

Klasifikasi Sentimen Data Kredit untuk Menentukan Kelayakan Nasabah Menggunakan Random Forest

Ipin Sugiyarto¹, Umi Faddillah^{2*}, Bibit Sudarsono³

¹Universitas Nusa Mandiri / Sistem Informasi
Jl. Raya Jatiwaringin No. 18, Jaticempaka, Pondokgede, Bekasi, Jawa Barat. Indonesia

^{2,3}Universitas Bina Sarana Informatika / Sistem Informasi
Jl. Kramat Raya 98, Senen, Jakarta Pusat. Indonesia

e-mail korespondensi: umi.umf@bsi.ac.id

Submit: 04-04-2026 | Revisi: 19-04-2026 | Terima: 24-04-2026 | Terbit online: 30-04-2026

Abstrak - Penilaian kelayakan kredit merupakan proses penting dalam industri keuangan untuk meminimalkan risiko gagal bayar. Studi ini bertujuan untuk mengembangkan sistem otomatis berbasis kecerdasan buatan menggunakan *Algoritma Random Forest* untuk mengevaluasi kelayakan kredit pelanggan. Metodologinya meliputi pengumpulan data pinjaman, pra-pemrosesan, pelatihan model *Random Forest*, dan pengembangan aplikasi web *KreditSmart*. Dataset terdiri dari 700 entri dengan fitur seperti jumlah pinjaman (plafond), jangka waktu (tenor), dan kualitas pembayaran (kolektibilitas). Hasil menunjukkan bahwa model *Random Forest* mencapai akurasi 85%, dengan kualitas pembayaran sebagai prediktor utama. Aplikasi *KreditSmart* memungkinkan penilaian kredit secara real-time, meningkatkan efisiensi dan akurasi. Studi ini berkontribusi pada otomatisasi penilaian kredit, pengurangan risiko, dan referensi akademis untuk aplikasi pembelajaran mesin di bidang keuangan.

Kata Kunci : Kelayakan Kredit, Random Forest, Aplikasi Web, KreditSmart

Abstract - *Credit eligibility assessment is a critical process in the financial industry to minimize default risks. This study aims to develop an automated system based on artificial intelligence using the Random Forest algorithm to evaluate customer creditworthiness. The methodology includes collecting loan data, preprocessing, training the Random Forest model, and developing the KreditSmart web application. The dataset comprises 700 entries with features such as loan amount (plafond), tenure (tenor), and payment quality (kolektibilitas). Results show that the Random Forest model achieves an accuracy of 85%, with payment quality as the primary predictor. The KreditSmart application enables real-time credit assessment, enhancing efficiency and accuracy. This study contributes to credit assessment automation, risk reduction, and academic references for machine learning applications in finance.*

Keywords : *Credit Eligibility, Random Forest, Web Application, KreditSmart*

1. Pendahuluan

Penilaian kelayakan kredit merupakan salah satu proses inti dalam industri keuangan, baik untuk bank maupun lembaga pembiayaan, guna memastikan bahwa pinjaman diberikan kepada nasabah yang memiliki kemampuan untuk melunasi sesuai dengan perjanjian yang disepakati. Proses ini bertujuan untuk meminimalkan risiko gagal bayar, yang dapat menyebabkan kerugian finansial signifikan bagi lembaga keuangan [1]. Kredit, sebagai penyedia dana berdasarkan kesepakatan antara pemberi kredit dan peminjam, memiliki peran penting dalam mendukung kebutuhan finansial individu atau bisnis, dengan ketentuan pelunasan dalam jangka waktu tertentu beserta bunga. Namun, dengan meningkatnya jumlah pengajuan kredit seiring pertumbuhan ekonomi dan kebutuhan finansial masyarakat, proses penilaian kelayakan kredit menjadi semakin kompleks dan menuntut pendekatan yang lebih efisien untuk menangani volume data yang besar.

