *naïve bayes*, *random forest* dan *gradient boosting*. Metode yang digunakan sama dengan penelitian sebelumnya. Penelitian ini menggunakan *k-fold cross validation* untuk mengevaluasi performa metode *machine learning* yang diterapkan. *Cross validation* yang digunakan bernilai 10, yaitu dengan membagi dataset menjadi 10 bagian. Pengujian penelitian ini akan dilakukan menggunakan tiga sampel, sampel linear, sampel berlapis dan sampel acak.

2. METODE

Metode pada penelitian ini adalah dengan menerapkan algoritma kalsifikasi yang terdiri dari *decision tree, naïve bayes*, *random forest* dan *gradient boosting*. Tahapan penelitian dapat dilihat pada gambar 1.



Gambar 1. Tahapan Penelitian

2.1. Pengumpulan Data

Data yang digunakan berasal dari Kaggle dengan melakukan beberapa perubahan yang disesuaikan dengan kebutuhan untuk penelitian. Data diabetes berjumlah 768 data dengan 6 atribut.

Jenis Kelamin	Usia	Tensi Darah	Glukosa	BMI	Hasil
polynominal	*integer*	*integer*	*integer*	*real*	*polynominal*
P	50	72	148	33.600	Positif
L	31	66	85	26.600	Negatif
P	32	64	183	23.300	Positif
P	21	66	89	28.100	Negatif
L	33	40	137	43.100	Positif
P	30	74	116	25.600	Negatif

Gambar 2. Tabel Data Diabetes

2.2. Preprocessing

Preprocessing dilakukan untuk merubah data awal sehingga data yang sebelumnya memiliki variabel dan atribut yang tidak sesuai menjadi sesuai dan dapat diproses sesuai dengan kebutuhan.

2.2.1. Data Cleaning

Data cleaning adalah proses mendeteksi, memperbaiki dan menghapus catatan, tabel dan database yang salah atau tidak akurat [11]. Saat melakukan prepocessing data, terdapat 1 missing data pada atribut BMI. Data cleaning dilakukan untuk menghilangkan *missing value* yang terdapat pada dataset diabetes.

Name	Type	Missing
Jenis Kelamin	Polynominal	0
Usia	Integer	0
Tensi Darah	Integer	0
Glukosa	Integer	0
BMI	Integer	0
Hasil	Polynominal	0

Gambar 3. Hasil Data Cleaning

2.3. *Processing*

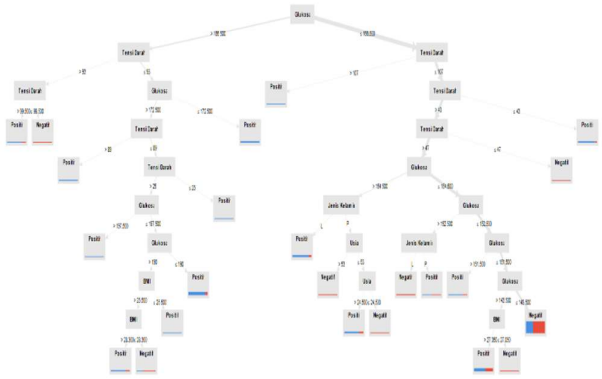
Proses data dilakukan dengan beberapa pengujian *sampling*. Pengujian *sampling* yang dilakukan berupa *linear,* sampel acak *(shuffled)* dan sampel berlapis *(stratified)* dengan menggunakan model algoritma yang telah ditentukan.

2.3.1. *Cross Validation*

Cross validation merupakan metode yang digunakan untuk pemisahan data, sehingga data dapat diproses untuk mendapatkan metrik yang lebih banyak dan lebih baik [12]. Metode *cross validation* secara umumnya digunakan dalam permodelan statistik dan pembelajaran mesin. Hasil evaluasi dari *cross validation* bersifat prediktif. Penelitian ini menggunakan *cross validation* dengan nilai K = 10.

2.3.2. *Decision Tree*

Metode pohon keputusan digunakan untuk memprediksi nilai kelas untuk menjadi variabel nya berdasarkan dari nilai entropi yang tertinggi [13]. Untuk menentukan akar dari pohon keputusan memilih atribut dengan cara menghitung nilai gain dari masing-masing atribut, nilai gain yang paling tinggi yang akan menjadi akar pertama [14].



