



Klasifikasi Transaksi Penipuan Pada Kartu Kredit Menggunakan Metode *Resampling* Dan Pembelajaran Mesin

Mukhlis Febriady, Samsuryadi*, Dian Palupi Rini

Fakultas Ilmu Komputer, Magister Ilmu Komputer, Universitas Sriwijaya, Palembang, Indonesia

Email: ¹mukhlisfebriady@gmail.com, ^{2,*}samsuryadi@unsri.ac.id, ³dprini@unsri.ac.id

Email Penulis Korespondensi: samsuryadi@unsri.ac.id

Abstrak—Tingginya angka penipuan kartu kredit menyebabkan banyak kerugian baik dari sisi pengguna maupun penyedia layanan kartu kredit. Karena laju transaksi kartu kredit sangat cepat maka, perlu dilakukan deteksi penipuan kartu kredit sedini mungkin. Namun tantangan lain yang tidak kalah penting adalah jumlah data yang tidak seimbang (*imbalance data*) antara transaksi valid dan tidak valid. Salah satu solusi untuk permasalahan *imbalance data* adalah menggunakan metode *resampling* yang dapat memperbaiki kuantitas data sehingga hasil akurasi semakin baik. Pada penelitian ini tiga jenis metode *resampling*, SMOTE, *bootstrap*, dan *Jackknife* diimplementasikan. Selanjutnya untuk memvalidasi keberhasilan metode *resampling* tiga jenis metode pembelajaran mesin digunakan. Metode pembelajaran mesin tersebut adalah SVM, ANN, dan *Random Forest*. Dari hasil pengujian didapatkan kombinasi metode *resampling* SMOTE dan *random forest* menghasilkan kinerja terbaik dengan nilai akurasi, presisi, recall dan F1-score masing-masing sebesar 99.95%, 81.63%, 90.91% dan 86.02%.

Kata Kunci: *Imbalance Data*; Metode *Resampling*; Penipuan Kartu Kredit; Pembelajaran Mesin; Kartu Kredit.

Abstract—The high number of credit card fraud causes a lot of losses for both users and credit service providers. Because the rate of credit card transactions is very fast, it is necessary to detect credit card fraud as early as possible. However, another challenge that is no less important is the amount of data that is imbalanced between valid and invalid transactions. One solution to the problem of data imbalance is to use a resampling method that can improve the quantity of data so that the accuracy results are good. In this study, three types of resampling methods were implemented, SMOTE, bootstrap, and jackknife. Furthermore, to validate the success of the *resampling* method, three types of machine learning methods were used. The machine learning methods are SVM, ANN, and *random forest*. From the test results, it was found that the combination of *resampling* SMOTE and *random forest* methods produced the best performance with values of accuracy, precision, recall and F1-score of 99.95%, 81.63%, 90.91% and 86.02%, respectively.

Keywords: Imbalance Data; *Resampling* Method; Credit Card Fraud; Machine Learning; Credit Card.

1. PENDAHULUAN

Kemudahan dalam mengakses informasi dapat menyebabkan penyalahgunaan informasi terutama pada tindak kejahatan mulai dari penipuan, penyerangan dan berbagai kasus lainnya [1], [2]. Salah satu bentuk manipulasi data pada bidang e-commerce adalah penipuan pada kartu kredit [3]. Transaksi kartu kredit adalah metode pembayaran yang paling umum dalam beberapa tahun terakhir. Meskipun kartu kredit bukanlah alat pembayaran yang paling diminati di Indonesia. Namun jumlah masyarakat pengguna kartu kredit di Indonesia tidaklah sedikit. Bank Indonesia (BI) mencatat pada bulan Januari 2013 jumlah pemegang kartu kredit telah mencapai 14.591.371. Pada awal tahun 2013 ini, nilai transaksinya mencapai Rp 17,96 triliun [4].

Tingginya angka penipuan kartu kredit menyebabkan banyaknya kasus yang terjadi dan sulit untuk dilakukan pencegahan. Sehingga perlu dilakukan deteksi penipuan kartu kredit sedini mungkin. Deteksi penipuan adalah proses mendeteksi apakah suatu transaksi sah atau tidak [5]. Sistem deteksi penipuan otomatis diperlukan terutama mengingat lalu lintas data transaksi yang besar, dan tidak mungkin bagi manusia untuk memeriksa secara manual setiap transaksi satu per satu apakah itu curang atau tidak. Sistem deteksi penipuan otomatis dengan menggunakan teknik pembelajaran mesin dan metode lain sudah banyak berkembang [3].

