Pemanfaatan Decision Tree pada Algoritma Random Forest untuk Klasifikasi Kanker Payudara

Adhimas Aryo Bimo - 13523052
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13523052@std.stei.itb.ac.id adhimas.bimo@gmail.com

***Abstrak*—Kanker Payudara menjadi penyakit kanker yang paling sering terjadi saat ini. Pada tahun 2022 sendiri penderita kanker payudara menyentuh angka 670.000 jiwa. Walaupun sangat umum terjadi, dengan melakukan diagnosis dini peluang kesembuhan dapat meningkat. Untuk membantu melakukan diagnosis pada kanker payudara dapat dimanfaatkan dengan menggunakan pendekatan *machine learning*. Makalah ini bertujuan untuk mengetahui pemanfaatan Decision Tree dalam Random Forest dan meneliti bagaimana Random Forest dapat menghasilkan keluaran yang lebih baik dari Decision Tree. Hasil penelitian menunjukkan bahwa teknik penggabungan Decision Tree pada model Random Forest berhasil menurunkan tingkat kesalahan hingga 50% pada prediksi kelas tumor ganas. Selain itu, dapat diketahui bahwa fitur "perimeter_worst" dan "area_worst" memiliki pengaruh yang signifikan pada proses klasifikasi.**

***Kata Kunci*—Decision Tree, Kanker Payudara, Random Forest**

I. PENDAHULUAN

Menurut Organisasi Kesehatan Dunia (WHO), kanker payudara merupakan kanker paling umum pada wanita yang terjadi di 157 dari 185 negara pada tahun 2022. Kanker ini terjadi hampir pada setiap negara di dunia. Pada tahun 2022, secara global kanker payudara menyebabkan hingga 670.000 kematian. Namun, dengan melakukan diagnosis dini, peluang untuk sembuh dari kanker payudara dapat meningkat. Oleh karena itu, diperlukan pendekatan teknologi untuk membantu dalam proses diagnosis kanker payudara berdasarkan data medis yang tersedia.

Gambar 1.1 Simbol Kanker Payudara
Sumber : https://www.kaggle.com/datasets/yasserh/breast-cancer-dataset

Untuk melakukan proses diagnosis tersebut, dapat digunakan klasifikasi kanker payudara menggunakan pendekatan *machine learning*. Salah satu algoritma *machine learning* yang paling sering digunakan adalah Decision Tree (Pohon Keputusan). Algoritma tersebut merupakan algoritma yang sederhana dan bekerja dengan membagi data berdasarkan aturan-aturan logis. Decision Tree memberikan interpretasi yang mudah dipahami karena cara kerjanya yang cukup intuitif dan menyerupai logika manusia dalam mengambil keputusan. Selain itu, struktur pohon Decision Tree juga mudah untuk divisualisasikan. Namun, Decision Tree sering kali menghadapi masalah seperti *overfitting*, terutama ketika diterapkan pada dataset yang kompleks.

Untuk mengatasi permasalahan tersebut, algoritma Random Forest diperkenalkan. Random Foreest merupakan algoritma *ensemble learning* yang menggabungkan banyak Decision Tree untuk meningkatkan akurasi prediksi. Selain itu, Random Forest memiliki teknik *bootstrap sampling* dan *random feature selection* untuk mengatasi risiko *overfitting*.

Makalah ini bertujuan untuk menganalisis dan mengevaluasi penggunaan algoritma Decision Tree dalam Random Forest untuk klasifikasi kanker payudara. Dalam makalah ini akan dilakukan percobaan menggunakan Wisconsin Breast Cancer Dataset yang berisi data medis seperti ukuran, tekstur, dan perimeter tumor untuk menentukan apakah suatu tumor bersifat jinak (*benign / B*) atau ganas (*malignant / M)*.

Hasil dari makalah ini diharapkan dapat memahami cara kerja Decision Tree pada Algoritma Random Forest dan melakukan analisis performa perbandingan kedua model *machine learning* menggunakan metrik *accuracy* score dan *balanced accuracy score*.

Dengan demikian, makalah ini tidak hanya berkontribusi pada pemahaman teknis algoritma *machine learning*, tetapi juga menunjukkan bagaimana teknologi dapat digunakan untuk membantu menyelesaikan masalah medis yang kompleks.

II. LANDASAN TEORI

A. Kanker Payudara

Kanker payudara (*carcinoma mammae*) merupakan jenis kanker yang berkembang dari jaringan payudara, baik dari epitel duktus maupun lobulusnya. Penyakit ini terjadi ketika sel-sel mengalami kehilangan kendali atas mekanisme normalnya, sehingga tumbuh dengan cepat, tidak teratur, dan tanpa kendali.

Kanker payudara merupakan kanker yang paling sering terdiagnosis pada wanita, mencakup lebih dari 10% dari seluruh kasus baru kanker setiap tahunnya. Penyakit ini juga menjadi penyebab kematian akibat kanker nomor dua terbanyak di kalangan wanita di seluruh dunia. Kanker payudara sering kali berkembang secara perlahan tanpa gejala yang jelas, dan sebagian besar kasus baru terdeteksi melalui pemeriksaan rutin.