Gambar 4. Pohon Keputusan

Decision tree memisahkan *dataset* menjadi sub grup yang semakin kecil dan homogen berdasarkan pada fitur-fitur input yang tersedia dan menghasilkan output sesuai dengan nilai target. Penggunaan algoritma pohon keputusan terhadap dataset yang digunakan menghasilkan pohon keputusan yang diujikan dengan menggunakan *filter examples* terhadap data hasil Pohon keputusan dapat dilihat pada gambar 4.

2.3.3. *Naive Bayes*

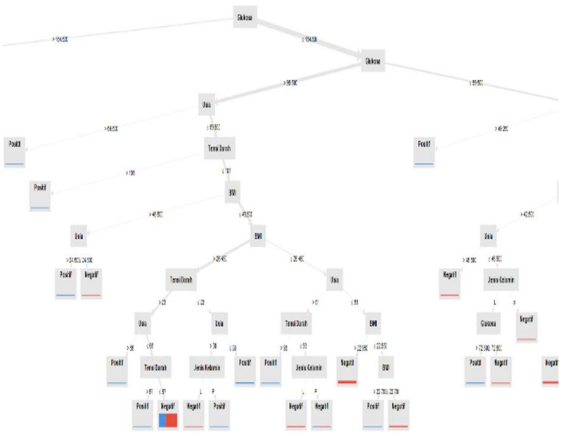
Naïve bayes adalah algoritma klasifikasi pada *machine learning. Naive Baye*s dapat digunakan untuk klasifikasi dengan menghitung probabilitas kelas tertentu berdasarkan dari nilai fitur [15]. Perhitungan probabilitas berdasarkan kelas, didasarkan pada model perhitungan probabilitas posterior dan melakukan pemilihan kelas berdasarkan nilai probabilitas posterior tertinggi.

Attribute	Parameter	Positif	Negatif
Jenis Kelamin	valueP	0.478	0.426
Jenis Kelamin	valueL	0.522	0.574
Jenis Kelamin	valueunknown	0.000	0.000
Usia	mean	37.067	31.159
Usia	standard deviation	10.968	11.668
Tensi Darah	mean	70.826	68.184
Tensi Darah	standard deviation	21.492	18.063
Glukosa	mean	141.267	109.980
Glukosa	standard deviation	31.940	26.141
BMI	mean	35.143	30.284
BMI	standard deviation	7.253	7.655

Gambar 5. Data Distribusi

2.3.4. *Random Forest*

Random Forest merupakan algoritma pembelajaran mesin yang sangat efektif untuk melakukan klasifikasi dan regresi. Algoritma *random forest* bekerja dengan membangun sejumlah besar pohon keputusan selama pelatihan dan menggabungkan hasilnya untuk meningkatkan akurasi dan prediksi untuk menghindari *overfitting* [16], [17].



Gambar 6. *Random Forest Tree*

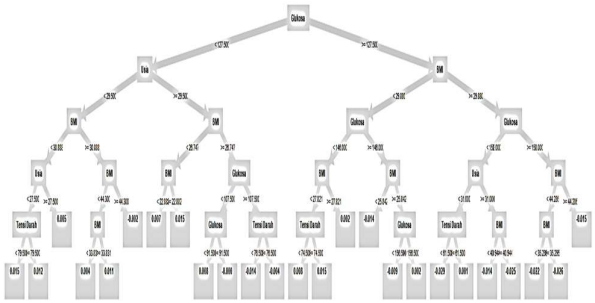
Random Feature Selection merupakan langkah untuk yang ada setiap *node* dalam pohon keputusan[18], fitur yang mempertimbangkan data yang dapat displit dan dipilih secara acak, yang meningkatkan variasi antar pohon dan mengurangi korelasi antar pohon[19].

2.3.5. *Gradient Boosting*

Gradient Boosting merupakan algoritma klasifikasi yang terdapat pada *machine learning*. Algoritma *gradient boosting* bekerja dengan membangun model secara bertahap, setiap model baru menghasilkan tahapan untuk mencoba memeperbaiki kesalah dari model sebelumnya. *Gradient boosting* merupakan algoritma yang sangat baik untuk melakukan prediksi terhadap suatu penyakit

https://doi.org/10.25077/TEKNOSI.v10i2.2024.118-125

[20][21]. Memiliki kemampuan mengatasi *outliers* dan meningkatkan akurasi prediksi melalaui model bertingkat. Pohon *gradient booster* dari data hasil dapat dilihat pada gambar 7.



