KOMIK (Konferensi Nasional Teknologi Informasi dan Komputer)
Volume 9, Nomor 1, November 2025
DOI: 10.30865/komik.v9i1.9569
ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 42-49

Deteksi dan Pencegahan Fraud Internal di Perusahaan Menggunakan Algoritma Random Forest

Riyan Sanjaya[*], **Putri Ramadhani**

Fakultas Ilmu Komputer Dan Teknologi Informasi, Teknik Informatika, Universitas Budi Darma, Medan, Indonesia
Email: [1,*]reyriyansanjaya@gmail.com,[2]pramadhaniput@gmail.com
[*]) reyriyansanjaya@gmail.com

Abstrak− Fraud internal merupakan permasalahan krusial yang dapat menyebabkan kerugian finansial signifikan bagi perusahaan, termasuk CV. Smartindo Telekom. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem deteksi fraud internal berbasis data karyawan guna mendukung strategi deteksi yang efektif. Sistem dikembangkan dengan menerapkan algoritma *Random Forest* yang berlandaskan pada kerangka teori *Fraud Diamond*, mencakup empat elemen utama yaitu tekanan (*pressure*), kesempatan (*opportunity*), rasionalisasi (*rationalization*), dan kemampuan (*capability*). Data penelitian terdiri atas indikator finansial dan operasional seperti selisih stok, jumlah transaksi, hutang, rotasi kerja, absensi, telat absen, lembur tanpa pengawasan, transaksi anomali, pelanggaran disiplin, jabatan dan lama bekerja. Melalui teknik *bootstrap sampling* sebanyak 1.000 kali, dihasilkan 1.000 pohon keputusan untuk membentuk model klasifikasi yang stabil dan akurat. Sistem diimplementasikan dalam aplikasi berbasis web menggunakan *framework Flask* dengan visualisasi interaktif hasil analisis. Hasil evaluasi menunjukkan tingkat akurasi, presisi, *recall*, dan *F1-score* mencapai 100%, dengan faktor dominan yang memengaruhi potensi fraud adalah selisih stok, jumlah transaksi, hutang, dan rotasi kerja. Penelitian ini berhasil menghasilkan sistem deteksi fraud yang efektif sekaligus memberikan kontribusi nyata terhadap penguatan strategi pencegahan fraud internal.

Kata Kunci: Deteksi Fraud; Fraud Internal; Random Forest; Fraud Diamond; Pembelajaran Mesin

Abstract− Internal fraud is a critical issue that can lead to significant financial losses for companies, including CV. Smartindo Telekom. This study aims to design and implement an internal fraud detection system based on employee data to support effective detection strategies. The system is developed using the Random Forest algorithm grounded in the Fraud Diamond framework, which consists of four key elements: pressure, opportunity, rationalization, and capability. The research data includes financial and operational indicators such as stock discrepancies, number of transactions, debt, job rotation, attendance records, late check-ins, unsupervised overtime, anomalous transactions, disciplinary violations, job position, and length of employment. Through 1,000 bootstrap sampling iterations, a total of 1,000 decision trees were generated to build a stable and accurate classification model. The system is implemented in a web-based application using the Flask framework with interactive analytical visualizations. Evaluation results show accuracy, precision, recall, and F1-score reaching 100%, with dominant factors influencing fraud potential identified as stock discrepancies, number of transactions, debt, and job rotation. This study successfully produces an effective fraud detection system while contributing significantly to strengthening internal fraud prevention strategies.

Keywords: Fraud Detection; Internal Fraud; Random Forest; Fraud Diamond; Machine Learning