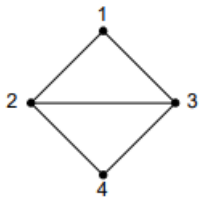
Walaupun kanker payudara sering terjadi, melakukan deteksi dini kanker payudara dapat meningkatkan peluang kesembuhan hingga 80-90%.

B. Graf

Graf merupakan struktur yang terdiri dari simpul (*node*) dan sisi (*edges*) yang digunakan untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut. Secara matematis, graf dapat dinotasikan sebagai *G = (V,E)* yang dalam hal ini:

V = himpunan tidak kosong dari simpul-simpul (*Vertices*)
= { v1 , v2 , ... , vn }

E = himpunan sisi (*edges*) yang menghubungkan sepasang simpul
= {e1 , e2 , ... , en }

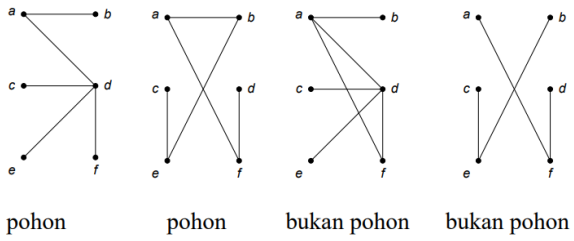


Gambar 2.1 Graf Sederhana
Sumber
https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/20-Graf-Bagian1-2024.pdf

C. Pohon

Pohon adalah graf berhubung tak-berarah (*undirected graph*) yang tidak memiliki sirkuit maupun sisi ganda. Pohon dalam matematika diskrit memiliki properti. Misalkan *G = (V, E)* adalah graf tak-berarah sederhana dan jumlah simpulnya n. Maka, semua pernyataan di bawah ini adalah ekivalen:

1. G adalah pohon.
2. Setiap pasang simpul di dalam G terhubung dengan lintasan tunggal.
3. G terhubung dan memiliki m = n – 1 buah sisi.
4. G tidak mengandung sirkuit dan memiliki m = n – 1 buah sisi.
5. G tidak mengandung sirkuit dan penambahan satu sisi pada graf akan membuat hanya satu sirkuit.
6. G terhubung dan semua sisinya adalah jembatan.



Gambar 2.2 Pohon
Sumber :
https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/23-Pohon-Bag1-2024.pdf

D. Decision Tree

Decision Tree adalah algoritma *machine learning* berbasis pohon (*tree-based algorithm*) yang dapat digunakan dalam klasifikasi atau regresi. Struktur Decision Tree mirip dengan struktur Pohon Berakar, yakni memiliki akar, simpul, cabang dan daun.

Decision Tree akan membagi dataset menjadi beberapa subset berdasarkan fitur yang paling informatif menggunakan metrik tertentu, seperti Gini Index, Information Gain, atau Entropi.

Entropi adalah ukuran ketidakpastian atau ketidakteraturan dalam sebuah dataset. Konsep ini berasal dari teori informasi yang dikembangkan oleh Claude Shannon. Secara matematis entropi dapat dituliskan sebagai berikut.

$$H(S) = -\sum_{i=1}^{k} p_i \, log_2(p_i)$$

Keterangan:

- H(S): Entropy dataset S,
- k: Jumlah kelas dalam dataset,
- p_i: Proporsi (probabilitas) sampel dalam kelas ke-i.

Semakin tinggi nilai entropi, maka dataset memiliki distribusi kelas yang merata. Misal, sebuah dataset memiliki 10 sampel di mana 4 kelas adalah A dan 6 kelas adalah B, maka.

- Probabilitas kelas A $p_A = 4/10 = 0.4$
- Probabilitas kelas B $p_B = 6/10 = 0.6$

Maka didapatkan entropi sebesar

$$H(S) = -(0.4 \, log_2(0.4) + 0.6 \, log_2(0.6)) = 0.97$$

Entropi mendekati 1, menunjukkan dataset memiliki ketidakpastian yang tinggi.

Information Gain dalam Decision Tree digunakan untuk mengukur pengurangan Entropi setelah data dibagi berdasarkan suatu fitur. Secara matematis dirumuskan sebagai berikut:

$$IG(S,A) = H(S) - \sum_{v \in A} \frac{|S_v|}{|S|} H(S_v)$$

Keterangan:

- H(S) : Entropi dataset sebelum dibagi
- A : Fitur yang digunakan untuk membagi dataset.
- S_v: Subset dari S berdasarkan nilai v pada fitur A
- $|S_v|$: Jumlah sampel dalam subset S_v
- $|S|$: Jumlah total sampel dalam dataset

Semakin tinggi Information Gain, semakin baik fitur tersebut dalam memisahkan data. Decision Tree akan memilih fitur dengan Information Gain tertinggi di setiap simpul.

Misal menggunakan data sebelumnya, jika H(S) = 0.97, maka setelah membagi dataset berdasarkan fitur XX, $H(S_v)$ rata-rata turn menjadi 0.6, maka:

$$IG(S,A) = 0.97 - 0.6 = 0.37$$

Fitur XX memilik Information Gain sebesar 0.37.

