JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 12 No. 3, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v12i3.4910

DETEKSI ANOMALI MENGGUNAKAN *ENSEMBLE LEARNING* DAN *RANDOM OVERSAMPLING* PADA PENIPUAN TRANSAKSI KEUANGAN

Dewa Raka Krisna Saputra[1], Yisti Vita Via[2], Andreas Nugroho Sihananto[3]

[1,2,3]Universitas Pembangunan Nasional "Veteran" Jawa Timur; Jl. Rungkut Madya No.1, Gn. Anyar, Kec. Gn. Anyar, Surabaya, Jawa Timur 60294, Telp. (031) 8706369

Received: 12 Juli 2024
Accepted: 31 Juli 2024
Published: 7 Agustus 2024

Keywords:
Transaksi Keuangan, Deteksi Anomali, *Ensemble Learning*, *Random Oversampling*, *Machine Learning*.

Corespondent Email:
yistivia.if@upnjatim.ac.id

Abstrak. Di era digital, transaksi keuangan semakin beralih ke metode nontunai, karena sifatnya yang nyaman dan efisien. Namun, peningkatan penggunaan kartu kredit dan transaksi *online* juga meningkatkan risiko kejahatan finansial. Penelitian ini mengkaji metode *ensemble learning* dan *random oversampling* dalam mendeteksi anomali pada transaksi keuangan, khususnya penipuan kartu kredit. Algoritma klasifikasi yang digunakan meliputi *Decision Tree* (DT), *Random Forest* (RF), *Logistic Regression* (LR), dan *Naive Bayes* (NB), dengan pendekatan *ensemble learning* seperti *Bagging*, *Boosting*, dan *Stacking*. Hasil penelitian menunjukkan bahwa metode *ensemble learning* secara signifikan meningkatkan performa deteksi penipuan dibandingkan model dasar (*base model*). Khususnya teknik *stacking* menunjukkan peningkatan AUC yang signifikan, dengan beberapa algoritma mencapai AUC sempurna (1.00). *Random Forest* (RF) dengan metode *ensemble learning* menunjukkan performa yang sangat konsisten dan optimal dalam mendeteksi anomali penipuan. Penelitian ini menegaskan bahwa metode *ensemble learning*, terutama *stacking*, efektif dalam membedakan antara transaksi sah dan mencurigakan, sehingga dapat diandalkan untuk deteksi penipuan keuangan.

Abstract. *In the digital era, financial transactions are increasingly turning to non-cash methods, because they are convenient and efficient. However, increased use of credit cards and online transactions also increases the risk of financial crime. This research examines ensemble learning and random oversampling methods in detecting anomalies in financial transactions, especially credit card fraud. The classification algorithms used include Decision Tree (DT), Random Forest (RF), Logistic Regression (LR), and Naive Bayes (NB), with ensemble learning approaches such as Bagging, Boosting, and Stacking. The research results show that the ensemble learning method significantly improves fraud detection performance compared to the base model. In particular, stacking techniques show significant AUC improvements, with some algorithms achieving perfect AUC (1.00). Random Forest (RF) with the ensemble learning method shows very consistent and optimal performance in detecting fraud anomalies. This research confirms that ensemble learning methods, especially stacking, are effective in distinguishing between legitimate and suspicious transactions, making them reliable for financial fraud detection.*

1. PENDAHULUAN

Pertumbuhan ekonomi pesat membuat transaksi jual beli online lebih praktis dan mengurangi penggunaan uang tunai[1]. Transaksi nontunai, terutama kartu kredit, menjadi populer karena kemudahannya[2]. Kartu kredit menjadi salah satu pilihan popular dalam penggunaan media transaksi nontunai dan menggantikan uang tunai dalam proses jual-beli barang dan jasa di berbagai tempat [1].

Pada Desember 2021, nilai transaksi kartu kredit mencapai Rp25,91 triliun[1]. namun risiko penipuan meningkat, seperti kasus penipuan kartu kredit BNI yang menyebabkan kerugian Rp1 miliar[3].Lalu, penipuan transaksi berupa transaksi elektronik, tiket transportasi, pembayaran, penjualan, dan kartu kredit. Berdasarkan hal tersebut, deteksi penipuan transaksi keuangan sangat dibutuhkan. Salah satu alat yang digunakan, yaitu dengan pendekatan *machine learning* untuk mendeteksi anomali dalam transaksi keuangan [4].

Deteksi anomali adalah proses pencarian data dengan perilaku yang sangat berbeda dari biasanya [5]. Ini dapat dilakukan dengan algoritma klasifikasi seperti *Decision Tree* (DT), *Random Forest* (RF), *Bayesian Network* (BN), *Naïve Bayes* (NB), *Support Vector Machine* (SVM), dan lainnya. Algoritma ini bekerja dengan mendeteksi transaksi sesuai pola yang ditentukan dari data transaksi sah dan penipuan [4]. Untuk hasil yang lebih baik, metode *ensemble learning* digunakan pada penelitian ini, yaitu dengan menggabungkan beberapa model, seperti *Bagging*, *Boosting*, dan *Stacking* [6]. Penelitian ini membandingkan algoritma *base learning* dan *ensemble learning* dalam mendeteksi penipuan transaksi keuangan.