Beberapa metode yang telah dikembangkan untuk mendeteksi transaksi kartu kredit sah atau tidak telah dimulai sejak lama, salah satu penelitian menggunakan metode pembelajaran mesin yang dilakukan pada tahun 2019 oleh [5]. Penelitian ini menggunakan algoritma machine learning *random forest* sebagai model klasifikasinya. Metode *random forest* menerapkan konsep *bootstrap* sehingga dapat menghindari terjadinya *overfitting*. Selanjutnya [6] mengimplementasikan metode *Artificial Neural Network* (ANN) pada kasus deteksi penipuan kartu kredit. Metode ANN memiliki kelebihan dalam mempelajari pola dari data untuk memperkirakan hubungan nonlinier antara informasi input dan output akhir. Selain itu ANN memiliki kemampuan untuk menangkap fitur abstrak di seluruh kumpulan data. Demi mendapat hasil akurasi terbaik dalam pembangunan model pendeteksi transaksi pada kartu kredit, penelitian-penelitian sejenis terus dilakukan.

Beberapa metode yang kembali dikembangkan, diantaranya dengan menggunakan metode machine learning [7] [8], metode *pipelining and ensemble learning* [9], dengan menggunakan metode *Graph Based Anomaly* (GBA) [1], serta dengan menggunakan metode penghapusan fitur secara rekursif dan algoritma klasifikasi *Support Vector Machine* (SVM) untuk melakukan deteksi kasus penipuan kartu kredit [10]. Salah satu keunggulan metode SVM adalah memiliki kinerja yang baik pada kasus klasifikasi kelas biner (dua kelas). Selain itu, metode SVM lebih efektif pada kasus data berdimensi tinggi.

Berdasarkan uraian dari beberapa penelitian sebelumnya, belum didapat hasil pasti nilai akurasi terbaik yang dapat digunakan sebagai metode pembangunan model pendeteksi transaksi kartu kredit. Hal ini juga



dikarenakan data yang ada merupakan data yang tidak seimbang (imblance data) [7], [8]. Ketidakseimbangan data dapat mempengaruhi hasil akhir dalam memperoleh nilai akurasi maka, diperlukan metode *resampling* yang dapat memperbaiki kuantitas data sehingga hasil akurasi semakin baik. Metode *resampling* yang paling sering digunakan adalah metode SMOTE [8], [10]–[12]. Padahal masih ada beberapa metode *resampling* lain yang dapat digunakan seperti metode *Jackknife* dan *bootstrap* [13], serta algoritma residual *resampling* [14].

Dari karakteristik data kartu kredit yang termasuk ke dalam data imbalance, sehingga dalam penelitian ini akan dilakukan sebuah proses data *resampling* dan klasifikasi data. Metode *resampling* yang digunakan diantaranya SMOTE, *jackknife* dan *bootstrap*. Selain itu untuk menghasilkan sebuah model yang dapat membedakan transaksi kartu kredit yang valid dan invalid, maka beberapa metode machine learning seperti SVM, ANN, dan *Random Forest*, akan dibandingkan untuk mendapatkan kinerja model yang terbaik.