Gambar 7. *Gradient Boosting Tree*

2.4. Evaluasi

Pada tahapan ini, setelah dilakukan proses data dengan menerapkan implementasi pada masing-masing algoritma akan menghasilkan beberapa attribut yaitu, *confusion matrix, accuracy, precision, recall* dan *f-measure.*

2.4.1. Confusion Matrix

Confusion matrix digunakan sebagai alat untuk evaluasi yang digunakan untuk mengukur performa algoritma klasifikasi [22]. Matriks ini memberikan gambaran yang lebih mendetail tentang prediksi yang benar dan salah dibandingkan hanya menggunakan akurasi. *Confusion matrix* memberikan gambaran yang mendetail tentang performa model klasifikasi dengan menampilkan jumlah prediksi benar dan salah untuk setiap kelas. Melakukan analisis dengan *confusion matrix* untuk menghitung berbagai metrik evaluasi yang penting seperti akurasi, presisi, *recall*, dan *F-measure* [23]. Metrik-metrik ini memberikan pemahaman yang lebih komprehensif tentang kemampuan model dalam memprediksi penyakit, dalam konteks dataset yang tidak seimbang.

Tabel 1. *Confusion Matrix*

	Predicted Positive	***Predicted Negative***
Actual Positive	TP	FN
Actual Negative	FP	TN

2.4.1 Accuracy

Akurasi adalah salah satu metrik evaluasi yang digunakan untuk mengukur performa model didalam *machine learning*[24]. Akurasi mengukur seberapa sering model membuat prediksi yang benar. Secara matematis, akurasi didefinisikan sebagai:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Akurasi adalah metrik yang umum digunakan, akurasi dapat digunakan dalam beberapa situasi yang bukan termasuk metrik terbaik untuk mengevaluasi model. Akurasi dapat digunakan untuk mengatasi *Imbalanced Classes*. Akurasi digunakan untuk dataset memiliki distribusi kelas yang tidak seimbang. Akurasi dapat digunakan sebagai model untuk konsekuensi berbeda dari kesalahan. Dalam beberapa aplikasi, kesalahan tertentu mungkin lebih serius daripada kesalahan lainnya, contoh, dalam diagnosa medis, kesalahan dalam memprediksi penyakit serius *(false negative)* bisa jauh lebih kritis dibandingkan *false positive.*

2.4.2 Precision

Precision merupakan metrik yang digunakan untuk mengevaluasi ketepatan dari prediksi *positif* yang dibuat oleh model. *Precision* memberikan informasi tentang seberapa banyak dari prediksi yang diklasifikasikan sebagai *true positif*. Perhitungan *precision* dalam *confusion matrix*, perlu memahami elemen-elemen dalam *confusion matrix*. *Precision* adalah rasio dari jumlah prediksi *true positif* terhadap jumlah total prediksi *false positif*. *Precision* berfokus pada seberapa akurat prediksi *positif* yang dibuat oleh model. *Precision* dapat dihitung menggunakan rumus sebagai berikut :

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

2.4.3 Recall

Recall merupakan metrik yang digunakan untuk mengukur seberapa baik model *machine learning* dapat menemukan semua contoh *true positif* yang terdapat di*dataset*. *Recall* juga dikenal sebagai sensitivitas atau *true positive rate* (TPR). *Recall* adalah rasio dari jumlah prediksi *true positif* terhadap jumlah total dari contoh *true positif*. *Recall* berfokus pada seberapa baik model dalam menemukan semua contoh *positif* yang ada. *Recall* dihitung menggunakan rumus :

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

2.4.4 F-measure

F-measure adalah metrik evaluasi yang digunakan untuk mengukur keseimbangan antara *precision* dan *recall* dalam model klasifikasi. *F-measure* dapat memberikan penilaian yang lebih baik ketika ada ketidak seimbangan antara jumlah kelas *positif* dan *negatif*, untuk mengevaluasi model klasifikasi. Menghitung *f-measure* untuk mendapatkan penilaian yang lebih seimbang tentang performa model. Untuk menghitung nilai *f-measure* dapat menggunakan rumus perhitungan sebagai berikut:

$$F - measure = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (4)$$

3. HASIL

Hasil didapat dengan melakukan beberapa pengujian menggunakan tiga sampel dan melakukan implementasi dari masing-masing algoritma yang digunakan.

3.1. Decisison Tree

Proses data dengan algoritma *decision tree*, meliputi tiga sampel yang diujikan dengan *cross validation* k=10 menghasilkan *confusion matrik* yang dapat dilihat pada tabel 2.