Gini Index merupakan salah satu metode untuk mengukur homogenitas dataset. Semakin kecil nilai Gini Index, semakin baik karena subset tersebut menjadi lebih homogen. Gini Index dapat di tuliskan secara matematis sebagai berikut:

$$Gini(S) = 1 - \sum_{i=1}^{k} p^2 i$$

Keterangan :

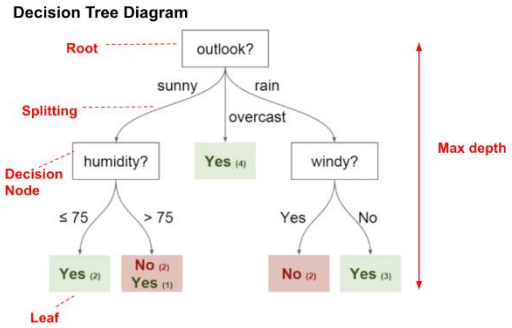
- Gini(S) : Gini Index dataset S.
- k : Jumlah kelas dalam dataset.
- p_i : Proporsi (probabilitas) sampel dalam kelas ke-i

Dengan menggunakan probabilitas sampel kelas A dan kelas B dapat dihitung nilai Gini Index:

$$Gini(S) = 1 - (0.6^2 + 0.4^2) = 0.48$$

Gini Index menunjukkan bahwa dataset cukup heterogen.

Dalam Decision Tree, Entropi, Information Gain, dan Gini Index digunakan untuk memilih fitur terbaik dalam membagi dataset di setiap simpul (node).

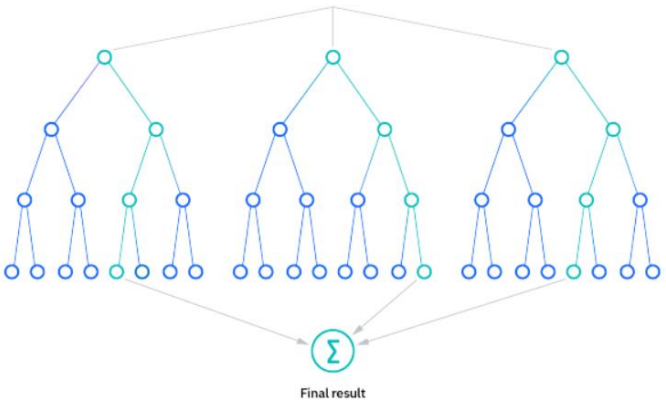


Gambar 2.3 Contoh Decision Tree
Sumber :
https://www.trivusi.web.id/2022/06/algoritma-decision-tree.html

E. Random Forest

Random Forest adalah algoritma *machine learning* berbasis *ensemble learning* yang menggunakan kombinasi dari banyak Decison Tree. Algoritma Random Forest ditemukan oleh Leo Breiman dan Adele Cutler.

Kombinasi dari gabungan Decision Tree membuat algoritma Random Forest menjadi lebih tinggi akurasi, stabil, dan lebih tahan terhadap *overfitting*. Random Forest



Gambar 2.4 Random Forest
Sumber :
https://www.ibm.com/id-id/topics/random-forest

Dalam membuat berbagai kombinasi Decision Tree, Random Forest akan melakukan Bootstrap Sampling (Bagging). Bagging memiliki metode seperti berikut, dataset asli akan dibagi menjadi beberapa subset. Subset tersebut akan dilatih ke tiap Decision Tree.

Selain Bagging, Random Forest juga memiliki metode Random Feature Selection. Metode ini dilakukan agar setiap subset yg dilatih di tiap Decision Tree memiliki fitur yang berbeda. Hal ini dilakukan untuk menciptakan variasi antara Decision Tree.

Hasil dari semua Decision Tree akan disimpulkan tergantung jenis kelas dari sebuah dataset. Jika kelas yang diprediksi merupakan klasifikasi, maka hasil ditentukan berdasarkan voting, sedangkan untuk regresi akan ditentukan dari agregat semua hasil Decision Tree.

F. Metrik Evaluasi

Dalam makalah ini akan digunakan 2 metrik evaluasi yakni, akurasi (*accuracy*) dan akurasi seimbang (*balanced accuracy*). Akurasi adalah metrik evaluasi yang mengukur proporsi prediksi benar terhadap total jumlah prediksi, Akurasi dapat ditulis secara matematis dengan persamaan berikut:

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

Keterangan :

- TP: True Positive (jumlah kelas positif yang diprediksi benar),
- TN: True Negative (jumlah kelas negatif yang diprediksi benar),
- FP: False Positive (jumlah kelas negatif yang diprediksi sebagai positif),
- FN: False Negative (jumlah kelas positif yang diprediksi sebagai negatif).

Untuk mendapatkan analisis yang lebih baik, dapat dipertimbangkan hasil prediksi tiap kelas. Oleh karena itu, digunakan *balanced accuracy*.

$$Balanced\ Accuracy = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right)$$

Metrik ini menghitung rata-rata sensitivitas (*recall*) dari setiap kelas, sehingga memberikan gambaran kinerja model yang lebih seimbang tanpa bias terhadap kelas mayoritas. Dengan *balanced accuracy*, performa model pada kelas minoritas dapat dianalisis dengan lebih baik, mendukung pengambilan keputusan yang lebih informatif.

III. METODE

A. Persiapan Eksperimen dan Modul

Percobaan ini dilakukan menggunakan bahasa pemrograman Python versi 3.12 Beberapa pustaka (modul) yang digunakan dalam eksperimen adalah sebagai berikut.