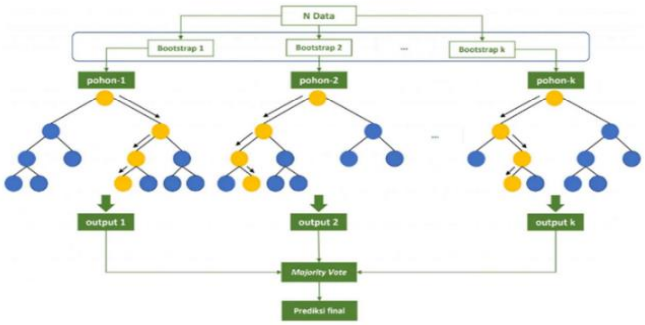
1. PENDAHULUAN

Integritas menjadi fondasi penting dalam menjaga keberlanjutan bisnis modern, namun kecurangan internal (internal fraud) masih menjadi tantangan serius bagi banyak organisasi. CV. Smartindo Telekom, perusahaan yang bergerak di bidang distribusi produk telekomunikasi, menghadapi risiko fraud dari aktivitas karyawan, khususnya tenaga penjualan (sales), seperti manipulasi data, penggelapan hasil penjualan, dan penyalahgunaan stok barang. Tekanan yang mengacu pada dorongan internal yang membuat seseorang terdorong untuk melakukan kecurangan, yang biasanya berakar dari masalah finansial, beban pekerjaan yang terlalu tinggi, atau kebutuhan pribadi yang mendesak. Kasus-kasus seperti ini tidak hanya berdampak pada kerugian finansial, tetapi juga menurunkan kepercayaan pelanggan, mengganggu stabilitas operasional, dan merusak reputasi perusahaan di mata pemangku kepentingan. Untuk memahami faktor penyebab fraud, berbagai model teoretis telah dikembangkan, seperti *Fraud Triangle*, *Fraud Diamond*, dan *Fraud Pentagon*. Dari ketiganya, *Fraud Diamond Theory* dinilai paling relevan karena mencakup empat elemen penyebab utama, yaitu *Pressure* (tekanan), *Opportunity* (kesempatan), *Rationalization* (pembenaran), dan *Capability* (kemampuan) [1]. Model ini memberikan kerangka konseptual untuk memetakan potensi risiko dan memahami motivasi pelaku secara komprehensif.

Dari sisi teknologi, perkembangan *machine learning* telah membuka peluang baru dalam deteksi dini terhadap aktivitas kecurangan. *Mechine learning* juga dipakai untuk prediksi penempatan karir dengan model klasifikasi seperti Random Forest dan SVM [2]. Salah satu algoritma yang terbukti unggul dalam mendeteksi pola anomali adalah *Random Forest*, yang merupakan metode *ensemble learning* berbasis sejumlah *decision tree* dengan teknik *bootstrap aggregation* untuk meningkatkan akurasi dan mengurangi *overfitting* [3]. *Random Forest* diperkenalkan oleh Leo Breiman (2001) sebagai metode *ensemble* berbasis sekumpulan *decision tree*. Setiap pohon dibangun dengan pemilihan atribut acak tanpa *pruning*, dan klasifikasi akhir ditentukan lewat voting mayoritas. *Random Forest* meningkatkan akurasi melalui *bootstrap resampling* dan *random feature selection*, serta efektif pada data tidak lengkap[4]. Metode ini unggul dalam menangani data fraud yang tidak seimbang dan *nonlinier*, sehingga ideal untuk deteksi prediktif [5] Secara struktural, *Random Forest* merupakan kumpulan (*ensemble*) pohon keputusan yang bekerja paralel dan

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 42-49

independen, lalu digabungkan untuk menghasilkan prediksi akhir. Berikut ini adalah ilustrasi struktur dari algoritma *Random Forest*:



Gambar 1. Struktur *Random Forest*

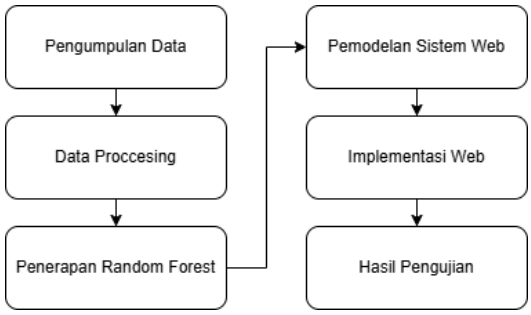
Beberapa penelitian sebelumnya menunjukkan efektivitas dari algoritma *Random Forest* dalam mendeteksi fraud di berbagai sektor. Ravina Armiani (2022) menerapkan *Random Forest* untuk mendeteksi penipuan transaksi kartu kredit dan memperoleh akurasi sebesar 90,68% dengan nilai *F1-score* mencapai 90,23% [6]. Sarmini dkk. menegaskan bahwa *Random Forest* dan *XGBoost* efektif untuk mendeteksi penipuan pada *e-commerce*, meskipun diperlukan data *augmentation* untuk mengatasi ketidakseimbangan data [7]. Penelitian Chengwei Liu (2015) membandingkan *Random Forest*, *Logistic Regression*, dan *SVM*, dan menyimpulkan bahwa *Random Forest* menghasilkan akurasi tertinggi dalam mendeteksi transaksi mencurigakan [8]. Selanjutnya, Fauziah (2022) menunjukkan bahwa *Random Forest* mencapai akurasi hampir sempurna pada data pelatihan maupun pengujian dalam kasus deteksi fraud kartu kredit [9]. Penelitian terbaru oleh Kurniawan (2025) juga menegaskan keunggulan algoritma ini dalam mendeteksi penipuan pada transaksi daring berskala besar dan kompleks [10]. Dalam konteks deteksi fraud, algoritma ini mampu mengenali pola data kompleks dengan tingkat keberhasilan yang tinggi, bahkan pada dataset yang tidak seimbang