Penelitian oleh Sudiyarno, dkk. menunjukkan bahwa penggunaan *ensemble learning* menghasilkan nilai akurasi sebesar 96,8% dalam mendeteksi anomali, dan 77,4% jika hanya menggunakan *single classifier* (Naïve Bayes) [7]. Penelitian oleh Situmorang & Yahfizham [8], juga membuktikan bahwa *ensemble learning* dengan *boosting*, *bagging*, dan *stacking* lebih unggul daripada *single classifier*. Lalu, untuk mengatasi ketidakseimbangan data, dimana data transaksi normal lebih banyak daripada transaksi *fraud* sehingga dapat mengurangi performa model, digunakan metode *random oversampling*. Kinerja model akan dievaluasi dan dibandingkan menggunakan *confusion matrix* dan ROC-AUC *Score* dengan algoritma *Decision Tree* (DT), *Random Forest* (RF), *Logistic Regression* (LR), dan *Naive Bayes* (NB). Sehingga penelitian ini diberi judul "Deteksi Anomali Menggunakan *Ensemble Learning* dan *Random Oversampling* Pada Penipuan Transaksi Keuangan" dan akan membandingkan metode *Ensemble Learning* dengan *Single Classifier* menggunakan data transaksi penipuan kartu kredit.

2. TINJAUAN PUSTAKA

2.1 Penipuan Transaksi

Penipuan transaksi keuangan adalah upaya untuk mendapatkan keuntungan finansial atau menyebabkan kerugian dengan trik implisit maupun eksplisit[9]. Penipuan ini dapat terjadi melalui metode seperti *phishing*, *impersonate*, *vishing*, dan *smishing* [10].

2.2 Deteksi Anomali

Deteksi anomali adalah proses mencari data dengan perilaku yang berbeda dari biasanya [5]. Anomali dihasilkan oleh aktivitas abnormal seperti serangan dunia maya dan penipuan kartu kredit.

2.3 *Ensemble Learning*

Ensemble learning merupakan kerangka konseptual dalam *machine learning* yang menggabungkan lebih dari satu jenis algoritma atau model untuk membuat prediksi, dengan mengambil bobot prediksi dari masing-masing algoritma [7]. Dalam penelitian ini, digunakan metode *bagging*, *boosting*, dan *stacking*.

2.3.1 *Bagging*

Bagging melibatkan penggunaan subset *data training* secara acak untuk melatih algoritma, dan hasilnya digabungkan melalui suara terbanyak atau perhitungan rata-rata, meningkatkan akurasi dan mengurangi *overfitting* [11].

2.3.2 *Boosting*

Boosting bekerja dengan membangun model secara berurutan dan menyesuaikan bobot *data training* berdasarkan kesalahan yang terjadi, sehingga performa keseluruhan meningkat dengan mengurangi bias [11].

2.3.3 *Stacking*

stacking menggabungkan luaran dari beberapa model untuk menciptakan model baru yang lebih akurat dan handal dalam mendeteksi anomali, karena memanfaatkan kelebihan dari berbagai model yang berbeda [11].

2.4 *Base Learner*

Base learner, atau *single classifier*, adalah model pembelajaran *machine learning*

yang berfungsi sebagai komponen dasar untuk mendefinisikan satu atau lebih estimator dari fungsi prediksi [12]. *Logistic Regression*, *Decision Tree, Naïve Bayes,* dan *Random Forest* adalah model *classifier* yang digunakan dalam penelitian ini.

2.4.1 *Logistic Regression*

Logistic Regression adalah model komputasi yang menentukan pengaruh variabel independen pada *binary variable dependent*, dengan data berkode 0 dan 1 [13]. *Logistic Regression* dapat dinotasikan pada persamaan berikut.

$$\ln\left(\frac{p}{1-p}\right) = B_0 + B_1 X \quad (2.1)$$

ln = Logaritma natural
B_0 = Konstanta
B_1 = Koefisien masing-masing variable
X = Variabel independent
P= Probabilitas logistik yang dirumuskan sebagai berikut:

$$p = \frac{\exp(B_0 + B_1 x)}{1 + \exp(B_0 + B_1 x)} = \frac{e^{B_0 + B_1 X}}{1 + e^{B_0 + B_1 X}} \quad (2.2)$$

Exp atau e = fungsi exponent

2.4.2 *Decision Tree*

Algoritma Decision Tree (DT) populer untuk klasifikasi dan prediksi, merepresentasikan fakta dalam bentuk pohon keputusan yang membagi data besar menjadi bagian lebih kecil [14]. Menentukan akar pada *Decision Tree* dapat dilakukan dengan mencari nilai gain tertinggi. Berikut merupakan rumus perhitungan gain.

$$Gain\ (S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} \times Entropy(S_i) \quad (2.3)$$

Keterangan:
S : Himpunan kasus
A : *Variable*
n : Total partisi variabel A
|Si| : Total kasus dalam partisi ke-i
|S| : Total kasus pada S

Pada perhitungan gain juga terdapat *entropy*. *Entropy* sendiri merupakan nilai ketidakmurnian pada suatu ciri. Dibawah ini merupakan rumus untuk perhitungan *entropy*.

$$Entropy(S) = \sum_{i=1}^{n} pi \times log_2 pi \quad (2.4)$$

Keterangan:
S : Himpunan kasus
n : Total partisi S
Pi : Proporsi dari Si terhadap S

Ciri dengan nilai *gain* tertinggi akan menjadi akar dari *Decision Tree* yang akan dibuat. Perhitungan *Decision Tree* akan terhenti ketika seluruh *record* pada node (*n*) telah memperoleh kelas yang sama dan tidak terdapat ciri pada *record* yang dipartisi kembali.