- Pandas : Digunakan untuk analisis data

- Matplotlib : Digunakan untuk visualisasi data
- Scikit-Learn : Digunakan untuk pelatihan model

B. Persiapan Dataset

Dalam percobaan ini, akan digunakan Wisconsin Breast Cancer Dataset yang dapat diakses di kaggle atau UCI Machine Learing Repository. Berikut merupakan ringkasan singkat deskripsi dataset. Kode dapat diakses pada link Github.

Tabel 3.1 Deskripsi Dataset

Nama	Wisconsin Breast Cancer Dataset.
Sumber Dataset	Kaggle
Jumlah Sample	569 sample
Jumlah Fitur	31 kolom (termasuk id)
Target	2 kelas (Malignant / M / Tumor Ganas dan Benign / B / Tumor jinak) pada kolom 'diagnosis'

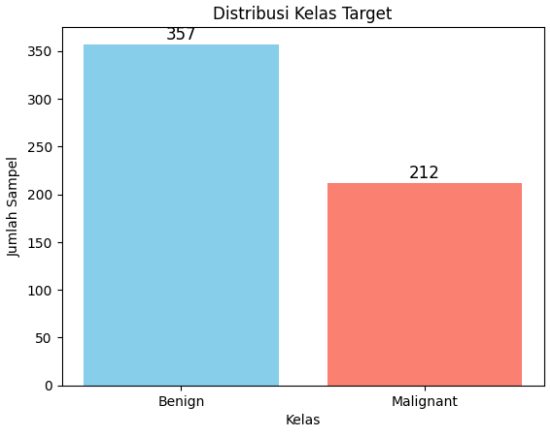
Selanjutnya akan dilakukan pembersihan fitur yang tidak berhubungan dengan target. Fitur yang akan dibersihkan dari dataset ini adalah "id". Oleh karena itu, kolom "id" akan di *drop*.

```
data = pd.read_csv('breast-cancer.csv')
data = data.drop(columns=['id'])
data.head()
```
✓ 0.0s

	diagnosis	radius_mean	texture_mean	perimeter_mean	area_mean	smoothness_mean
0	M	17.99	10.38	122.80	1001.0	0.11840
1	M	20.57	17.77	132.90	1326.0	0.08474
2	M	19.69	21.25	130.00	1203.0	0.10960
3	M	11.42	20.38	77.58	386.1	0.14250
4	M	20.29	14.34	135.10	1297.0	0.10030

Gambar 3.1 Tampilan Sebagian Dataset
Sumber : Olahan Penulis

Untuk dapat menganalisis komposisi data lebih baik, akan ditampilkan proporsi kelas target pada gambar berikut.



Gambar 3.2 Distribusi Kelas Target
Sumber : Olahan Penulis

Pada diagram tersebut terlihat bahwa distribusi kelas tidak sama rata dan cenderung mengarah pada target Benign. Walaupun begitu, dapat diukur Entropi untuk mengetahui ketidakpastian sebagai ketidakpastian awal dalam dataset.

- $p_{Benign} = \frac{357}{569} = 0.627$
- $p_{Malignant} = \frac{212}{569} = 0.373$

Dengan menggunakan persamaan Entropi, dapat dilakukan subtitusi dan didapatkan hasil sebagai berikut.

$$H(S) = -(p_{Benign} log_2(p_{Benign}) + p_{Malignant} log_2(p_{Malignant}))$$
$$H(S) = -(0.672 log_2(0.672) + 0.373 log_2(0.373)$$
$$H(S) = 0.952$$

Dari hasil entropi tersebut, terlihat bahwa tingkat ketidakpastian dalam dataset masih cenderung tinggi (mendekati 1). Walaupun distribusi dataset tidak seimbang, nilai entropi mendekati 1 menunjukkan bahwa proporsi kelas Malignant (M) masih signifikan. Namun masih perlu diperhatikan mengenai proporsi distribusi yang tidak seimbang pada kedua kelas.

Selanjutnya, untuk menyiapkan data evaluasi dilakukan pembagian dataset menjadi 70:30 dengan stratifikasi pada target (distribusi kelas pada 70% dan 30% dataset sama) dengan 70% menjadi data latih dan 30% menjadi data validasi.

```
TARGET = 'diagnosis'
X = data.drop(columns=[TARGET])  # Semua fitur kecuali target
y = data[TARGET]  # Target

# Pembagian dataset menjadi 70:30 dengan stratifikasi pada target
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, stratify=y, random_state=42
)
```
✓ 0.0s

Gambar 3.3 Pembagian Dataset
Sumber : Olahan Penulis

C. Pelatihan Model Decision Tree

Pelatihan dan evaluasi diperlukan agar kita dapat melatih model Decision Tree menggunakan dataset yang tersedia. Dengan melatih model menggunakan 70% dari dataset, dapat ditentukan performa model dengan melakukan evaluasi menggunakan data validasi 30% dari dataset asli. Berikut merupakan kode untuk melakukan pelatihan pada model Decision Tree.

```
# Train Decision Tree
tree_clf = DecisionTreeClassifier(criterion='entropy',
                                  max_depth=3,
                                  random_state=42)
tree_clf.fit(X_train, y_train)

# Prediksi
y_pred = tree_clf.predict(X_test)
```

Gambar 3.4 Kode Latih dan Prediksi pada Model Decision Tree
Sumber : Olahan Penulis

Model Decision Tree yang dimiliki oleh pustaka Scikit-Learn memiliki parameter yang digunakan untuk menentukan kualitas pembagian data. Pembagian tersebut didasari berdasarkan

Entropi dan Gini (lihat II.D). Dalam metode ini, akan digunakan metode Entropi karena kelas yang ada pada dataset memiliki distribusi yang tidak seimbang.