Berdasarkan tinjauan tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem deteksi serta pencegahan fraud internal menggunakan algoritma *Random Forest* di CV. Smartindo Telekom, dengan mengintegrasikan kerangka teori *Fraud Diamond* sebagai dasar analisis faktor risiko. Sistem yang dikembangkan tidak hanya berfungsi mendeteksi potensi fraud secara akurat, tetapi juga memberikan rekomendasi strategis untuk pencegahan, dengan menyoroti faktor dominan seperti selisih stok, jumlah transaksi, hutang, dan rotasi kerja. Dengan demikian, penelitian ini berkontribusi terhadap penguatan kontrol internal perusahaan melalui pemanfaatan kecerdasan buatan untuk pengambilan keputusan yang lebih berbasis data.

2. METODE PENELITIAN

2.1 Tahapan Penelitian

Kerangka kerja penelitian ini dirancang untuk memberikan gambaran sistematis mengenai tahapan-tahapan dalam penerapan algoritma *Random Forest* guna mendeteksi potensi fraud.



Gambar 2. Tahapan Penelitian

Penjelasan dari tahapan penelitian sebagai berikut:

1. Pengumpulan data pada penelitian ini menggunakan data karyawan internal CV. Smartindo Telekom yang mencakup indikator *Fraud Diamond (Pressure, Opportunity, Rationalization, Capability)*.
2. *Data Processing* dimulai dengan mengidentifikasi 13 variabel independen yang diklasifikasikan berdasarkan empat elemen *Fraud Diamond*, yaitu *Pressure, Opportunity, Rationalization*, dan *Capability*. Selanjutnya dilakukan pra-pemrosesan data melalui normalisasi variabel numerik menggunakan *Min-Max Scaling*, pengkodean variabel kategorikal ke dalam bentuk biner atau numerik. Tahap berikutnya adalah penentuan label fraud, yaitu memberikan nilai 1 jika karyawan memenuhi minimal dua indikator *Fraud Diamond* dan nilai 0 jika kurang dari dua indikator

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 42-49

terpenuhi. Setelah seluruh data bersih, ternormalisasi, dan terstruktur, data tersebut digunakan sebagai input untuk pemodelan menggunakan algoritma *Random Forest*.

3. Pada tahapan ini, dilakukan penerapan algoritma *Random Forest* untuk mendeteksi indikasi fraud meliputi:
   a. Pembagian data *dataset* dibagi menjadi data latih (*training*) dan data uji (*testing*), umumnya dengan rasio 80:20.
   b. Algoritma *Random Forest* dilatih menggunakan data latih untuk mengenali pola-pola yang berkaitan dengan indikasi fraud
   c. Prediksi dan voting setiap pohon keputusan dalam *Random Forest* memberikan prediksi masing-masing, kemudian hasil akhir ditentukan berdasarkan voting mayoritas (*majority voting*).
4. Tahap ini, penulis akan merancang sebuah sistem website menerapakan teori *Fraud Diamond* dan menggunakan algoritma *Random Forest* untuk mendeteksi fraud dengan data sampel yang ada.
5. Pada tahap ini peneliti akan mengimplementasikan data yang telah didapat dan akan diolah oleh sistem untuk menghasilkan data laporan hasil fraud.
6. Setelah melakukan semua tahapan maka hasil pengujian dijabarkan untuk memberikan data informatif tentang karyawan yang berpotensi fraud dan fitur apa saja yang paling dominan sebagai pemicu fraud, dimana tersebut ditindak lanjuti sebagai peningkatan pencegahan fraud di perusahaan.

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Sampel data terdiri dari 30 karyawan sales CV. Smartindo Telekom, Berikut adalah data sampel yang telah didapat, dapat dilihat pada data dibawah ini:

Tabel 1. Sampel Data Karyawan

No	NK	JL	H(Jt)	A (%)	TA(%)	RK	LA	PD	LK	JM	NT (Jt)	TS	ST
1	K1	Sales Senior	12	90	12	Ya	5	1	6	150	225	8	12
2	K2	Sales Junior	0	95	5	Tidak	1	0	2	95	650	1	0
3	K3	Sales Senior	7	80	20	Ya	4	1	5	120	890	5	6
4	K4	Sales Junior	0	98	2	Tidak	0	0	1	80	520	0	0
5	K5	Sales Senior	15	75	25	Ya	7	2	7	160	411	10	15
6	K6	Sales Junior	4	92	8	Ya	2	1	3	110	760	3	2
7	K7	Sales Junior	0	85	18	Tidak	0	0	2	85	590	0	0
8	K8	Sales Senior	9	96	4	Ya	5	1	6	140	353	7	9
9	K9	Sales Junior	0	78	22	Tidak	0	0	2	92	610	1	0
10	K10	Sales Senior	18	94	6	Ya	8	3	8	170	355	12	18
30	K11	Sales Senior	16	93	8	Ya	8	3	8	165	482	11	16