2.4.3 *Naïve Bayes Classifier*

Naive Bayes adalah metode klasifikasi yang berbasis pada *teorema Bayes* dan mengasumsikan independensi antara fitur-fitur (daun) yang diamati[15]. *Naive Bayes* dapat dinotasikan pada persamaan berikut,

$$P(y|X) = \frac{p(X|y).p(y)}{p(x)} \quad (2.5)$$

Keterangan:
P(y|X) : Nilai probabilitas dari hipotesis y yang terjadi jika ada bukti X.
P(X|y) : Nilai probabilitas dari bukti X yang berdampak pada hipotesis y.
P(y) : Nilai probabilitas awal dari hipotesis y yang tidak dipengaruhi oleh bukti X.
P(X) : Nilai probabilitas awal bukti X yang tidak dipengaruhi hipotesis y.

2.4.4 *Random Forest*

Random Forest, yaitu metode yang bekerja dengan menciptakan banyak pohon keputusan secara acak, dengan prediksi akhir berdasarkan "suara" terbanyak [16].

$$Entropy(S) = \sum_{i=1}^{n} pi \times log_2 pi \quad (2.6)$$

Keterangan:
S : Himpunan kasus
n : Total partisi S
Pi : Proposi dari Si terhadap S

Rumus *information gain* dapat dilihat pada persamaan dibawah ini:

$$Entropy(S) = \sum_{i=1}^{n} pi \times log_2 pi \quad (2.7)$$

Keterangan:

S : Himpunan kasus
n : Total partisi S
Pi : Proporsi dari Si terhadap S

2.5 *Random Oversampling*

Dalam penelitian ini, digunakan metode *random oversampling*, yaitu teknik yang mengulang sampel dari kelas minoritas secara acak hingga jumlahnya setara dengan kelas mayoritas, [17].

2.6 Pengukuran Performa

Penelitian ini menggunakan lima pengukuran utama untuk menilai performa model klasifikasi, yaitu dengan nilai *accuracy*, *Precision*, *Recall*, *F1-score*, dan AUC-ROC *score*, yang semuanya dapat diperoleh dari *Confusion Matrix* [18].

Tabel 3. 1 Confusion Matrix

Confusion Matrix		*Kelas Prediksi*	
		Positif	*Negatif*
Kelas Sebenarnya	*Positif*	*TP*	*FN*
	Negatif	*FP*	*TN*

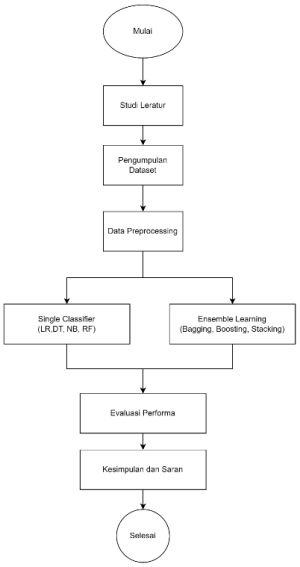
Keterangan:

a. *True Positive* (TP), yaitu data positif yang diprediksi benar.
b. *True Negative* (TN), yaitu data negatif yang diprediksi benar.
c. *False Positive* (FP), yaitu data negatif yang diprediksi positif.
d. *False Negative* (FN), yaitu data positif yang diprediksi negatif.

Nilai *accuracy*, *precision*, *recall*, dan *F1-score* dihitung berdasarkan nilai-nilai ini. ROC (*Receiver Operating Characteristics*) *curve*, digunakan untuk menampilkan performa klasifikasi, menggambarkan *confusion matrix* dalam grafik dua dimensi dengan *false positive* pada sumbu horizontal dan *true positive* pada sumbu vertikal [19].

3. METODE PENELITIAN

Penelitian ini terdiri dari beberapa tahapan. Gambar 1 menunjukkan alur dari penelitian ini. Tahapan pertama adalah studi literatur untuk mengumpulkan teori landasan yang mendukung penelitian, lalu pengumpulan data penipuan transaksi keuangan. Selanjutnya dilakukan preprocessing untuk membersihkan dan menyeragamkan bentuk data. Setelah itu, data siap untuk diuji pada model *classifier* dengan metode *ensemble learning* dan *base learning*. Setelah pengujian model selesai, dilakukan evaluasi performa model klasifikasi.



Gambar 3. 1 Alur Penelitian

3.1 Studi Literatur

Tahapan studi literatur bertujuan mengumpulkan pengetahuan untuk mengatasi masalah penelitian dengan mengumpulkan sumber dari buku, jurnal, dan situs terkait deteksi anomali, data mining, algoritma base learner, dan ensemble learning. Ini membantu peneliti membandingkan metode dan menerapkan teori dalam model penelitian.

3.2 Pengumpulan Data

Penelitian ini menggunakan dataset transaksi keuangan sintetis dari PaySim, diunduh dari Kaggle.