2. METODOLOGI PENELITIAN

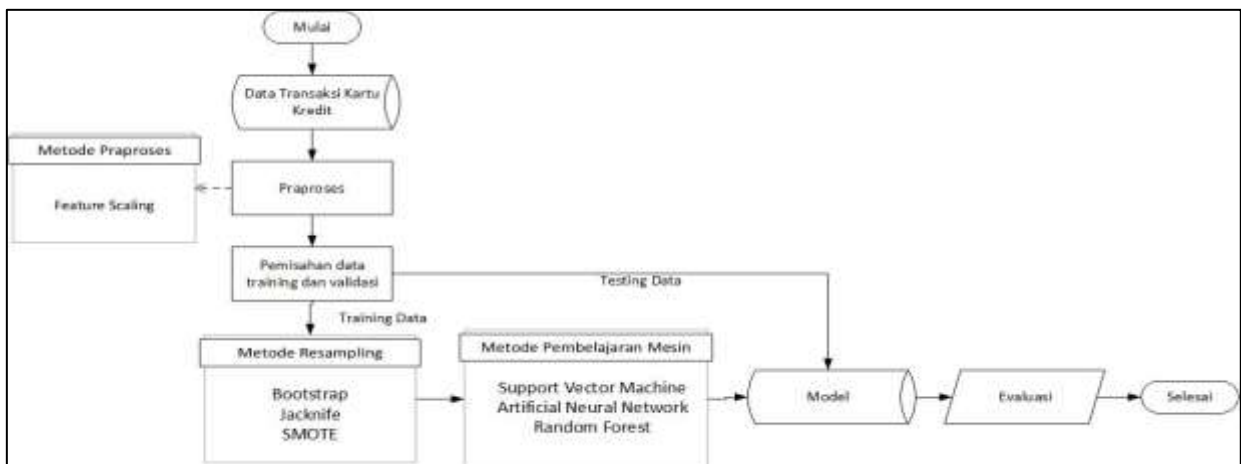
2.1 Dataset

Dataset yang digunakan sebagai masukan dalam penelitian ini bersumber dari <https://www.kaggle.com/mlg-ulb/creditcardfraud>. Jumlah data sebanyak 284.807 dataset transaksi. Pembagian data untuk data training dan data testing adalah sebanyak 70:30. Data training berjumlah 199.364 transaksi sedangkan data testing berjumlah 85.443 transaksi.

2.2 Pra-Pengolahan Data

Pada tahap ini, dilakukan proses pra-pengolahan data berupa *Resampling* data. Jumlah data transaksi yang tidak seimbang dapat menyebabkan model yang dihasilkan kurang dapat dipercaya karena banyaknya data sebagai inputan berpengaruh terhadap model yang dihasilkan. Karena data transaksi yang tersedia mengalami ketidakseimbangan antara transaksi asli dan transaksi palsu.

Maka sebelum data diolah, akan dilakukan proses *Resampling* terlebih dahulu. Metode *Resampling* yang akan digunakan dalam penelitian ini adalah metode *Bootstrap*, *Jacknife* dan SMOTE. Ketiga metode *Resampling* yang digunakan bertujuan untuk melihat metode mana yang lebih optimal dalam memperbaiki kondisi data yang tidak seimbang sehingga menghasilkan model yang *robust* dan terpercaya.



Gambar 1. Flowchart Diagram.

2.3 Pengembangan Model

Data yang telah melalui tahap pra-pengolahan selanjutnya akan diolah dengan menggunakan tiga arsitektur pembelajaran mesin untuk menghasilkan model yang dapat mendeteksi transaksi penipuan kartu kredit. Tiga arsitektur yang akan digunakan diantaranya SVM, ANN dan *Random Forest*.

Setelah mendapatkan model pelatihan, selanjutnya akan dilakukan uji coba data untuk mengetahui nilai parameter yang ditetapkan. Tahap training dan pengujian menggunakan dataset dari sumber yang sama, namun data yang digunakan merupakan data yang berbeda sehingga model yang dibangun merupakan model yang *robust* dan terpercaya. Secara umum alur proses dari penelitian ini ditunjukkan pada gambar 1.

2.3.1 Support Vector Machine

Support Vector Machine (SVM) adalah sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linier dalam sebuah ruang fitur (*feature space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan *learning bias* yang berasal dari teori pembelajaran statistik [10]. Teori yang mendasari SVM sendiri sudah berkembang sejak 1960-an, tetapi baru diperkenalkan oleh Vapnik, Boser dan Guyon pada tahun 1992 dan sejak itu SVM berkembang dengan pesat.



SVM adalah salah satu teknik yang relatif baru dibandingkan dengan teknik lain, tetapi memiliki performansi yang lebih baik di berbagai bidang aplikasi seperti pengenalan tulisan tangan, klasifikasi teks dan lain sebagainya [15]. SVM digunakan untuk mencari *hyperplane* terbaik dengan memaksimalkan jarak antar kelas. *Hyperplane* adalah sebuah fungsi yang dapat digunakan untuk pemisah antar kelas. Dalam 2-D fungsi yang digunakan untuk klasifikasi antar kelas disebut sebagai *line* whereas, fungsi yang digunakan untuk klasifikasi antar kelas dalam 3-D disebut *plane* similarly, sedangkan fungsi yang digunakan untuk klasifikasi di dalam ruang kelas dimensi yang lebih tinggi di sebut *hyperplane*.