Keterangan :

NK	: Nama Karyawan	RK	: Rotasi Kerja	NT	: Nilai Transaksi
JL	: Jabatan Level	LA	: Lembur tampa Awasi	TS	: Transaksi Anomali
H	: Hutang (Jt)	PD	: Pelanggaran Disiplin	ST	: Selisih Stok
A	: Absensi	LK	: Lama kerja		
TA	: Telat Absen	JM	: Jumlah Transaksi		

3.1.1 Data Proccesing

Penelitian ini menggunakan 13 variabel independen (X) yang dikelompokkan berdasarkan empat elemen utama dalam teori Fraud Diamond. Berikut adalah tabel lengkap dengan Target Y sesuai aturan *Fraud Diamond* yang sudah ditentukan diatas:

Tabel 2. Data Setelah Dilakukan Data *Proccesing*

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 42-49

NK	JL	H(Jt)	A (%)	TA(%)	RK	LA	PD	LK	JM	NT (Jt)	TS	ST	Target (Y)
K1	2	0,6	0,65	0,42	1	5	1	6	0,71	0	0,57	0,6	1
K2	1	0	0,87	0,13	0	1	0	2	0,17	0,57	0,07	0	0
K3	2	0,35	0,22	0,75	1	4	1	5	0,41	0,89	0,36	0,3	1
K4	1	0	1	0	0	0	0	1	0,02	0,4	0	0	0
K5	2	0,75	0	0,96	1	7	2	7	0,8	0,25	0,71	0,75	1
K6	1	0,2	0,74	0,25	1	2	1	3	0,31	0,72	0,21	0,1	1
K7	1	0	0,43	0,67	0	0	0	2	0,07	0,49	0	0	0
K8	2	0,45	0,91	0,08	1	5	1	6	0,61	0,17	0,5	0,45	1
K9	1	0	0,13	0,83	0	0	0	2	0,14	0,52	0,07	0	0
K10	2	0,9	0,83	0,17	1	8	3	8	0,9	0,17	0,86	0,9	1
K30	2	0,8	0,78	0,25	1	8	3	8	0,85	0,34	0,79	0,8	1

3.2 Penerapan Algoritma Random Forest

3.2.1 *Bootstrap Sampling (Bagging)*

Tahap awal pembentukan model *Random Forest* dilakukan dengan metode *bootstrap sampling* atau *bagging*. Dari dataset latih berjumlah 24 data (K1–K24), dibentuk tiga dataset *bootstrap* baru dengan ukuran yang sama, yaitu S1, S2 sampai dengan S1000. Masing-masing dataset diperoleh dengan cara pengambilan sampel acak dengan pengembalian (*random sampling with replacement*), sehingga memungkinkan adanya duplikasi data di setiap *subset*. Dataset S1, S2 sampai dengan S1000 kemudian digunakan sebagai basis pembentukan 1000 pohon keputusan yang berbeda dalam *ensemble Random Forest*.

3.2.2 Pembentukan Pohon Keputusan (CART)*)*

Disini penulis hanya membuat 2 pembentukan pohon keputusan saja sebagai gambaran tiap pohon dibuat. Setiap dataset *bootstrap* digunakan untuk membentuk satu pohon keputusan menggunakan metode *Classification and Regression Tree (CART)*.

1. Pembentukan pohon 1 berdasarkan data bootstrap ke 1:
   Jumlah sampel: 24 (Terdapat duplikat)
   Jumlah sampel unik: 13
   value = [7, 17] = 7 data Tidak Fraud dan 17 data Fraud
   Threshold: Fitur (Pelanggaran Disiplin) ≤ 0.5,
   Mencari nilai *threshold*:
   Urutkan nilai Fitur unik, kecil ke besar: 0, 1, 2, 3
   Ambil titik tengah antar nilai unik sebagai kandidat *threshold*: (0 + 1) / 2 = 0,5 ← *Threshold*
   (1 + 2) / 2 = 1,5
   (2 + 3) / 2 = 2,5
   Threshold terbaik adalah 0.5 karena menghasilkan *Gini Split* terkecil: 0.00
   Proporsi:
   P_1 = 7/ 24 = 0.29
   P_2 = 17 / 24 = 0.71
   Gini Sebelum *Split*:

$$Gini(D) = 1 - (P_1)^2 - (P_2)^2 = 1 - (0.29)^2 - (0.71)^2$$
$$= 1 - 0.09 - 0.050 = 0.413$$