3.3 *Data Preprocessing*

a. Exploratory Data Analysis (EDA)

Tahapan EDA bertujuan untuk memahami karakteristik data. Tahapan ini dilakukan dengan menganalisis korelasi dengan menggunakan *visualiasi* dan *heatmap correlation.*

b. Data Cleaning

Tahapan *data cleaning* ini dilakukan untuk membersihkan data *missing value* atau *noise*. Proses ini dilakukan untuk untuk menghapus baris yang mengandung *missing value* dapat menggunakan metode '*dropna*()'.

c. Data Selection

Tahapan *data selection* ini dilakukan untuk memilih atribut yang relevan dan menghapus data yang tidak digunakan dalam penilitan.

d. Data Transformation

Tahapan ini dilakukan untuk mengubah data ke dalam bentuk yang sesuai, dengan cara melakukan suatu penjumlahan atau agregasi. Setelah melalui *mapping* selanjutnya dilakukan perubahan format kategori menjadi numerik menggunkan metode *one-hot encoding.*

e. Splitting Data

Langkah selanjutnya adalah membagi data menjadi dua bagian yaitu sebagai data *training* dan data *testing*. Data *training* ini digunakan untuk melatih model sedangkan data *testing* digunakan untuk memvalidasi model yang dibangun.

3.4 *Resampling*

Proses *random oversampling* dilakukan dengan menambah data dari kelas minoritas ke dalam *data training* secara acak. Proses penambahan ini berulang hingga jumlah data kelas minoritas sama dengan jumlah kelas mayoritas. Dengan menerapkan teknik *random oversampling* (*ROS*), dapat meminimalisir ketidak seimbangan kelas.

3.5 Implementasi Model

Dalam penelitian ini, metode yang digunakan adalah model *base learning Decision Tree (DT), Logistic Regression (LR), Naïve Bayes (NB), dan Random Forest (RF)* dan *ensemble learning (bagging, boosting,* dan *stacking).* Implementasi model dibagi menjadi 2 skenario yaitu, skenario 1 dan skenario 2. Selanjutnya, setelah pengujian dilakukan, akan dilakukan evaluasi terhadap performa masing-masing model dan menganalisis hasilnya. Kemudian akan dilakukan perbandingan antara nilai-nilai yang sudah diperolah dari kedua skenario.

3.6 Evaluasi Model

Output dari tahap implementasi skenario 1 dan skenario 2 masing masing akan dibandingkan performanya berdasarkan nilai *accuracy, precisión, recall. F1-score* menggunakan confusión matrix dan *ROC-AUC Score*. Setelah hasil dari semua model skenario dibuat, selanjutnya dapat disimpulkan dalam penelitian ini skenario mana yang memiliki performa lebih baik.

3.7 Analisa Hasil

Tahapan terakhir, yaitu menganalisis hasil dari kinerja dari 2 metode yaitu *base learning* dan *ensemble learning*, evaluasi dilakukan menggunakan *confusion matrix* dan *ROC-AUC score* untuk mendapatkan keunggulan dan kelemahan masing-masing model dalam deteksi anomali.

4. HASIL DAN PEMBAHASAN

4.1 Pengumpulan Data

Pada penelitian ini data yang digunakan, yaitu dataset PaySim yang merupakan dataset transaksi keuangan elektronik yang dibuat secara sintetis berisi total 6.362.620 transaksi dengan 8.213 transaksi *fraud*. Dataset ini terdiri dari 11 fitur dan 6.362.620 baris dengan tipe fitur yang berisi variabel numerik dan kategori. Pada proses pengumpulan data peneliti menggunakan dataset yang diunduh melalui kaggle dan disimpan ke file dalam format *csv*.

	step	type	amount	nameOrig	oldbalanceOrg	newbalanceOrig	nameDest	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud
0	1	PAYMENT	9839.64	C1231006815	170136.0	160296.36	M1979787155	0.0	0.0	0	0
1	1	PAYMENT	1864.28	C1666544295	21249.0	19384.72	M2044282225	0.0	0.0	0	0
2	1	TRANSFER	181.00	C1305486145	181.0	0.00	C553264065	0.0	0.0	1	0
3	1	CASH_OUT	181.00	C840083671	181.0	0.00	C38997010	21182.0	0.0	1	0
4	1	PAYMENT	11668.14	C2048537720	41554.0	29885.86	M1230701703	0.0	0.0	0	0

Gambar 4. 1 Dataset Paysim

4.2 Preprocessing Data

Pada tahapan ini bertujuan untuk memastikan bahwa data yang akan digunakan telah dipersiapkan untuk diproses oleh algoritma. Melalui serangkaian langkah, dataset disesuaikan agar memenuhi kebutuhan analisis lebih lanjut dan mengoptimalkan kinerja model. Tahapan ini memastikan bahwa data yang digunakan dalam analisis telah disiapkan secara optimal untuk proses selanjutnya.