2.3.2 Artificial Neural Network

Artificial Neural Network (ANN) adalah algoritma yang mampu melakukan ekstraksi pola data meskipun pola tersebut bersifat kompleks. Algoritma ini biasa digunakan pada kasus klasifikasi dan prediksi. ANN adalah sekumpulan neuron yang terorganisir dalam lapisan-lapisan (layers), diantaranya:

- Input layer: lapisan yang membawa data masuk kedalam system untuk kemudian di proses pada layer selanjutnya.
- Hidden layer: lapisan antara input layer dan output layer, dimana artificial neuron yang memiliki sekumpulan input pembobot '*weight*' dan prosedur untuk menghasilkan output neuron melalui *activation function*.
- Output layer: lapisan terakhir dari neuron yang menghasilkan output system.

2.3.3 Random Forest

Random Forest adalah kombinasi dari masing – masing *tree* yang baik kemudian dikombinasikan ke dalam satu model. *Random Forest* bergantung pada sebuah nilai vector random dengan distribusi yang sama pada semua pohon yang masing-masing *decision tree* yang memiliki kedalaman maksimal.

Random Forest adalah *classifier* yang terdiri dari *classifier* yang berbentuk pohon (*tree*) $\{h(\mathbf{x}, \theta_k), k = 1, \dots\}$ dimana θ_k adalah random vector yang didistribusikan secara independen dan masing masing *tree* pada sebuah unit akan memilih *class* yang paling populer pada input \mathbf{x} .

2.4 Metode Resampling Data

2.4.1 Synthetic Minority Oversampling Technique Strategy (SMOTE)

Salah satu strategi untuk mengurangi ketidakseimbangan kelas dengan membuat contoh baru di lingkungan kelas minoritas adalah SMOTE [16]. Pembuatan contoh kelas minoritas baru dilakukan dengan merata-ratakan contoh kelas minoritas yang berdekatan (bertetangga). Untuk fitur kategoris, rata-rata terdiri dari pemungutan suara mayoritas untuk kategori: kategori yang paling banyak diwakili di lingkungan menjadi kategori fitur kategorikal yang sesuai dari contoh yang baru dibuat. Metode ini menunjukkan peningkatan kinerja pengklasifikasi dengan menggunakan strategi penyeimbangan ini. Namun, metode SMOTE tidak mempertimbangkan label contoh yang berdekatan (bertetangga) saat membuat contoh dari kelas minoritas. Hal ini dapat menyebabkan tumpang tindih antara kelas minoritas dan mayoritas [12].

2.4.2 Bootstrap

Metode *bootstrap* adalah metode *Resampling* pada sampel data dengan syarat pengembalian data untuk penyelesaian masalah statistik ukuran suatu sampel dapat mewakili data pada populasi sebenarnya. Biasanya data *Resampling* diambil berulang kali sehingga dapat mewakili populasi data. Algoritma *bootstrap* terdiri dari *simple random sampling with replacement* (SRSWR) pada sample asli, ukuran sample baru biasanya sama dengan sampel asli. Sampel yang diperoleh dari iterasi pertama pada suatu algoritma disebut dengan sampel *bootstrap*, dan dilambangkan dengan $S^{(b)}$. Misal $\kappa_i^{(b)}$ adalah berapa kali unit i dari sample asli muncul pada sampel *bootstrap* $S^{(b)}$. Definisikan bobot *bootstrap* sebagai $w_i^{(b)} = w_i \kappa_i^{(b)}$, untuk $i \in s$. Kemudian *bootstrap* memperkirakan total diberikan dengan persamaan [17]:

$$Y^{(b)} = \sum_{i \in s} w_i^{(b)} y_i \quad (1)$$

Dasar pendekatan *Bootstrap* adalah dengan memperlakukan sampel sebagai populasi dan dengan menggunakan sampling Monte Carlo untuk membangkitkan dan mengkonstruksi estimator empiris dari distribusi sampling statistik. Distribusi sampling sebagai nilai-nilai statistik yang dihitung dari sampel yang tak terhingga yang acak dan berukuran n dari suatu populasi yang diberikan. Sampling Monte Carlo mengambil konsep ini untuk membangun distribusi sampling suatu estimator dengan mengambil sejumlah besar sampel berukuran n secara acak yang berasal dari populasi dan menghitung statistik tersebut dari nilai-nilai distribusi sampling yang ada [18].