Tabel 2. *Confusion Matrix Decision Tree*

Sampling	*True Positif*	*True Negative*	*False Positif*	*False Negative*
Linear	120	449	51	148
Shuffled	116	450	50	152
Stratified	124	454	46	144

Tabel 3. Hasil Evaluasi *Decision Tree*

Sampling	*accuracy (%)*	*Precision (%)*	*Recall (%)*	*f-measure (%)*
Linear	74.09	74.95	89.73	81.58
Shuffled	73.71	74.74	90.28	81.56
Stratified	75.26	76.05	90.80	82.68

Penelitian sebelumnya pada sampel berlapis (*stratified*) [8] menghasilkan akurasi 69%, *precision* 67%, *recall* 69% dan f-*measure* 64%. Hasil pengujian pada sample berlapis (*stratified*) menghasilkan akurasi 75.26%, *precision* 76.05%, *recall* 90.80% dan f-measure 82.68%. Hasil dari pengujian dengan *confusion matrix* menghasilkan nilai *accuracy, precision, recall* dan f-*measure* tertinggi yang terdapat pada sampel *stratified*.

3.2. Naïve Bayes

Confusion matrix naïve bayes menggunakan *cross validation* dengan jumlah fold k=10 dengan menggunakan 3 jenis *sampling* menghasilkan data yang terdapat pada tabel 4.

Tabel 4. *Confusion Matrix Naive Bayes*

Sampling	*True Positif*	*True Negative*	*False Positif*	*False Negative*
Linear	156	435	65	112
Shuffled	155	433	67	113
Stratified	154	432	68	114

Tabel 5. Hasil Evaluasi *Naive Bayes*

Sampling	*accuracy (%)*	*Precision (%)*	*Recall (%)*	*f-measure (%)*
Linear	76.95	79.24	86.86	82.79
Shuffled	76.57	79.39	87.02	82.82
Stratified	76.31	79.32	86.40	82.58

Hasil evaluasi dari penelitian sebelumnya menggunakan sampel berlapis (*stratified*) dengan hasil nilai akurasi 78%, *precision* 77%, *recall* 78% dan *f-measure* 77% [8]. Penelitian ini menghasilkan nilai akurasi 76.31%, *precision* 79.32%, *recall* 86.40%, *f-measure* 82.58%. Hasil evaluasi berdasarkan dari *confusion matrix* menghasilkan nilai *accuracy* tertinggi pada *sampling linear*, hasil nilai *precision, recall* dan *f-measure* tertinggi berada pada *sampling shuffled*.

3.3. Random Forest

Proses data dengan algoritma random forest yang meliputi tiga sample yang diujikan dengan menggunakan *cross validation* k=10 menghasilkan *confusion* matrik sebagai berikut :

Tabel 6. *Confusion Matrix Random Forest*

Sampling	*True Positif*	*True Negative*	*False Positif*	*False Negative*
Linear	129	451	49	139
Shuffled	122	454	46	146
Stratified	123	456	46	145

Tabel 7. Hasil Evaluasi *Random Forest*

Sampling	*accuracy (%)*	*Precision (%)*	*Recall (%)*	*f-measure (%)*
Linear	75.53	76.30	90.19	82.51
Shuffled	75.01	75.81	91.11	82.47
Stratified	75.13	76.10	90.80	82.65

Penelitian sebelumnya menggunakan sampel acak (*shuffled*) dengan 520 data dan 17 atribut, menghasilkan nilai akurasi 99.3%, *precision* 99.5%, *recall* 99.1% dan *f-measure* 99% [9]. Penelitian ini menggunakan 768 data dengan 6 atribut, menghasilkan akurasi 75.01%, *precision* 75.81%, *recall* 91.11% dan *f-measure* 82.47% menggunakan sampel acak. Hasil evaluasi pada sampel berlapis *(stratified*) menghasilkan nilai akurasi 75.13%, *precision* 76.10%, *recall* 90.80% dan *f-measure* 82.65%. Penelitian sebelumnya menghasilkan akurasi 79%, *precision* 90%, *recall* 78% dan *f-measure* 83% [10] pada penggunaan data sampel berlapis. Hasil evaluasi menunjukkan *accuracy* dan *precision* tertinggi terdapat pada *sampling linear*, sedangkan nilai *recall* tertinggi terdapat pada sampling acak (*shuffled*) dan nilai *f-measured* tertinggi terdapat pada sampling *stratified*.