4.2.1 Exploratory Data Analysis

Pada tahap ini dilakukan eksplorasi data untuk mengetahui informasi dari dataset sehingga dapat melakukan langkah-langkah persiapan data, pemilihan fitur, serta memahami karakteristik data yang digunakan.

a. Menampilkan Informasi Dataset

Proses eksplorasi data ini dilakukan untuk mengetahui informasi mengenai dataset yang digunakan sehingga dapat melakukan langkah-langkah persiapan data, pemilihan fitur, serta memahami karakteristik data yang digunakan. Berikut adalah infromasi dari dataset.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6362620 entries, 0 to 6362619
Data columns (total 11 columns):
 #   Column          Dtype
---  ------          -----
 0   step            int64
 1   type            object
 2   amount          float64
 3   nameOrig        object
 4   oldbalanceOrg   float64
 5   newbalanceOrig  float64
 6   nameDest        object
 7   oldbalanceDest  float64
 8   newbalanceDest  float64
 9   isFraud         int64
 10  isFlaggedFraud  int64
dtypes: float64(5), int64(3), object(3)
memory usage: 534.0+ MB
```

Gambar 4. 2 Informasi tipe data pada Dataset

Informasi yang diperoleh dari fungsi df.info menunjukan bahwa dataset memiliki 6,362,620 baris dan 11 kolom dan terdiri dari tipe data berupa variable numerik dan kategori yang memungkinkan untuk melalui tahap *data transformation* menggunakan *one hot encoding*

b. Menampilkan Ringkasan Dataset

Tahap selanjutnya untuk mengetahui ringkasan statistik deskriptif dari data digunakan berikut adalah ringkasan dari statistik dataset.

```
df.describe()
```

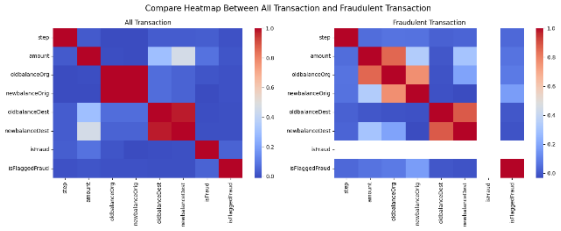
	step	amount	oldbalanceOrg	newbalanceOrig	oldbalanceDest	newbalanceDest	isFraud	isFlaggedFraud
count	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06	6.362620e+06
mean	2.433972e+02	1.798619e+05	8.338831e+05	8.551137e+05	1.100702e+06	1.224996e+06	1.290820e-03	2.514687e-06
std	1.423320e+02	6.038582e+05	2.888243e+06	2.924049e+06	3.399180e+06	3.674129e+06	3.590480e-02	1.585775e-03
min	1.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
25%	1.560000e+02	1.338957e+04	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00	0.000000e+00
50%	2.390000e+02	7.487194e+04	1.420800e+04	0.000000e+00	1.327057e+05	2.146614e+05	0.000000e+00	0.000000e+00
75%	3.350000e+02	2.087215e+05	1.073152e+05	1.442584e+05	9.430367e+05	1.111909e+06	0.000000e+00	0.000000e+00
max	7.430000e+02	9.244552e+07	5.958504e+07	4.958504e+07	3.560159e+08	3.561793e+08	1.000000e+00	1.000000e+00

Gambar 4. 3 Statistik Dataset

Ringkasan statistik diatas memberikan Gambaran tentang distribusi data dalam dataset. Analisis ini mencakup statistik seperti count (jumlah baris pada kolom), mean (nilai rata-rata), std (standar deviasi), min (nilai minimal pada kolom), max (nilai maksimal pada kolom), serta nilai percentil 25%, 50%, dan 75%. Hasil analisis menunjukkan bahwa pada dataset, terdapat korelasi antara kolom "oldbalanceDest" dan "newbalanceDest", serta antara "oldbalanceOrig" dan "newbalanceOrig". Korelasi ini menunjukkan bahwa salah satu kolom dari pasangan yang berkorelasi dapat dihilangkan tanpa kehilangan informasi penting, sehingga dapat mengurangi dimensi data dan kompleksitas model tanpa mengorbankan kualitas prediksi.

c. Heatmap Korelasi

Pada tahap ini dilakukan heatmap korelasi menggunakan visualiasi korelasi heatmap. Berikut ini adalah *visualisasi* dari *heatmap correlation.*



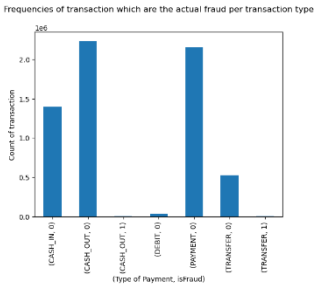
Gambar 4. 4 Heatmap Korelasi

Untuk mengukur korelasi antar variabel, digunakan heatmap korelasi dengan metode *Pearson*, yang mengukur hubungan linear antara dua variabel dengan koefisien korelasi berkisar antara -1 hingga 1. Hasil visualisasi ditampilkan pada Gambar 4.4, dengan warna merah menunjukkan korelasi positif yang kuat dan warna biru menunjukkan korelasi negatif yang kuat. Analisis menunjukkan bahwa transaksi curang memiliki korelasi kuat antara variabel jumlah transaksi (*amount*), saldo awal (*oldbalanceOrig*), dan saldo akhir (*newbalanceDest*). Dalam transaksi secara umum, terdapat pola korelasi lebih luas, sedangkan dalam transaksi curang, pola korelasi lebih spesifik. Identifikasi transaksi curang dapat difokuskan pada variabel seperti 'amount', 'oldbalanceDest', dan 'newbalanceOrig' karena menunjukkan pola korelasi kuat dalam transaksi curang.

d. Visualiasi Data Transaksi Penipuan

Langkah selanjutnya, dilakukan analisis frekuensi transaksi penipuan per jenis

transaksaksi dari hasil visualisasi dari masing-masing jenis transaksi menggunakan source code berikut.



Gambar 4. 5 Transaksi Penipuan Per Jenis Transaksi

Gambar 4.5 menunjukkan frekuensi transaksi penipuan berdasarkan jenis transaksi. Jenis transaksi CASH_OUT dan TRANSFER adalah yang paling rentan terhadap penipuan, dengan jumlah transaksi penipuan yang lebih tinggi dibandingkan jenis lainnya. Sebaliknya, jenis transaksi CASH_IN dan PAYMENT tidak menunjukkan adanya penipuan, menunjukkan keamanan yang lebih tinggi. Berdasarkan analisis ini, langkah pencegahan penipuan harus difokuskan pada transaksi CASH_OUT dan TRANSFER untuk mengurangi risiko penipuan.

4.2.2 Data Cleaning

Pada proses ini dilakukan pengecekan terhadap *missing value* atau nilai kosong pada dataset. Berikut adalah hasil dari proses pengecekan missing value.