2.4.3 Jackknife

Metode *Resampling* yang lain, yakni dengan pendekatan *jackknife*. *Jackknife* sendiri adalah alternatif dari *bootstrap*. *Jackknife* merupakan metode *Resampling* lebih umum digunakan apabila variabel regressor adalah fix. Ada dua cara dalam proses *jackknife* resampling, yaitu [13] :



Estimasi parameter dengan OLS, namun menghilangkan satu per satu observasi, dilakukan berulang sampai (n-1) kali dan mencari taksiran parameter dari rata-rata parameter (beta) setiap kali *resampling* dilakukan.

Estimasi model dengan menghilangkan d-observasi sekaligus, dilakukan berulang sampai sebanyak S kali, dimana S (S = n kombinasi d). Taksiran parameter diperoleh dengan merata-ratakan parameter yang diperoleh disetiap kali *resampling*.

2.5 Evaluasi kinerja pengklasifikasi pembelajaran mesin

Cara tradisional untuk mengevaluasi pengklasifikasi pembelajaran mesin menggunakan metode *Confusion Matrix* yang menjelaskan perbedaan antara kebenaran dasar dari dataset dan prediksi model (lihat tabel 1). *Confusion Matrix* adalah titik awal yang baik untuk mengevaluasi kinerja pengklasifikasi. Model yang sempurna akan memiliki semua contoh positif diprediksi positif dan semua contoh negatif diprediksi negatif: FN dan FP akan sama dengan 0.

Tabel 1. *Confusion Matrix*

	Prediksi Positif	Prediksi Negatif
Aktual Positif	TP (<i>True Positive</i>)	TN (<i>True Negative</i>)
Aktual Negatif	FP (<i>False Positive</i>)	FN (<i>False Negative</i>)

Persamaan untuk perhitungan *confusion matrix* adalah:

$$\text{Presisi} = \frac{TP}{TP+FP} \tag{2}$$

$$\text{Sensitivitas/Recall} = \frac{TP}{TP+FN} \tag{3}$$

$$\text{Akurasi} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$\text{F1 Score} = \frac{2 \times \text{Presisi} \times \text{Sensitivitas}}{\text{Presisi} + \text{Sensitivitas}} \tag{5}$$

3. HASIL DAN PEMBAHASAN

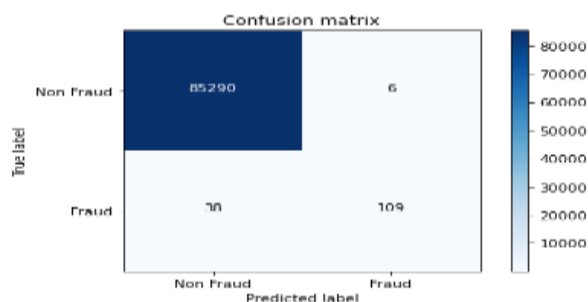
3.1 Hasil Klasifikasi Tanpa Menggunakan Metode Resampling

Tahap awal pengujian dilakukan tanpa menggunakan metode *resampling*. Proses pelatihan dilakukan pada metode pembelajaran mesin SVM, ANN, dan *Random Forest*. Pada metode SVM, parameter yang digunakan adalah nilai *overlapping* sebesar 10 dan kernel *radial basis function*. Untuk model ANN menggunakan arsitektur dengan 4 hidden layer dengan masing-masing memiliki 32, 16, 8, dan 4 node, sedangkan untuk metode *Random Forest* menggunakan jumlah tree sebesar 100 tree. Hasil pengujian tanpa menggunakan metode *Resampling* ditunjukkan pada tabel 2.

Tabel 2. Kinerja Metode Pembelajaran Mesin Tanpa Metode Resampling

Metode	Akurasi	Precision	Recall	F1-Score
SVM	99.94%	66.67%	97.03%	79.03%
ANN	26.53%	70.07%	0.16%	0.33%
Random Forest	99.95%	74.15%	94.78%	83.21%

Pada tabel 2 terlihat bahwasanya metode SVM memberikan kinerja klasifikasi terbaik dibandingkan dengan metode lainnya pada kasus tanpa proses *resampling*. Hasil ini didukung dengan hasil *confusion matrix* pada data set uji seperti yang ditunjukkan pada gambar 2.