   Split dilakukan pada fitur Pelanggaran Disiplin ≤ 0.5 menghasilkan dua cabang:
   Daun Kiri (*True* / Pelanggaran Disiplin ≤ 0.5):
   Jumlah sampel: 7 (Terdapat duplikat)
   Jumlah sampel unik: 5
   Value = [7, 0] → 7 data Tidak Fraud, 0 Data Fraud
   Proporsi:
   P_1 = 7 /7 = 1
   P_2 = 0 / 7 = 0
   Gini Sebelum *Split*:

$$Gini(D) = 1 - (P_1)^2 - (P_2)^2 = 1 - (1)^2 - (0)^2$$
$$= 1 - 1 - 0 = 0$$

Kanan (*False* / Pelanggaran Disiplin ≤ 0.5):
Jumlah sampel: 17 (Terdapat duplikat)
Jumlah sampel unik: 8
Value = [0, 17] → 0 data Tidak Fraud, 17 Fraud

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 42-49

Proporsi:

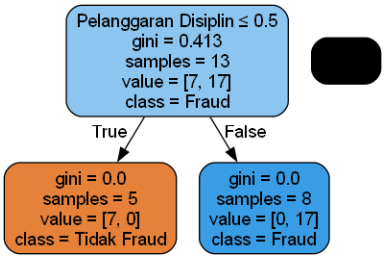
$P_1 = 0 / 17 = 0$

$P_2 = 17 / 17 = 1$

Gini Sebelum *Split*:

$$Gini(D) = 1 - (P_1)^2 - (P_2)^2 = 1 - (0)^2 - (1)^2$$
$$= 1 - 0 - 1 = 0$$

Berikut adalah pohon 1 *random forest*:



Gambar 3. Pohon 1

2. Pembentukan pohon 2 berdasarkan data *bootstrap* ke 2:

Jumlah sample: 24 (Terdapat duplikat)

Jumlah sample unik: 15

value = [11, 13] = 11 data tidak Fraud, 13 Data Fraud

Threshold: Lama Kerja (th) ≤ 2.5,

Mencari nilai *threshold*:

Urutkan nilai Lama Kerja (th) unik, kecil ke besar: [1, 2, 3, 4, 5, 6, 7,8]

Ambil titik tengah antar nilai unik sebagai kandidat *threshold*: (1 + 2) / 2 = 1,5

(2 + 3) / 2 = 2,5 ← *Threshold*

(3 + 4) / 2 = 3,5

(4 + 5) / 2 = 4,5

(5 + 6) / 2 = 5,5

(6 + 7) / 2 = 6,5

(7 + 8) / 2 = 7,5

Threshold terbaik adalah 2.5, karena menghasilkan *Gini Split* terkecil yaitu 0.14

Proporsi:

$P_1 = 11 / 24 = 0.458$

$P_2 = 13 / 24 = 0.542$

Gini Sebelum *Split*:

$$Gini(D) = 1 - (P_1)^2 - (P_2)^2 = 1 - (0.458)^2 - (0.542)^2$$
$$= 1 - 0.210 - 0.293 = 0.497$$

Split dilakukan pada fitur Lama Kerja (th) ≤ 2.5 menghasilkan dua cabang: Kiri (*False* / Lama Kerja (th) ≤ 2.5):

Jumlah sampel: 11 (Terdapat duplikat)

Jumlah sampel unik: 6

Value = [11, 0] → 14 data Tidak Fraud, 0 Fraud Proporsi:

$P_1 = 11 / 11 = 1$

$P_2 = 0 / 11 = 0$

Gini Sebelum *Split*:

$$Gini(D) = 1 - (P_1)^2 - (P_2)^2 = 1 - (1)^2 - (0)^2$$
$$= 1 - 1 - 0 = 0$$

Kanan (*True* / Lama Kerja (th) ≤ 2.5):

Jumlah sampel: 13 (Terdapat duplikat)

Jumlah sampel unik: 9

Value = [0, 13] → 0 data Tidak Fraud, 13 data Fraud

Proporsi:

$P_1 = 0 / 13 = 0$

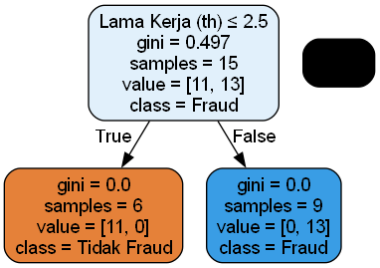
$P_2 = 13 / 13 = 1$

Gini sebelum *Split*:

$$Gini(D) = 1 - (P_1)^2 - (P_2)^2 = 1 - (0)^2 - (1)^2$$
$$= 1 - 0 - 1 = 0$$

Berikut adalah pohon 2 random forest:

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 42-49



Gambar 4. Pohon 2

3.2.3 Proses Voting

Tahap ini, di mana masing-masing pohon yang telah dilatih memberikan prediksi, lalu hasil akhirnya ditentukan berdasarkan mayoritas suara (*voting*) pada data uji. Data uji setelah dilakukan data *proccesing* (Data K25 – K30). Penulis akan membuat perhitungan dan penelusuran (*trace*) untuk setiap data uji pada tiga model *random forest* yang diberikan. Berikut adalah *trace voting random forest*:

1. *Trace Voting Random Forest* pada Data Uji K25
   a. Pohon 1: *Variabel* Pelanggaran Disiplin = 0, nilai *threshold* ≤ 0.5. Karena 0 ≤ 0.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   b. Pohon 2: *Variabel* Lama Kerja = 2 tahun, *threshold* ≤ 2.5. Karena 2 ≤ 2.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   c. Pohon 3: *Variabel* Rotasi Kerja = 0, *threshold* ≤ 0.5. Karena 0 ≤ 0.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   Voting mayoritas: 3 suara Tidak Fraud → Prediksi akhir = Tidak Fraud. Sesuai dengan label aktual (Target Y = 0).
2. *Trace Voting Random Forest* pada Data Uji K26
   a. Pohon 1: Pelanggaran Disiplin = 1, *threshold* ≤ 0.5. Karena 1 > 0.5, maka mengikuti cabang *False* → prediksi = Fraud.
   b. Pohon 2: Lama Kerja = 4 tahun, *threshold* ≤ 2.5. Karena 4 > 2.5, maka mengikuti cabang *False* → prediksi = Fraud.
   c. Pohon 3: Rotasi Kerja = 1, *threshold* ≤ 0.5. Karena 1 > 0.5, maka mengikuti cabang *False* → prediksi = Fraud.
   Voting mayoritas: 3 suara Fraud → Prediksi akhir = Fraud. Sesuai dengan label aktual (Target Y = 1).
3. *Trace Voting Random Forest* pada Data Uji K27
   a. Pohon 1: Pelanggaran Disiplin = 0, *threshold* ≤ 0.5. Karena 0 ≤ 0.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   b. Pohon 2: Lama Kerja = 1 tahun, *threshold* ≤ 2.5. Karena 1 ≤ 2.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   c. Pohon 3: Rotasi Kerja = 0, *threshold* ≤ 0.5. Karena 0 ≤ 0.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   Voting mayoritas: 3 suara Tidak Fraud → Prediksi akhir = Tidak Fraud. Sesuai dengan label aktual (Target Y = 0).
4. *Trace Voting Random Forest* pada Data Uji K28
   a. Pohon 1: Pelanggaran Disiplin = 2, *threshold* ≤ 0.5. Karena 2 > 0.5, maka mengikuti cabang *False* → prediksi = Fraud.
   b. Pohon 2: Lama Kerja = 7 tahun, *threshold* ≤ 2.5. Karena 7 > 2.5, maka mengikuti cabang *False* → prediksi = Fraud.
   c. Pohon 3: Rotasi Kerja = 1, *threshold* ≤ 0.5. Karena 1 > 0.5, maka mengikuti cabang *False* → prediksi = Fraud.
   Voting mayoritas: 3 suara Fraud → Prediksi akhir = Fraud. Sesuai dengan label aktual (Target Y = 1).
5. *Trace Voting Random Forest* pada Data Uji K29
   a. Pohon 1: Pelanggaran Disiplin = 0, *threshold* ≤ 0.5. Karena 0 ≤ 0.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   b. Pohon 2: Lama Kerja = 2 tahun, *threshold* ≤ 2.5. Karena 2 ≤ 2.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   c. Pohon 3: Rotasi Kerja = 0, *threshold* ≤ 0.5. Karena 0 ≤ 0.5, maka mengikuti cabang *True* → prediksi = Tidak Fraud.
   Voting mayoritas: 3 suara Tidak Fraud → Prediksi akhir = Tidak Fraud. Sesuai dengan label aktual (Target Y = 0).
6. *Trace Voting Random Forest* pada Data Uji K30
   a. Pohon 1: Pelanggaran Disiplin = 3, *threshold* ≤ 0.5. Karena 3 > 0.5, maka mengikuti cabang *False* → prediksi = Fraud.
   b. Pohon 2: Lama Kerja = 8 tahun, *threshold* ≤ 2.5. Karena 8 > 2.5, maka mengikuti cabang *False* → prediksi = Fraud.
   c. Pohon 3: Rotasi Kerja = 1, *threshold* ≤ 0.5. Karena 1 > 0.5, maka mengikuti cabang *False*→ prediksi =

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 42-49

Fraud.