```
Missing Values Check:
step              0
type              0
amount            0
nameOrig          0
oldbalanceOrg     0
newbalanceOrig    0
nameDest          0
oldbalanceDest    0
newbalanceDest    0
isFraud           0
isFlaggedFraud    0
dtype: int64
```

Gambar 4. 6 Pengecekan Missing Value

Gambar diatas menunjukan hasil dari pengecekan *missing value* bahwa tidak ada nilai kosong di setiap kolom dataset sehingga tidak perlu dilakukan cleaning pada dataset.

4.2.3 Data Selection

Pada tahap ini dilakukan seleksi fitur, Identifikasi transaksi curang dapat difokuskan pada variabel-variabel seperti '*amount'*, '*oldbalanceDest'*, dan '*newbalanceOrig'* karena menunjukkan pola korelasi yang kuat dalam transaksi curang sehingga dilakukan seleksi fitur yang tidak digunakan.

4.2.4 Data Transformation

Tahapan transformasi data ini perlu dilakukan karena dataset mengandung variabel kategorikal yang perlu dikonversi menjadi variabel numerik

Out[27]:

	amount	newbalanceOrig	oldbalanceDest	isFraud	type_CASH_IN	type_CASH_OUT	type_DEBI
0	9839.64	160296.36	0.0	0	0.0	0.0	0.(
1	1864.28	19384.72	0.0	0	0.0	0.0	0.(
2	181.00	0.00	0.0	1	0.0	0.0	0.(
3	181.00	0.00	21182.0	1	0.0	1.0	0.(
4	11668.14	29885.86	0.0	0	0.0	0.0	0.(

Gambar 4. 7 Hasil Transformasi Data

Dapat dilihat pada Gambar 4.7, variabel kategorikal seperti *'type'* telah diubah menjadi variabel numerik melalui proses *One Hot Encoding*. Variabel-variabel ini kemudian ditampilkan dalam format biner (*True*/*False*) setelah transformasi selesai dilakukan

4.2.5 Splitting Data

Proses splitting data dilakukan, untuk membagi fitur (X) dan target (y) dataset menjadi data pelatihan dan data uji. Kolom x akan berisi semua kolom kecuali kolom '*isFraud'*, sedangkan kolom y berisikan nilai dari kolom '*isFraud*' sebagai target atau label. Setelah data dibagi menjadi fitur dan target, data tersebut kemuidian dibagi lagi menjadi 70% untuk data pelatihan dan 30% untuk data uji.

4.3 Resampling

Sebelum memasuki proses pengujian model dilakukan proses *resampling* dengan menggunakan metode *RandomOverSampling* untuk menangani ketidakseimbangan data pelatihan menggunakan pemanfaatan *library imbalanced*-learn dengan teknik *RandomOverSampler*. *RandomOverSampler* akan diterapkan kepada pelatihan, kemudian disimpan. Sehingga menghasilkan output sebagai berikut.