Gambar 2. Confusion Matrix Model SVM



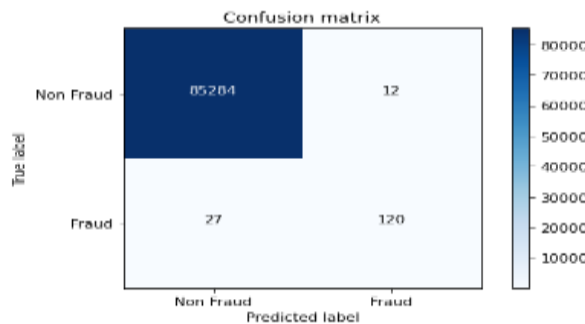
3.2 Hasil Klasifikasi Menggunakan Metode Resampling SMOTE

Pada tahap pengujian kedua dilakukan dengan menggunakan metode *Resampling SMOTE*. Proses pelatihan dilakukan pada metode pembelajaran mesin SVM, ANN, dan *Random Forest*. Pada metode SVM, parameter yang digunakan adalah nilai *overlapping* sebesar 10 dan kernel *radial basis function*. Untuk model ANN menggunakan arsitektur dengan 4 hidden layer dengan masing-masing memiliki 32, 16, 8, dan 4 node, sedangkan untuk metode *Random Forest* menggunakan jumlah tree sebesar 100 tree. Hasil pengujian menggunakan metode *resampling* ditunjukkan pada tabel 3.

Tabel 3. Kinerja Metode Pembelajaran Mesin Menggunakan Metode *Resampling SMOTE*.

Metode	Akurasi	Precision	Recall	F1-Score
SVM	99.07%	83.67%	13.71%	23.56%
ANN	99.89%	82.99%	63.21%	71.76%
<i>Random Forest</i>	99.95%	81.63%	90.91%	86.02%

Pada tabel 3 terlihat bahwasanya metode *Random Forest* memberikan kinerja klasifikasi terbaik dibandingkan dengan metode lainnya pada kasus *resampling* menggunakan metode SMOTE. Hasil ini didukung dengan hasil *confusion matrix* pada data set uji seperti yang ditunjukkan pada gambar 3.



Gambar 3. *Confusion Matrix* Metode *Random Forest* menggunakan SMOTE

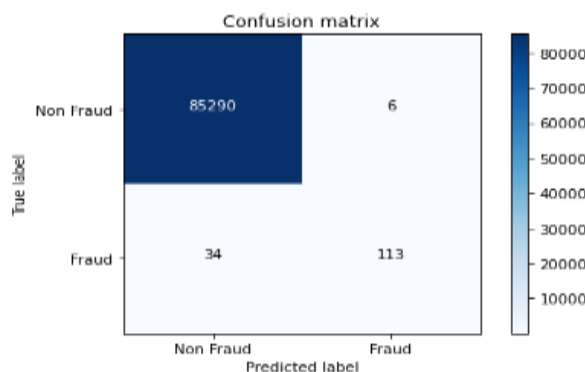
3.3 Hasil Klasifikasi Menggunakan Metode Resampling Jackknife

Pada tahap pengujian kedua dilakukan dengan menggunakan metode *Resampling Jackknife*. Proses pelatihan dilakukan pada metode pembelajaran mesin SVM, ANN, dan *Random Forest*. Pada metode SVM, parameter yang digunakan adalah nilai *overlapping* sebesar 10 dan kernel *radial basis function*. Untuk model ANN menggunakan arsitektur dengan 4 hidden layer dengan masing-masing memiliki 32, 16, 8, dan 4 node, sedangkan untuk metode *Random Forest* menggunakan jumlah tree sebesar 100 tree. Hasil pengujian tanpa menggunakan metode *resampling* ditunjukkan pada tabel 4.

Tabel 4. Kinerja Metode Pembelajaran Mesin menggunakan Metode *Resampling Jackknife*.

Metode	Akurasi	Precision	Recall	F1-Score
SVM	98.14%	89.12%	7.70%	14.17%
ANN	99.90%	84.35%	65.61%	73.81%
<i>Random Forest</i>	99.95%	76.87%	94.96%	84.96%

Pada tabel 4 terlihat bahwasanya metode *Random Forest* memberikan kinerja klasifikasi terbaik dibandingkan dengan metode lainnya pada kasus *Resampling* menggunakan metode *Jackknife*. Hasil ini didukung dengan hasil *confusion matrix* pada data set uji seperti yang ditunjukkan pada gambar 4.



Gambar 4. *Confusion Matrix* Metode *Random Forest* menggunakan *Jackknife*.