3.4. Gradient Boosting

Proses data dilakukan dengan menggunakan tiga sampel, sampel *linear*, acak dan berlapis. Hasil dari proses data menggunakan *cross validation* k=10 dapat dilihat pada tabel 8.

https://doi.org/10.25077/TEKNOSI.v10i2.2024.118-125

Tabel 8. Confusion Matrix Gradient Boosted

Sampling	*True Positif*	*True Negative*	*False Positif*	*False Negative*
Linear	153	439	61	115
Shuffled	158	429	71	110
Stratified	157	437	63	111

Tabel 9. Hasil Evaluasi Gradient Boosting

Sampling	*accuracy (%)*	*Precision (%)*	*Recall (%)*	*f-measure (%)*
Linear	77.09	78.96	87.50	82.91
Shuffled	76.44	80.03	86.10	82.58
Stratified	77.34	79.87	87.40	83.39

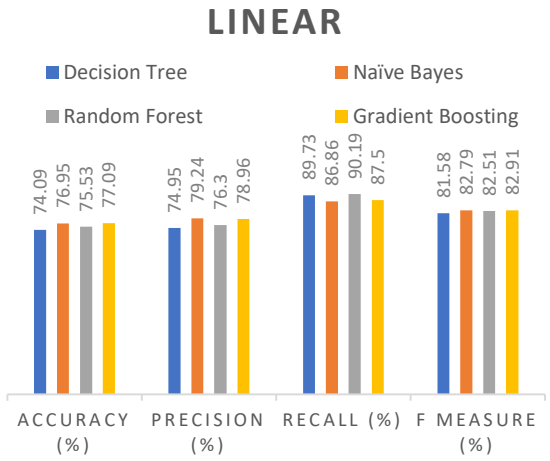
Penelitian terdahulu menggunakan sampel berlapis menghasilkan nilai akurasi 81%, *precision* 67%, *recall* 83% dan *f-measure* 74% [10] . Penelitian ini menghasilkan nilai akurasi 77.34%, *precision* 79.87%, *recall* 87.40% dan *f-measure* 83.39% pada *sample stratified*. Hasil pengujian dari *confusion matrix* menggunakan algoritma *gradient boosting*, dengan menggunakan tiga sampel menghasilkan nilai *accuracy* dan *precision* tertinggi pada sampel berlapis atau *stratified*. Nilai *recall* tertinggi terdapat pada *sampling linear* dan nilai *f-measure* tertinggi terdapat pada *sampling stratified*.

4. PEMBAHASAN

Pengujian berdasarkan dari masing-masing algoritma, menghasilkan *confusion matrix* yang dapat digunakan untuk melakukan pengujian selanjutnya, pengujian tersebut dilakukan dengan menggunakan tiga jenis sampel, sampel linear, sampel acak *(shuffled)* dan sampel berlapis *(stratified)*. Penelitian ini melakukan uji coba pada tiga jenis sampel (linear, acak, dan berlapis) dengan algoritma *machine learning* yang berbeda. Hasil ini menunjukkan bahwa algoritma *machine learning* memiliki performa yang bervariasi tergantung pada metode sampling yang digunakan. *Gradient Boosting* memberikan hasil terbaik pada sampel berlapis (*stratified*), sedangkan *Naïve Bayes* memberikan performa yang lebih tinggi pada sampel acak (*shuffled*).

4.1. Linear Sampling

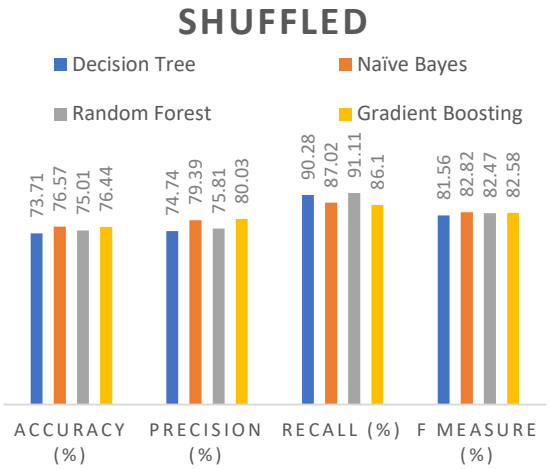
Hasil dari pengujian ke empat algoritma yang digunakan pada sampel *linear* dapat dilihat pada gambar 8. Algoritma dengan nilai tertinggi pada hasil *accuracy*, terdapat pada algoritma *gradient boosting* yang memiliki nilai 77.09%, nilai *precision* tertinggi terdapat pada algoritma *naive bayes* dengan nilai 79.24%, nilai *recall* tertinggi terdapat pada algoritma *random forest* dengan hasil nilai 90.19%, pada hasil perhitungan *f-measure* nilai tertinggi dihasilkan dengan algoritma *gradient boosting*, dengan hasil nilai 82.91%.