Voting mayoritas: 3 suara Fraud → Prediksi akhir = Fraud. Sesuai dengan label aktual (Target Y = 1).

3.2.4 Evaluasi Model Random Forest

Evaluasi kinerja model *Random Forest* dalam penelitian ini penulis menggunakan *confusion matrix*. *Confusion matrix* digunakan untuk menggambarkan jumlah prediksi yang benar dan salah berdasarkan kelas aktual. Berikut dibawah ini adalah rumusnya:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} = \frac{3 + 3}{3 + 3 + 0 + 0} = 100\%$$

$$Precision = \frac{TP}{TP + FP} = \frac{3}{3 + 0} = 100\%$$

$$Recall = \frac{TP}{TP + FN} \quad \frac{3}{3 + 0} = 100\%$$

$$F1\ Score = \frac{2\ x\ Precission\ x\ Recal}{Precission + FP} \quad \frac{2\ x\ (\ 1\ x\ 1\ )}{2 + 0} = 100\%$$

Tujuan dari contoh ini adalah untuk mempermudah pembaca memahami proses pembentukan pohon, pemilihan atribut, serta mekanisme voting mayoritas dalam menghasilkan keputusan akhir. Penggunaan jumlah pohon yang sedikit dipilih agar penjelasan algoritma dapat divisualisasikan secara sederhana tanpa menimbulkan kompleksitas yang berlebihan[11].

3.2.5 Hasil Pengujian

Pada tahap pengujian ini, penulis membangun model Random Forest menggunakan 1.000 data *bootstrap sampling* dan 1.000 pohon keputusan untuk meningkatkan stabilitas model dan menurunkan kesalahan klasifikasi. Hasil pembangunan 1.000 pohon dari data *bootstrap* kemudian digabungkan menggunakan mayoritas suara (*voting*) untuk menentukan prediksi akhir. Analisis dari model *random forest* adalah sebagai berikut:

1. *Accuracy* model: 100.0%
2. *Precision* Fraud: 100.0%
3. *Recall* Fraud: 100.0%
4. *F1-Score* Fraud: 100.0%

Berdasarkan evaluasi di atas:

1. Jika *precision* dan recall mendekati 100%, model *random forest* bekerja sangat baik dalam mendeteksi fraud.
2. Jika *precision* tinggi tapi recall rendah, model akurat tapi melewatkan banyak fraud.
3. Jika *recall* tinggi tapi precision rendah, model banyak mendeteksi fraud tapi juga menghasilkan *false positive*.
4. *F1-Score* memberikan keseimbangan antara *precision* dan *recall*.

Eksperimen menunjukkan bahwa jumlah pohon dalam algoritma *random forest* sangat berpengaruh signifikan terhadap stabilitas prediksi dan jumlah fitur yang teridentifikasi. Model dengan 1.000 pohon menghasilkan probabilitas prediksi yang konsisten (mendekati 0% atau 100%) serta mengidentifikasi lebih banyak fitur penting. Sebaliknya, model dengan 3 pohon memperlihatkan probabilitas yang fluktuatif, sehingga hasil voting rentan terhadap variasi antar pohon dan hanya menghasilkan sedikit fitur penting. Temuan ini mendukung prinsip dasar *Random Forest* bahwa penambahan jumlah pohon meningkatkan akurasi dan stabilitas prediksi, meskipun peningkatannya akan melandai setelah jumlah tertentu.

4. KESIMPULAN

Model Random Forest terbukti sangat efektif untuk deteksi fraud internal, dibuktikan oleh capaian metrik evaluasi sempurna: Accuracy 100%, Precision 100%, Recall 100%, dan F1-Score 100%. Kinerja ini ditopang konfigurasi 1.000 pohon keputusan yang menghasilkan prediksi probabilitas tegas dan konsisten, mendekati 100% pada aktivitas berisiko tinggi serta mendekati 0% pada aktivitas normal. Dengan demikian, model mampu mengidentifikasi indikasi fraud pada karyawan berisiko sebagai dasar tindakan pencegahan. Analisis feature importance memvalidasi kerangka Fraud Diamond dan menunjukkan bahwa elemen Opportunity (kesempatan) adalah pendorong paling dominan, ditandai pentingnya variabel Selisih Stok, Rotasi Kerja, Lembur Tanpa Pengawasan, dan Transaksi Anomali. Temuan ini menegaskan bahwa celah prosedur dan lemahnya pengawasan menjadi pemicu utama. Elemen Pressure (tekanan) dan Capability (kemampuan) juga berpengaruh kuat, terutama Hutang sebagai indikator tekanan finansial dan Jumlah Transaksi sebagai indikator kapabilitas operasional. Sebaliknya, indikator Rationalization seperti Pelanggaran Disiplin cenderung memiliki kontribusi lebih rendah. Implikasi praktisnya, perusahaan sebaiknya memfokuskan kontrol pada penguatan manajemen stok, penegakan rotasi kerja, perbaikan tata kelola lembur, serta pemantauan transaksi anomali. Selain itu, diperlukan pengawasan atau audit tambahan bagi karyawan dengan hutang tinggi atau volume transaksi sangat besar, agar risiko dapat ditekan secara terukur. Langkah tersebut dapat diintegrasikan ke kebijakan kepatuhan,