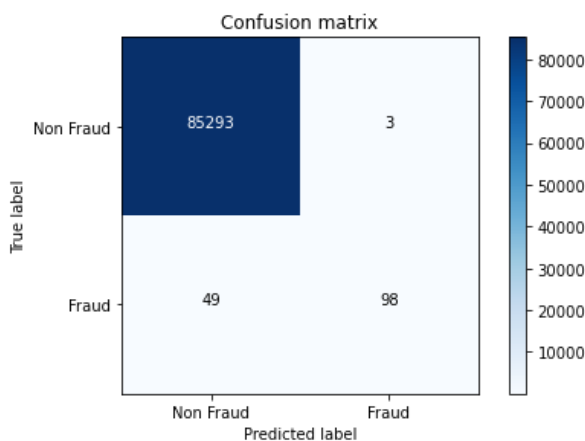
3.4 Hasil Klasifikasi menggunakan Metode *Resampling Bootstrap*

Pada tahap pengujian kedua dilakukan dengan menggunakan metode *Resampling bootstrap*. Proses pelatihan dilakukan pada metode pembelajaran mesin SVM, ANN, dan *Random Forest*. Pada metode SVM, parameter yang digunakan adalah nilai overlapping sebesar 10 dan kernel radial basis function. Untuk model ANN menggunakan arsitektur dengan 4 hidden layer dengan masing-masing memiliki 32, 16, 8, dan 4 node, sedangkan untuk metode *Random Forest* menggunakan jumlah tree sebesar 100 tree. Hasil pengujian tanpa menggunakan metode *Resampling* ditunjukkan pada tabel 5.

Tabel 5. Kinerja Metode Pembelajaran Mesin Menggunakan Metode *Resampling Bootstrap*.

Metode	Akurasi	Precision	Recall	F1-Score
SVM	99.94%	66.67%	97.03%	79.03%
ANN	99.89%	82.99%	63.21%	71.76%
<i>Random Forest</i>	99.94%	66.67%	97.03%	79.03%

Pada tabel 4 terlihat bahwasanya metode *Random Forest* memberikan kinerja klasifikasi terbaik dibandingkan dengan metode lainnya pada kasus *Resampling* menggunakan metode *bootstrap*. Hasil ini didukung dengan hasil *confusion matrix* pada data set uji seperti yang ditunjukkan pada gambar 5.



Gambar 5. Confusion Matrix Metode *Random Forest* menggunakan *bootstrap*

4. KESIMPULAN

Penelitian ini mencoba mengatasi fenomena imbalance data pada dataset transaksi kartu kredit menggunakan metode *resampling*. Tiga metode *Resampling* yang digunakan yaitu SMOTE, *bootstrap*, dan *Jackknife*. Selain itu tiga metode pembelajaran mesin digunakan untuk memvalidasi hasil *Resampling* data. Pengujian dilakukan dengan membagi dataset ke dalam dua jenis dataset yaitu dataset pelatihan dan pengujian dengan proporsi masing-masing sebesar 70 dan 30 persen. Dari hasil pengujian terlihat bahwa metode *resampling* berhasil mengatasi fenomena data imbalance. Sebagai contoh nilai F1-score pada dataset yang tidak dilakukan metode *resampling* masing-masing sebesar 79.03%, 0.33%, dan 83.21% untuk SVM, ANN dan *Random Forest*. Selanjutnya dengan menggunakan metode *resampling* SMOTE, nilai F1-Score meningkat 3-40% menjadi 23.56%, 71.76%, dan 86.02% untuk SVM, ANN dan *Random Forest*. Khusus untuk metode SVM terjadi penurunan kinerja sekitar 40%. Hal ini terjadi karena metode SVM memiliki kelemahan dalam memproses data dalam jumlah yang besar serta dimensi yang tinggi. Dari pengujian semua metode *Resampling* berhasil mengatasi data imbalance dan hasil perbandingan metode *Resampling* diperoleh metode SMOTE yang memiliki kinerja terbaik. Selain itu jika dilihat dari sisi metode pembelajaran mesin, metode *Random Forest* berhasil melampaui metode pembelajaran mesin lainnya. Kombinasi metode *Resampling* SMOTE dan *Random Forest* menghasilkan metrik kinerja sebesar 99.95%, 81.63%, 90.91% dan 86.02% masing-masing untuk Akurasi, Presisi, Recall dan F1-Score.