Saat memilih fitur untuk membagi dataset, Decision Tree menggunakan Information Gain yang didasarkan pada pengurangan Entropi. Ketika dataset memiliki distribusi kelas yang tidak seimbang, model akan memberikan bobot yang lebih besar pada distrbusi data yang lebih tidak pasti sehingga cenderung memilih fitur yang memecah dataset menjadi subset yang lebih homogen.

Model Decision Tree juga dapat diatur kedalamannya (max_depth) (lihat gambar 2.3) .Semakin dalam pohon yang dibuat maka semakin kompleks model dalam melakukan pembagian data. Kedalaman pohon yang berlebih dapat mengakibatkan pohon menjadi terlalu *overfit* pada data. Untuk percobaan ini, parameter max_depth akan dibatasi kedalamannya hanya mencapai 3 level saja.

Parameter random_state merupakan parameter yang digunakan sebagai *seed* untuk angka acak yang digunakan oleh model dalam melakukan pembagian dataset. Hal ini diperlukan agar model dapat mereproduksi ulang hasil eksperimen sehingga tiap kali kode dijalankan hasilnya tetap sama.

D. Evaluasi dan Analisis Model Decision Tree

Model Decision Tree akan dievaluasi menggunakan 2 metrik, yakni *accuray score* dan *balanced accuracy score*. Hasil evaluasi tersebut dapat dilihat pada gambar berikut.

```python
# Evaluasi Akurasi dan Metrik Lainnya
accuracy = accuracy_score(y_test, y_pred)
balanced_accuracy = balanced_accuracy_score(y_test, y_pred)

# Buat DataFrame untuk Tabel Hasil Evaluasi
results = {
    "Metric": ["Accuracy", "Balanced Accuracy"],
    "Score": [accuracy, balanced_accuracy]
}
results_df = pd.DataFrame(results)

# Tampilkan Hasil Evaluasi
print("\nHasil Evaluasi Model:")
results_df
```

Gambar 3.5 Kode Evaluasi Decision Tree
Sumber : Olahan Penulis

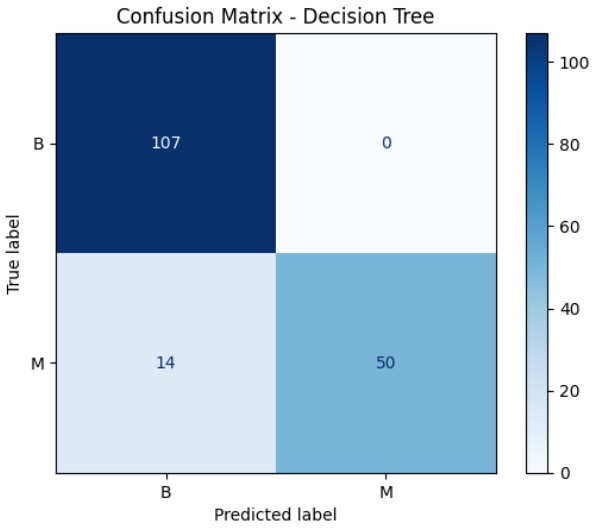
IV. HASIL & PEMBAHASAN

Hasil Evaluasi Model Decision Tree:

	Metric	Score
0	Accuracy	0.918129
1	Balanced Accuracy	0.890625

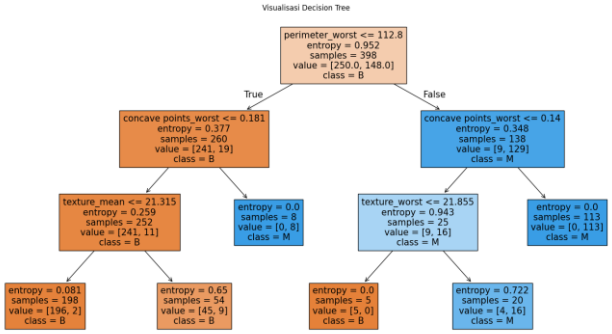
Gambar 4.1 Hasil Evaluasi Decision Tree
Sumber : Olahan Penulis

Terlihat skor yang cukup menakjubkan untuk model Decision Tree. Model tersebut memberikan hasil yang cukup baik hingga bisa mencapai akurasi lebih dari 90% dan skor akurasi seimbang hingga 89%. Agar dapat lebih jelas melihat apa yang diprediksi oleh model akan ditampilkan *confusion matrix* pada gambar berikut.



Gambar 4.2 Confusion Matrix Decision Tree
Sumber : Olahan Penulis

Pada gambar diatas terlihat, ternyata kesalahan dalam memprediksi tumor ganas (Malignant) pada model. Walaupun dapat mencapai akurasi hingga lebih dari 90%, akurasi tersebut sempurna untuk mengklasifikasi tumor jinak saja dan tidak berlaku pada tumor ganas. Dengan hal ini, model masih dianggap belum baik dalam memprediksi tumor ganas pada pasien. Untuk melihat lebih lanjut bagaimana cara kerja Decision Tree, bisa dilihat bagaimana cara model melakukan *splitting* dataset pada gamber berikut.