ISSN 2597-4645 (media online)
ISSN 2597-4610 (media cetak)
Page: 42-49

dashboard risiko, dan pelatihan internal, sehingga deteksi dini lebih cepat, biaya investigasi menurun, dan budaya anti-fraud semakin kuat berkelanjutan.

REFERENCES

[1] D. T. Wolfe and D. R. Hermanson, "The Fraud Diamond: Considering the Four Elements of Fraud," 2004. [Online]. Available: https://digitalcommons.kennesaw.edu/facpubs

[2] H. Mahmud Nawawi, A. Baitul Hikmah, A. Mustopa, and G. Wijaya, "Model Klasifikasi Machine Learning untuk Prediksi Ketepatan Penempatan Karir," *Jurnal SAINTEKOM*, vol. 14, no. 1, pp. 13–25, Mar. 2024, doi: 10.33020/saintekom.v14i1.512.

[3] L. Breiman, "Random Forests," *Mach Learn*, vol. 45, no. 1, pp. 5–32, Oct. 2001, doi: 10.1023/A:1010933404324.

[4] Suci Amaliah, M. Nusrang, and A. Aswi, "Penerapan Metode Random Forest Untuk Klasifikasi Varian Minuman Kopi di Kedai Kopi Konijiwa Bantaeng," *VARIANSI: Journal of Statistics and Its application on Teaching and Research*, vol. 4, no. 3, pp. 121–127, Dec. 2022, doi: 10.35580/variansiunm31.

[5] W. A. E. Sawadogo, R. Kafando, N. Bado, T. S. M. Ky, and T. F. Bissyande, "Advanced IoT-AI Integration for Predictive Management of Positive Temperature Cold Storage Systems," 2025, pp. 16–29. doi: 10.1007/978-3-031-96228-8_2.

[6] R. Armiani and E. P. Agustini, "Analisa Fraud Pada Transaksi Kartu Kredit Menggunakan Algoritma Random Forest," 2022. [Online]. Available: https://doi.org/10/25047/jtit.v9i2.297

[7] S. Sarmini, S. Sunardi, and A. Fadlil, "Performa Random Forest dan XGBoost pada Deteksi Penipuan E-Commerce Menggunakan Augmentasi Data CGAN," *Building of Informatics, Technology and Science (BITS)*, vol. 6, no. 3, pp. 1919–1931, Dec. 2024, doi: 10.47065/bits.v6i3.6430.

[8] C. Liu, Y. Chan, S. H. Alam Kazmi, and H. Fu, "Financial Fraud Detection Model: Based on Random Forest," *Int J Econ Finance*, vol. 7, no. 7, Jun. 2015, doi: 10.5539/ijef.v7n7p178.

[9] M. Fauziah and Y. F. Wijaya, "Fraud Detection Using Random Forest Classifier, Logistic Regression, and Gradient Boosting Classifier Algorithms on Credit Cards," 2022.

[10] U. N. W. Gerry William Mathew Kurniawan, "Pendeteksian Penipuan Menggunakan Pendekatan Metode Klasifikasi Random Forest," in *e-Proceeding of Engineering*, Feb. 2025, p. 2216.

[11] L. Rokach and O. Maimon, *Data Mining with Decision Trees*, vol. 81. WORLD SCIENTIFIC, 2014. doi: 10.1142/9097.

[12] S. Nabila, A. R. Putri, A. Hafizhah, F. H. Rahmah, and R. Muslikhah, "Pemodelan Diagram UML Pada Perancangan Sistem Aplikasi Konsultasi Hewan Peliharaan Berbasis Android (Studi Kasus: Alopet)," *Jurnal Ilmu Komputer dan Bisnis*, vol. 12, no. 2, pp. 130–139, Nov. 2021, doi: 10.47927/jikb.v12i2.150.