REFERENCES

[1] T. Pourhabibi, K. L. Ong, B. H. Kam, and Y. L. Boo, "Fraud detection: A systematic literature review of graph-based anomaly detection approaches," *Decis. Support Syst.*, vol. 133, no. April, p. 113303, 2020.

[2] V. N. Dornadula and S. Geetha, "Credit Card Fraud Detection using Machine Learning Algorithms," *Procedia Comput. Sci.*, vol. 165, pp. 631–641, 2019.

[3] Dornadula, V. N., & Geetha, S. "Credit Card Fraud Detection using Machine Learning Algorithms. *Procedia Computer Science*", 165, 631–641. 2019.

[4] S. N. Prasetyo, "Rumusan Pengaturan Credit Card Fraud Dalam Hukum Pidana Indonesia Ditinjau Dari Asas Legalitas," *J. Ilm. Huk. Leg.*, vol. 24, no. 1, p. 101, 2017.



- [5] Carcillo, Fabrizio, et al. "Combining unsupervised and supervised learning in credit card fraud detection." *Information sciences* 557 (2021): 317-331.
- [6] Ali, Mohammed Aamir, et al. "Consumer-facing technology fraud: Economics, attack methods and potential solutions." *Future Generation Computer Systems* 100 (2019): 408-427.
- [7] I. Sadgali, N. Sael, and F. Benabbou, "Performance of machine learning techniques in the detection of financial frauds," *Procedia Comput. Sci.*, vol. 148, no. Icds 2018, pp. 45-54, 2019.
- [8] X. Yu, X. Li, Y. Dong, and R. Zheng, "A Deep Neural Network Algorithm for Detecting Credit Card Fraud," in *2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, 2020, pp. 181-183.
- [9] H. Zhu, G. Liu, M. Zhou, Y. Xie, A. Abusorrah, and Q. Kang, "Optimizing Weighted Extreme Learning Machines for imbalanced classification and application to credit card fraud detection," *Neurocomputing*, vol. 407, pp. 50-62, 2020.
- [9] S. Bagga, A. Goyal, N. Gupta, and A. Goyal, "Credit Card Fraud Detection using Pipeling and Ensemble Learning," *Procedia Comput. Sci.*, vol. 173, no. 2019, pp. 104-112, 2020.
- [10] N. Rtayli and N. Enneya, "Enhanced credit card fraud detection based on SVM-recursive feature elimination and hyper-parameters optimization," *J. Inf. Secur. Appl.*, vol. 55, no. September, p. 102596, 2020.
- [11] G. Douzas and F. Bacao, "Geometric SMOTE a geometrically enhanced drop-in replacement for SMOTE," *Inf. Sci. (Ny)*, vol. 501, pp. 118-135, 2019.
- [12] D. Elreedy and A. F. Atiya, "A Comprehensive Analysis of Synthetic Minority Oversampling Technique (SMOTE) for handling class imbalance," *Inf. Sci. (Ny)*, vol. 505, pp. 32-64, 2019.
- [13] Wang, Leyang, and Fengbin Yu. "Jackknife resample method for precision estimation of weighted total least squares." *Communications in Statistics-Simulation and Computation* 50.5 (2021): 1272-1289.
- [14] S. S. Hosseini and M. M. Jamali, *Resampling methods combined with rao-blackwellized monte carlo data association algorithm*. Elsevier Inc., 2019.
- [15] Shaji, Anchana, et al. "Fraud Detection in Credit Card Transaction Using ANN and SVM." *International Conference on Ubiquitous Communications and Network Computing*. Springer, Cham, 2021.
- [16] Soltanzadeh, Paria, and Mahdi Hashemzadeh. "RCSMOTE: range-controlled synthetic minority over-sampling technique for handling the class imbalance problem." *Information Sciences* 542 (2021): 92-111.
- [17] Sagi, Omer, and Lior Rokach. "Ensemble learning: A survey." *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.4. 2018
- [18] Yang, Wenyi, et al. "Research on Bootstrapping Algorithm for Health Insurance Data Fraud Detection Based on Decision Tree." *2021 7th IEEE Intl Conference on Big Data Security on Cloud (BigDataSecurity), IEEE Intl Conference on High Performance and Smart Computing (HPSC) and IEEE Intl Conference on Intelligent Data and Security (IDS)*. IEEE, 2021.