Gambar 8. Hasil Pengujian Sampel *Linear*

4.2. Shuffled Sampling

Hasil dari pengujian ke empat algoritma yang digunakan pada sampel acak dapat dilihat pada gambar 9.



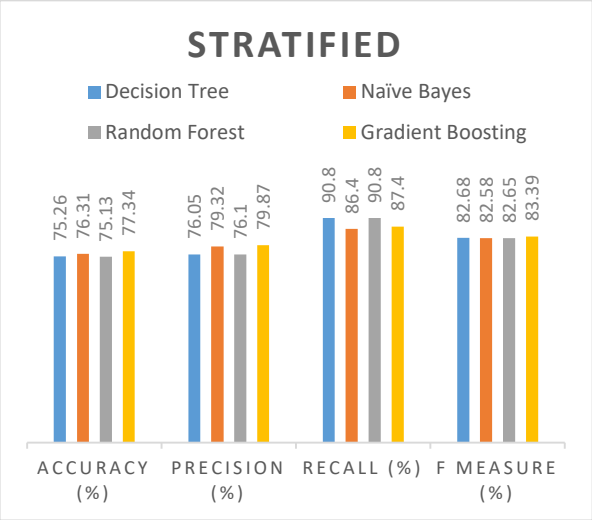
Gambar 9. Hasil Pengujian Sampel Acak

Algoritma dengan nilai tertinggi pada hasil *accuracy*, terdapat pada algoritma *naive bayes* yang memiliki nilai 76.57%, nilai *precision* tertinggi terdapat pada algoritma *gradient boosting* dengan nilai 80.03%, nilai *recall* tertinggi terdapat pada algoritma *random forest* dengan hasil nilai 91.11%, pada hasil perhitungan

f-measure nilai tertinggi dihasilkan dengan algoritma *naive bayes* dengan hasil nilai 82.81%.

4.3. Stratified Sampling

Hasil dari pengujian ke empat algoritma yang digunakan pada sampel berlapis dapat dilihat pada gambar 10.



Gambar 10. Hasil Pengujian Sampel Berlapis

Algoritma dengan nilai tertinggi pada hasil *accuracy*, terdapat pada algoritma *gradient boosting* yang memiliki nilai 77.34%, nilai *precision* tertinggi terdapat pada algoritma *gradient boosting* dengan nilai 79.87%, nilai *recall* tertinggi terdapat pada algoritma *decision tree* dan *random forest* dengan hasil nilai pada *decision tree* sebesar 90.80% dan *random forest* sebesar 90.80%. Hasil perhitungan f-*measure* nilai tertinggi dihasilkan dengan algoritma *gradient boosting* dengan hasil nilai 83.39%.

Penelitian ini memberikan wawasan baru mengenai pentingnya pemilihan metode sampling yang sesuai dengan algoritma tertentu untuk meningkatkan akurasi prediksi, sesuatu yang belum banyak dijelaskan dalam penelitian sebelumnya. Metode *sampling* dapat secara signifikan mempengaruhi performa model, yang dapat diaplikasikan lebih luas di bidang *machine learning* dan prediksi penyakit lainnya.

5. KESIMPULAN

Berdasarkan hasil komparasi dari beberapa algoritma yang digunakan, algoritma klasifikasi *naive bayes* dan *gradient boosting* memiliki nilai yang terbaik dari algoritma lainnya. Algoritma *gradient boosting* memiliki hasil yang tinggi terhadap nilai *accuracy* 77.09% dan *f-measure* 83.39% pada sampel *linear*. *Naive bayes* menghasilkan nilai yang terbaik terhadap pengujian sampel acak, dengan nilai *accuracy* 76.57% dan nilai *f-measure* 82.82%. Hasil pengujian sampel berlapis *(stratified)* yang memiliki nilai pada akurasi tertinggi terdapat pada algoritma *gradient boosting* dengan nilai *accuracy* 77.34% dan *f-measure* 83.39%. Pemilihan model yang cocok untuk penggunaan data linear dan berlapis adalah dengan menggunakan algoritma *gradient boosting* sedangkan untuk pengujian data acak dapat menggunakan algoritma *naive bayes*.

Penelitian ini memiliki beberapa kelemahan yang dapat dijasikan pertimbangan untuk penelitian selanjutnya adalah meskipun berbagai algoritma telah digunakan untuk prediksi penyakit diabetes, tidak ada algoritma tunggal yang selalu memberikan hasil terbaik di semua kondisi. Penelitian ini menunjukkan bahwa *Naïve Bayes* dan *Gradient Boosting* memiliki performa yang bervariasi tergantung pada metode sampling yang digunakan.