Gambar 4.3 Visualisasi Decision Tree
Sumber : Olahan Penulis

Pada gambar diatas terdapat beberapa hal yang dapat dijelaskan mengenai proses pembagian data dan informasi yang terkandung di setiap simpul. Akan diberikan penjelasan mengenai gambar diatas pada paragraf berikut.

Setiap simpul pada pohon keputusan memiliki “Fitur dan Threshold”. Kedua komponen ini yang akan membagi dataset menjadi beberapa subset tertentu. Pada root, fitur yang dijadikan nilai threshold adalah “perimeter_worst” dengan threshold <= 112.8. Jika benar, maka data yg benar akan diproses ke cabang kiri dan jika salah akan diproses ke cabang kanan.

Nilai Entropi menunjukkan ketidakpastian pada simpul tersebut. Pada root, memiliki nilai entropi yang sama seperti yang telah dihitung pada persiapan dataset, yakni 0.952 dan seiring lebih dalam, mode mencoba untuk menurunkan entropi pada simpul tersebut.

Pada gambar terlihat ada simpul yang memiliki entropy cukup besar tetapi tidak mengalami pemecahan data lagi . Hal ini yang menjadi kemungkinan mengapa model bisa salah melakukan prediksi. Terutama pada simpul yang memiliki entropi sebesar 0.722 dan 0.65 pada gambar.

Pada gambar juga dapat dilihat bahwa dataset melakukan pemecahan berdasarkan fitur tertentu. Dari sekian banyak fitur yang dimiliki oleh data, dipilih fitur yang paling berpenaruh pada hasil prediksi model. Dalam konteks ini, fitur yang dipilih adalah "perimeter_worst", "concave points_worst", "texture_mean", dan "texture_worst".

D. Pelatihan Model Random Forest

Pada percobaan ini, dataset yang digunakan akan sama seperti pada percobaan sebelumnya. Perbedaannya hanya mengganti modelnya saja. Berikut model yang akan digunakan dalam percobaan.

```
# Train Random Forest dengan kriteria Entropy
forest_clf = RandomForestClassifier(n_estimators=1000,
                                    criterion='entropy',
                                    random_state=42,
                                    max_depth=3)
forest_clf.fit(X_train, y_train)

# Prediksi
y_pred = forest_clf.predict(X_test)
```

Gambar 4.4 Kode Latih dan Prediksi pada Model Random Forest
Sumber : Olahan Penulis

Parameter yang digunakan pada Random Forest mirip dengan Decision Tree, tetapi ada perbedaan sedikit. Pada model Random Forest terdapat parameter "n_estimators" yang digunakan untuk menentukkan seberapa banyak Decision Tree yang akan digunakan untuk pelatihan model. Banyak Decision Tree yang digunakan adalah 1000 pohon pada percobaan kali ini.

D. Evaluasi dan Analisis Model Random Forest

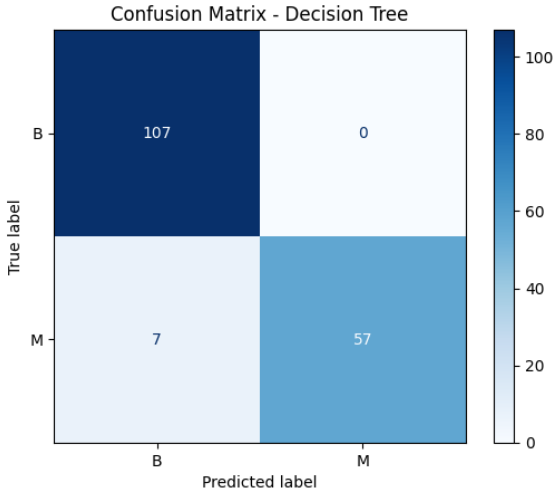
Metode evaluasi sama seperti percobaan sebelumnya. Berikut merupakan hasil evaluasi dari model Random Forest.

Hasil Evaluasi Model Random Forest:

	Metric	Score
0	Accuracy	0.959064
1	Balanced Accuracy	0.945312

Gambar 4.5 Evaluasi Model Random Forest
Sumber : Olahan Penulis

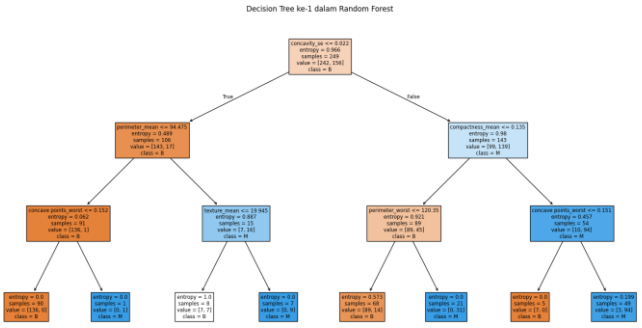
Terlihat skor akurasi dan akurasi seimbang pada model ini jauh lebih meningkat dibandingkan dengan model Decision Tree. Hal ini membuktikan bahwa model Decision Tree yang digabung dapat menghasilkan performa model yang lebih baik dalam melihat pola dalam suatu data. Untuk melihat lebih rinci hasil prediksi model dapat dilihat *confusion matrix* berikut.