```
Addressing Class Imbalance with RandomOverSampler:
After RandomOverSampler Label Distribution:
isFraud
0    4448085
1    4448085
Name: count, dtype: int64
```

Gambar 4. 8 Hasil Resampling Dataset

4.4 Implementasi Model

Setelah tahap *preprocessing data*, *dataset* yang sudah lebih bersih dan seragam siap untuk diimplementasikan kedalam model dengan menggunakan 2 skenario, yaitu menggunakan *base learning* dan *ensemble learning.*

4.4.1 Implementasi Model Skenario 1

Pada implementasi skenario 1 dimana keempat base learner diujikan tanpa menggunakan metode ensemble.

4.4.2 Implementasi Model Skenario 2

Proses pada skenario 2 tidak memiliki perbedaan yang cukup jauh dari skenario 1. Perbedaannya hanya terletak pada pendefinisian dan pemanggilan model yang digunakan adalah model *Ensemble* yang telah diimplementasikan kepada algoritma *base learner* pada skenario 1.

4.5 Evaluasi Model

Hasil yang didadaptkan dari eksperimen yang dilakukan pada masing-masing skenario berupa nilai *Accuracy, Precision, Recall, F1 Score,* dan *AUC ROC Score* dari masing- masing model algortima yang telah dibuat. Berikut adalah rincian nilai performa dari masing-masing skenario.

4.5.1 Skenario 1

Pada implementasi skenario 1 dimana keempat *base learner* diujikan tanpa menggunakan metode *ensemble*, algoritma Dari hasil evaluasi, dapat dilihat pada Tabel 4.1 bahwa *Random Forest dan Decision Tree* menunjukkan kinerja yang sangat tinggi dalam hal akurasi, presisi, *recall*, dan *F1 Score* pada dataset. Ini menandakan bahwa *Random Forest* dan *Decision Tree* adalah model yang kuat dan konsisten dalam melakukan prediksi tersebut dalam skenario di mana metode *ensemble* tidak digunakan.

Tabel 4. 1 Hasil Skenario 1

Classifier	***Accuracy***	**Precission**	**Recall**	**F1 Score**	**ROC AUC**
DT	99.99%	99.58%	99.50%	99.54%	99.75%
LR	44.38%	0.21%	92.27%	0.42%	84.88%
NB	97.57%	1.69%	31.70%	3.22%	79.84%
RF	99.99%	99.75%	99.50%	99.63%	99.79%

4.5.2 Skenario 2

Pada skenario 2, tiga metode *ensemble* diimplementasikan pada keempat base *learner*. Dari total sebanyak 12 pengujian model dengan dataset model yang menggunakan *stacking* menjadi model paling unggul, yaitu *random forest stacking* menunjukan nilai confusion matrix dengan kesalahan yang sangat sedikit. Secara keseluruhan, metode *ensemble Stacking* menunjukkan kinerja yang sangat baik dan cenderung meningkatkan performa model dasar secara signifikan. Model *Stacking* memberikan peningkatan yang signifikan dalam hal *precision*, *recall*, dan *F1 score* dibandingkan dengan metode *Bagging* dan *Boosting*, terutama untuk *Logistic Regression* dan *Naive Bayes*. Model *ensemble Stacking* secara konsisten menunjukkan hasil yang superior dalam skenario ini. Berikut adalah hasil keseluruhan pada skenario 2 pada Tabel 4.2

Tabel 4. 2 Hasil Skenario 2

Classifier	*Accuracy*	Precission	Recall	F1 Score	ROC AUC
DT	Bagging	99.99%	99.50%	99.50%	99.50%
	Boosting	99.99%	99.58%	99.50%	99.54%
	Stacking	99.99%	99.79%	99.50%	99.65%
LR	Bagging	44.34%	0.21%	92.27%	0.42%
	Boosting	47.62%	0.22%	91.08%	0.44%
	Stacking	99.99%	99.79%	99.50%	99.65%
NB	Bagging	97.56%	1.69%	31.74%	3.21%
	Boosting	12.59%	0.14%	97.00%	0.28%
	Stacking	99.99%	99.75%	99.50%	99.65%
RF	Bagging	99.99%	99.67%	99.54%	99.60%
	Boosting	99.99%	99.75%	99.50%	99.62%
	Stacking	99.99%	99.75%	99.50%	99.62%

4.6 Analisa Hasil

Dari keseluruhan model yang diujikan pada skenario 1 dan skenario 2 didapatkan bahwa metode *Stacking* memiliki nilai paling tinggi dengan berhasi unggul dari ketiga model *ensemble learning*. Sehingga dapat menunjukkan bahwa metode *ensemble stacking* dan menggunakan *random oversampling* sebagai resampling sangat efektif dalam melakukan deteksi penipuan.

5. KESIMPULAN

Berdasarkan penelitian yang dilakukan menggunakan metode *ensemble learning* dan

random oversampling untuk deteksi anomali transaksi keuangan, beberapa kesimpulan yang diperoleh adalah sebagai berikut:

1. Penerapan metode *ensemble learning* dan *random oversampling* pada deteksi transaksi keuangan menggunakan model *Logistic Regression*, *Decision Tree*, *Random Forest*, dan *Naïve Bayes* dengan teknik bagging, *boosting*, dan *stacking*. Teknik *stacking* secara khusus menunjukkan peningkatan *AUC* yang signifikan, dengan beberapa algoritma mencapai AUC sempurna 1.00. *Random Forest* (RF) dengan metode *ensemble learning* menunjukkan performa yang sangat konsisten dan optimal dalam mendeteksi anomali penipuan.