Salah satu masalah yang dihadapi dalam penelitian ini adalah bagaimana metode sampling yang berbeda dapat mempengaruhi hasil prediksi dari algoritma yang digunakan. kesimpulannya, penelitian ini menggaris bawahi bahwa kinerja algoritma dapat bergantung pada teknik sampling. *Gradient Boosting* memberikan hasil terbaik dengan sampling berlapis, dan *Naïve Bayes* lebih optimal dengan sampling acak

DAFTAR PUSTAKA

[1] D. Kelurahan, G. Semarang, S. Widiyanti, and D. N. Aini, *Penerapan Pemberian Ekstrak Kayu Manis Terhadap Penurunan Kadar Gula Darah Pada Penderita Diabetes Melitus*.

[2] N. Nina, H. Purnama, H. Z. N. Adzidzah, M. Solihat, M. Septriani, and S. Sulistiani, "Determinan Risiko dan Pencegahan terhadap Kejadian Penyakit Diabetes Melitus Tipe 2 pada Usia Produktif di Wilayah DKI Jakarta," *Journal of Public Health Education*, vol. 2, no. 4, pp. 377–385, Jul. 2023, doi: 10.53801/jphe.v2i4.148.

[3] Q. Aziz Gunawan, "Penyuluhan dan Cek Kadar Gula Darah Sewaktu Sebagai Upaya Deteksi Dini Diabetes Mellitus Tipe 2 di Kelurahan Sudirejo II".

[4] S. Rini, O. Ayu Dhea Manto, A. Irawan, P. Studi Sarjana Keperawatan, F. Kesehatan, and U. Sari Mulia, "Journal of Nursing Invention Hubungan Pola Hidup Dengan Kadar Gula Darah Pasien Dengan Diabetes Mellitus Tipe 2."

[5] Y. Nora Marlim, L. Suryati, and N. Agustina, "Deteksi Dini Penyakit Diabetes Menggunakan Machine Learning dengan Algoritma Logistic Regression," 2022.

[6] C.-Y. Chou, D.-Y. Hsu, and C.-H. Chou, "Predicting the onset of diabetes with machine learning methods," *J Pers Med*, vol. 13, no. 3, p. 406, 2023.

[7] K. R. Tan *et al.*, "Evaluation of machine learning methods developed for prediction of diabetes complications: a systematic review," *J Diabetes Sci Technol*, vol. 17, no. 2, pp. 474–489, 2023.

[8] T. Syamsudin, T. Handhayani,) Muhammad, and I. Syaifudin, "Jurnal Ilmu Komputer dan Sistem Informasi Perbandingan Klasifikasi Penyakit Diabetes Menggunakan Metode Machine Learning." [Online]. Available: https://www.kaggle.com/datasets/nanditapore/healthcar

[9] J. Teknika and A. Ria Supriyatna, "Teknika 17 (1): 163-172 Prediksi Penyakit Diabetes Menggunakan Algoritma Random Forest," *IJCCS*, vol. x, No.x, pp. 1–5.

https://doi.org/10.25077/TEKNOSI.v10i2.2024.118-125

[10] S. P. Nainggolan and A. Sinaga, "Comparative Analysis Of Accuracy Of Random Forest And Gradient Boosting Classifier Algorithm For Diabetes Classification," *Sebatik*, vol. 27, no. 1, pp. 97–102, Jun. 2023, doi: 10.46984/sebatik.v27i1.2157.

[11] B. N. Azmi, A. Hermawan, and D. Avianto, "Analisis Pengaruh komposisi data training dan data testing Pada penggunaan PCA Dan Algoritma decision tree untuk KLASIFIKASI Penderita Penyakit liver," *JTIM: Jurnal Teknologi Informasi Dan Multimedia*, vol. 4, no. 4, pp. 281–290, 2023.

[12] T. Leinonen, D. Wong, A. Wahab, R. Nadarajah, M. Kaisti, and A. Airola, "Empirical investigation of multi-source cross-validation in clinical machine learning," Mar. 2024, [Online]. Available: http://arxiv.org/abs/2403.15012

[13] M. Muhammad, J. Samodro, M. Kunta Biddinika, A. Fadlil, A. Dahlan, and Y. J. Ringroad Selatan, "Klasifikasi Penyakit Diabetes dengan Algoritma Decision Tree dan Naïve Bayes," vol. 6, no. 2.