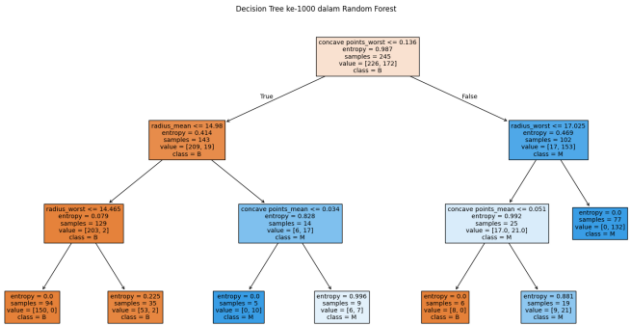
Gambar 4.6 Confusion Matrix Random Forest
Sumber : Olahan Penulis

Gambar di atas menunjukkan bahwa model Random Forest salah memprediksi kelas Malignant. Namun, jumlah kesalahan tersebut menurun hingga 50% ketika menggunakan model Random Forest. Hal ini cukup mengejutkan, kenaikan akurasi sebesar 4% mampu menurunkan kesalahan hingga 50%.

Proses yang terjadi pada Random Forest, mirip dengan proses yang terjadi pada Decision Tree. Namun, Pohon yang digunakan lebih banyak dan tiap pohon akan memiliki subset dan fitur yang berbeda. Namun, tetap dapat dilihat salah satu pohon pada proses latih model Random Forest.



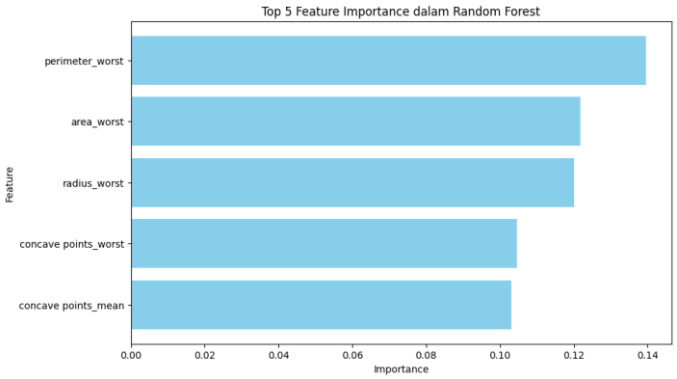
Gambar 4.7 Visualisasi Decision Tree ke-1 pada model Random Forest
Sumber : Olahan Penulis



Gambar 4.8 Visualisasi Decision Tree ke-1000 pada model Random Forest
Sumber : Olahan Penulis

Tentunya tidak akan cukup jika dibahas proses pemilihan pada tiap Decision Tree di model Random Forest. Namun, yang ingin digaris bawahi dari kedua gambar di atas adalah tiap tree memiliki cara pemecahan data yang berbeda, hal ini karena proses *Bagging* dan *Random Feature Selection* yang terjadi pada setiap Decision Tree di model Random Forest.

Selanjutnya, sulit untuk menentukkan fitur yang paling penting jika melihat pohonnya secara satu-satu. Namun, kita dapat mengetahui fitur terpenting dalam proses prediksi menggunakan *feature importances* yang ada pada model Random Forest.



Gambar 4.9 Top 5 Fitur Paling Berpengaruh pada Model Random Forest

Dari hasil latih model Random Forest, terlihat bahwa fitur yang paling berpengaruh pada model secara terurut dari yg paling penting adalah “perimeter_worst”, “area_worst”, “radius_worst”, concave point_worst”, “concave points_mean”.

Terdapat beberapa kesamaan fitur penting yang digunakan pada Random Forest dan Decision Tree pada percobaan sebelumnya. Hal ini wajar karena keduanya memiliki cara kerja yang mirip dan kemungkinan ada Decision Tree pada Random Forest yang mirip dengan Decision Tree pada percobaan sebelumnya.

A. Ringkasan Hasil

Percobaan pada makalah ini menggunakan model *machine learning* Decision Tree dan Random Forest menggunakan Wisconsin Breast Cancer Dataset. Dari percobaan ini algoritma Decision Tree mampu menunjukkan akurasi hingga 90% dan akurasi seimbang hingga 89% . Namun Decision Tree kesulitan dalam memprediksi kelas minoritas (tumor ganas/ malignant /M) yang ditunjukkan pada *confusion matrix*. Sebaliknya, Random Forest yang merupakan gabungan dari beberapa Decision Tree menghasilkan akurasi dan akurasi seimbang yang lebih baik untuk memprediksi kelas minoritas. Walaupun dalam akurasi hanya meningkat sebesar 4% ternyata model mampu mengurangi kesalahan tumor ganas hingga 50%.

Dalam percobaan juga dapat dilihat visualisasi Decision Tree dalam melakukan prediksi. Terlihat bahwa model memiliki fitur serta threshold tertentu dalam mengelompokkan data. Dari kedua percobaan juga didapat fitur yang menjadi penentu dalam melakukan pemecahan data, pada kasus ini fitur “perimeter_worst” dan “area_worst” adalah fitur yang paling signifikan.