2. Secara keseluruhan, penggunaan metode *ensemble learning*, terutama teknik *stacking*, menghasilkan tingkat akurasi tinggi dalam mendeteksi anomali penipuan pada transaksi keuangan. Perbandingan antara metode *ensemble learning* dan *base learning* menunjukkan bahwa teknik ensemble seperti *bagging*, *boosting*, dan *stacking* cenderung meningkatkan performa deteksi, dengan *stacking* sering kali memberikan hasil terbaik. *Random Forest* (RF) menunjukkan hasil yang paling konsisten dan optimal di kedua skenario dan dataset dengan penerapan metode ensemble learning, menjadikannya model yang sangat efektif untuk deteksi penipuan.

UCAPAN TERIMA KASIH

Saya mengucapkan terima kasih yang sebesar-besarnya atas bimbingan dan dukungan yang diberikan dalam penulisan jurnal ini. Semua kontribusi, baik langsung maupun tidak langsung, sangat berarti dalam proses penelitian ini. Terima kasih atas panduan, saran, dan dorongan yang telah diberikan.

DAFTAR PUSTAKA

[1] E. M. Ginting, E. S. Siburian, M. D. Syahfitri, and H. Hasyim, "Analisis Perilaku Konsumen dan Keamanan Kartu Kredit Perbankan," *Madani: Jurnal Ilmiah Multidisiplin*, vol. 1, no. 4, 2023.

[2] M. Giswandhani and A. Z. Hilmi, "Pengaruh kemudahan transaksi non-tunai terhadap sikap konsumtif masyarakat kota makassar," *Kareba: Jurnal ilmu komunikasi*, pp. 239–250, 2020.

[3] S. Wienanto and Z. Wuragil, "Kata BNI Soal Penipuan Kartu Kredit yang Rugikan 20 Nasabahnya Rp 1 Miliar," tempo.co.

[4] A. E. Wardoyo, "Deteksi Penipuan Kartu Kredit Menggunakan Algoritma Memetika Dan Pencarian Tersebar," *JUSTINDO (Jurnal Sistem dan Teknologi Informasi Indonesia)*, vol. 8, no. 2, pp. 87–98, 2023.

[5] M. Ahadi, "Undersampling Majority Class pada Kasus Imbalanced Dataset dan Aplikasinya Pada Deteksi Anomali Transaksi Kartu Kredit," 2019.

[6] L. M. Cendani and A. Wibowo, "Perbandingan Metode ensemble learning pada klasifikasi penyakit diabetes," *Jurnal Masyarakat Informatika*, vol. 13, no. 1, pp. 33–44, 2022.

[7] R. Sudiyarno, A. Setyanto, and E. T. Luthfi, "Peningkatan Performa Pendeteksian Anomali Menggunakan Ensemble Learning dan Feature Selection," *Creative Information Technology Journal*, vol. 7, no. 1, pp. 1–9, 2021.

[8] S. Situmorang and Y. Yahfizham, "Analisis Kinerja Algoritma Machine Learning Dalam Deteksi Anomali Jaringan," *Konstanta: Jurnal Matematika Dan Ilmu Pengetahuan Alam*, vol. 1, no. 4, pp. 258–269, 2023.

[9] F. Zamachsari and N. Puspitasari, "Penerapan Deep Learning dalam Deteksi Penipuan Transaksi Keuangan Secara Elektronik," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 2, pp. 203–212, 2021.

[10] E. P. Sari, D. A. Febrianti, and R. H. Fauziah, "Fenomena Penipuan Transaksi Jual Beli Online Melalui Media Baru Berdasarkan Kajian Space Transition Theory," *Deviance Jurnal Kriminologi*, vol. 6, no. 2, pp. 153–168, 2022.

[11] B. Torky, "Ensemble methods for the anomaly detection in enterprise systems," 2023.

[12] R. V Phillips, M. J. Van Der Laan, H. Lee, and S. Gruber, "Practical considerations for specifying a super learner," *Int J Epidemiol*, vol. 52, no. 4, pp. 1276–1285, 2023.

[13] M. Alfyando, F. T. Anggraeny, and A. N. Sihananto, "Perbandingan Algoritma Random Forest dan Logistic Regression Untuk Analisis Sentimen Ulasan Aplikasi Tumbuh Kembang Anak Di Play Store," *Jurnal Sistem Informasi dan Ilmu Komputer*, vol. 2, no. 1, pp. 77–86, 2024.

[14] D. Septhya *et al.*, "Implementasi Algoritma Decision Tree dan Support Vector Machine untuk Klasifikasi Penyakit Kanker Paru:

Implementation of Decision Tree Algorithm and Support Vector Machine for Lung Cancer Classification," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 1, pp. 15–19, 2023.

[15] R. R. Burhanuddin, "KLASIFIKASI PENYAKIT PADI MELALUI CITRA DAUN MENGGUNAKAN METODE NAIVE BAYES," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 2, 2024.

[16] N. L. P. C. Savitri, R. A. Rahman, R. Venyutzky, and N. A. Rakhmawati, "Analisis klasifikasi sentimen terhadap sekolah daring pada twitter menggunakan Supervised Machine Learning," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 7, no. 1, 2021.

[17] R. Aryanti, T. Misriati, and R. Hidayat, "Klasifikasi Risiko Kesehatan Ibu Hamil Menggunakan Random Oversampling Untuk Mengatasi Ketidakseimbangan Data," *KLIK: Kajian Ilmiah Informatika dan Komputer*, vol. 3, no. 5, pp. 409–416, 2023.

[18] F. N. Hermawan, "Deteksi anomali pada data internet of things menggunakan model ensemble learning.," UIN Syarif Hidayatullah Jakarta, 2021.

[19] D. Y. Utami, E. Nurlelah, and F. N. Hasan, "Comparison of Neural Network Algorithms, Naive Bayes and Logistic Regression to predict diabetes," *Journal of Informatics and Telecommunication Engineering*, vol. 5, no. 1, pp. 53–64, 2021.