[14] S. Kolo, "Impact Of Data Preprocessing And Balancing On Diabetes Prediction Using The Decision Tree Technique," *International Journal of Numerical Methods and Applications*, vol. 23, no. 2, pp. 157–180, Jun. 2023, doi: 10.17654/0975045223008.

[15] D. Saputra, W. Irmayani, D. Purwaningtias, and J. Sidauruk, "A Comparative Analysis of C4.5 Classification Algorithm, Naïve Bayes and Support Vector Machine Based on Particle Swarm Optimization (PSO) for Heart Disease Prediction," *International Journal of Advances in Data and Information Systems*, vol. 2, no. 2, Nov. 2021, doi: 10.25008/ijadis.v2i2.1221.

[16] J. Teknika and A. Ria Supriyatna, "Teknika 17 (1): 163-172 Prediksi Penyakit Diabetes Menggunakan Algoritma Random Forest," *IJCCS*, vol. x, No.x, pp. 1–5.

[17] M. Ali, M. N. Haider, S. A. Lashari, W. Sharif, A. Khan, and D. A. Ramli, "Stacking Classifier with Random Forest functioning as a Meta Classifier for Diabetes Diseases Classification," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 3453–3462. doi: 10.1016/j.procs.2022.09.404.

[18] A. V. Konstantinov, L. V. Utkin, S. R. Kirpichenko, B. V. Kozlov, and A. Y. Ageev, "Random Forests with Attentive Nodes," in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 454–463. doi: 10.1016/j.procs.2022.11.029.

[19] Gde Agung Brahmana Suryanegara, Adiwijaya, and Mahendra Dwifebri Purbolaksono, "Peningkatan Hasil Klasifikasi pada Algoritma Random Forest untuk Deteksi Pasien Penderita Diabetes Menggunakan Metode Normalisasi," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 1, pp. 114–122, Feb. 2021, doi: 10.29207/resti.v5i1.2880.

[20] M. Alnaggar, M. Handosa, T. Medhat, and M. Z. Rashad, "Thyroid Disease Multi-class Classification based on Optimized Gradient Boosting Model," *Egyptian Journal of Artificial Intelligence*, vol. 2, no. 1, pp. 1–14, Apr. 2023, doi: 10.21608/ejai.2023.205554.1008.

[21] M. R. Ansyari, M. I. Mazdadi, F. Indriani, D. Kartini, and T. H. Saragih, "Implementation of Random Forest and Extreme Gradient Boosting in the Classification of Heart Disease using Particle Swarm Optimization Feature Selection," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 5, no. 4, pp. 250–260, Sep. 2023, doi: 10.35882/jeeemi.v5i4.322.

[22] M. Hasnain, M. F. Pasha, I. Ghani, M. Imran, M. Y. Alzahrani, and R. Budiarto, "Evaluating Trust Prediction and Confusion Matrix Measures for Web Services Ranking," *IEEE Access*, vol. 8, pp. 90847–90861, 2020, doi: 10.1109/ACCESS.2020.2994222.

[23] E. Ismanto and M. Novalia, "dan Gradient Boosting untuk Klasifikasi Komoditas Performance Comparison Between C4.5 Algorithm, Random Forests, and Gradient Boosting for Commodity Classification."

[24] M. A. Naji, S. El Filali, K. Aarika, E. H. Benlahmar, R. A. Abdelouhahid, and O. Debauche, "Machine Learning Algorithms For Breast Cancer Prediction And Diagnosis," *Procedia Comput Sci*, vol. 191, pp. 487–492, 2021, doi: 10.1016/j.procs.2021.07.062.

NOMENKLATUR

TP	arti dari True Positif
TN	arti dari True Negatif
FP	arti dari variabel False Positif
FN	arti dari variabel False Negatif

BIODATA PENULIS

Nova Christina Sari
Merupakan peneliti dan dosen pada program studi Teknologi Informasi Universitas Muhammadiyah Semarang. Menyelesaikan pendidikan S2 di Magister Sistem Informasi Universitas Diponegoro Semarang. Fokus melakukan penelitian dibidang *machine learning* kesehatan, *data mining*, sistem informasi kesehatan dan *cyber security* .

Triana Linda Larasati
Seorang dokter umum di klinik simpang jawo kota jambi, yang beralamat di jalan tarumanegara no 01 RT 12 kel tanjung pinang kec jambi timur. Menyelesaikan pendidikan dokter di Fakultas Kedokteran Universitas Jambi