B. Ringkasan Pembahasan

Random Forest dapat menghasilkan prediksi jauh lebih baik dibanding Decision Tree karena Random Forest merupakan *ensemble learning* yang menggabung beberapa Decision Tree dalam memprediksi. Pada kasus ini Random Forest membuktikan keunggulan tersebut. Dengan menciptakan variasi antar Decision Tree, model mampu melihat pola yang lebih umum dan dapat memprediksi keluaran minoritas dan umum lebih baik serta akurat.

Peningkatan balanced accuracy pada Random Forest menegaskan pentingnya metode ensemble learning dalam kasus yang membutuhkan akurasi tinggi, seperti aplikasi medis. Balanced accuracy yang lebih tinggi juga menunjukkan bahwa Random Forest lebih efektif dalam menangani distribusi kelas yang tidak seimbang dibandingkan Decision Tree. Hal ini dikarenakan proses bootstrap sampling dan random feature selection yang meningkatkan kemampuan generalisasi model.

Selain itu, fitur "perimeter_worst" dan "area_worst" yang konsisten menjadi faktor penentu dalam kedua model membuktikan bahwa fitur tersebut memiliki relevansi yang signifikan dalam diagnosis kanker payudara. Hal ini memperkuat potensi penggunaan Random Forest sebagai alat pendukung keputusan dalam diagnosis medis.

Namun, beberapa tantangan tetap perlu diperhatikan. Distribusi data yang tidak seimbang pada dataset ini dapat menyebabkan bias dalam prediksi, meskipun sudah diperbaiki dengan penggunaan Random Forest. Dalam aplikasi nyata, penting untuk mengadopsi metode *balancing* dataset seperti *oversampling* atau *undersampling* agar model dapat memberikan hasil yang lebih optimal. Selain itu, tuning parameter lebih lanjut, seperti jumlah pohon dalam Random Forest atau kedalaman maksimum pohon, dapat meningkatkan performa model lebih jauh.

V. KESIMPULAN

Decision Tree berperan sangat penting dalam algoritma Random Forest karena sejatinya Decision Tree merupakan inti dari algoritma tersebut. Dari segi performa, penggunaan Random Forest memberikan hasil yang lebih akurat dan matang dibandingkan Decision Tree. Hal ini membuktikan betapa efektifnya metode *ensemble learning* dalam meningkatkan performa model.

Keunggulan Random Forest tidak hanya terlihat pada akurasi

yang lebih tinggi, tetapi juga pada kemampuannya untuk mengurangi kesalahan prediksi pada kelas minoritas (tumor ganas). Dengan fitur-fitur seperti "perimeter_worst" dan "area_worst" yang signifikan, model ini menunjukkan potensi besar dalam mendukung diagnosis kanker payudara.

Untuk implementasi di masa depan, penelitian ini dapat diperluas dengan menggunakan dataset lain yang lebih besar atau lebih kompleks untuk menguji generalisasi model. Selain itu, pengembangan lebih lanjut dapat mencakup penggunaan metode balancing data, eksplorasi algoritma *ensemble* lainnya, atau kombinasi dengan *deep learning* untuk meningkatkan akurasi prediksi. Penelitian ini menunjukkan bahwa teknologi *machine learning*, khususnya Random Forest, memiliki peran penting dalam membantu diagnosis medis yang lebih akurat dan cepat.

VI. LAMPIRAN

Implementasi kode program serta dataset yang digunakan dapat diakses pada link berikut : https://github.com/ryonlunar/makalah-matdis

DAFTAR PUSTAKA

[1] A. Rizka, M. K. Akbar, and N. A. Putri, "Carcinoma Mammae Sinistra T4bN2M1 Metastasis Pleura," *AVERROUS: Jurnal Kedokteran dan Kesehatan Malikussaleh*, vol. 8, no. 1, pp. 23–31, May 2022.

[2] Kementerian Kesehatan Republik Indonesia, "Deteksi dini kanker payudara dengan SADARI dan SADANIS," diakses: Jan. 1, 2025. [Online]. Tersedia: https://upk.kemkes.go.id/new/deteksi-dini-kanker-payudara-dengan-sadari-dan-sadanis#:~:text=Dengan%20menghindari%20potensi%20munculnya%20kanker,kesembuhan%20hingga%2080%2D90%25.

[3] R. Munir, "Matematika Diskrit," diakses: Des. 27, 2024. [Online]. Tersedia: https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/matdis.htm.

[4] IBM, "Decision Trees," diakses: Jan. 1, 2025. [Online]. Tersedia: https://www.ibm.com/think/topics/decision-trees.

[5] IBM, "Random Forest," diakses: Jan. 1, 2025. [Online]. Tersedia: https://www.ibm.com/id-id/topics/random-forest.

[6] Scikit-learn, "RandomForestClassifier," diakses: Jan. 8, 2025. [Online]. Tersedia: https://scikit-learn.org/1.5/modules/generated/sklearn.ensemble.RandomForestClassifier.html.

UCAPAN TERIMA KASIH

Saya berterima kasih kepada Yang Maha Esa, Allah SWT karena dengan rahmat dan bantuannya mampu menyelesaikan makalah ini. Saya juga mengucapkan terima kasih setulusnya kepada Ir. Rila Mandala, M.Eng, PhD., selaku dosen pada mata kuliah IF2120 – Matematika Diskrit karena dengan bantuan dan penjelasan beliau saya mampu menyelesaikan makalah ini.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 26 Desember 2024

Adhimas Aryo Bimo dan 13523052