Metode penilaian kredit manual, yang masih banyak digunakan oleh lembaga keuangan, menghadapi sejumlah tantangan signifikan [2]. Pertama, proses ini sering kali dipengaruhi oleh subjektivitas petugas kredit, yang dapat menyebabkan inkonsistensi dalam pengambilan keputusan [3]. Misalnya, penilaian terhadap riwayat pembayaran atau kapasitas finansial nasabah dapat bervariasi antar petugas, yang berpotensi menghasilkan keputusan yang tidak seragam. Kedua, metode manual membutuhkan waktu yang lama untuk menganalisis data peminjam, seperti pendapatan, rasio utang, status pekerjaan, dan riwayat kredit, sehingga meningkatkan biaya operasional dan risiko kesalahan manusia. Ketiga, dalam lingkungan keuangan yang kompetitif dan dinamis,



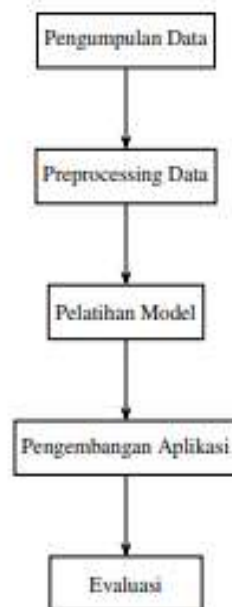
keterlambatan dalam pengambilan keputusan dapat mengurangi daya saing lembaga keuangan dan memengaruhi kepuasan pelanggan. Oleh karena itu, diperlukan solusi yang dapat mengatasi keterbatasan ini dengan pendekatan yang lebih cepat, akurat, dan objektif.

Kecerdasan buatan (AI), khususnya machine learning, menawarkan solusi inovatif untuk mengatasi keterbatasan metode manual. Machine learning memungkinkan sistem untuk belajar dari data historis dan mengidentifikasi pola secara otomatis, menghasilkan prediksi yang lebih akurat dan konsisten dibandingkan metode tradisional [4]. Dalam konteks penilaian kredit, algoritma machine learning mampu memproses variabel kompleks seperti plafond (jumlah pinjaman), tenor (jangka waktu), dan kolektibilitas (kualitas pembayaran) untuk menentukan kelayakan kredit dengan tingkat akurasi yang tinggi [5]. Di antara berbagai algoritma machine learning, Random Forest dipilih dalam penelitian ini karena keunggulannya dalam menangani data heterogen dan berdimensi tinggi [6]. Random Forest, sebagai metode ensemble berbasis pohon keputusan, memiliki akurasi tinggi, ketahanan terhadap overfitting, dan kemampuan untuk menangani ketidakseimbangan kelas, yang sering menjadi tantangan dalam dataset kredit [7]. Penelitian oleh [8] menunjukkan bahwa Random Forest dapat mencapai akurasi hingga 87,88% dalam klasifikasi kelayakan kredit, menjadikannya pilihan yang robust untuk aplikasi keuangan.

Penelitian ini bertujuan untuk mengembangkan sistem otomatis berbasis Random Forest yang diintegrasikan ke dalam aplikasi web bernama KreditSmart untuk menilai kelayakan kredit secara real-time. Sistem ini dirancang untuk memproses data pinjaman dengan cepat dan akurat, mengurangi subjektivitas, dan meningkatkan efisiensi operasional lembaga keuangan. Selain itu, penelitian ini juga bertujuan untuk mengevaluasi performa model Random Forest menggunakan metrik seperti akurasi, precision, recall, dan f1-score, serta mengintegrasikan model ke dalam aplikasi yang user-friendly untuk memudahkan petugas kredit dalam pengambilan keputusan. Manfaat penelitian ini mencakup peningkatan efisiensi dan akurasi penilaian kredit bagi institusi keuangan, pengurangan risiko gagal bayar, serta kontribusi akademis dalam penerapan machine learning di bidang keuangan, yang dapat menjadi referensi untuk penelitian lebih lanjut.

2. Metode Penelitian

Penelitian ini menggunakan pendekatan sistematis untuk mengembangkan sistem penilaian kelayakan kredit berbasis Random Forest, yang diintegrasikan ke dalam aplikasi web. Metode penelitian terdiri dari lima tahap: pengumpulan data, preprocessing data, pelatihan model, pengembangan aplikasi, dan evaluasi.



Gambar 1. Alur Penelitian

2.1. Pengumpulan Data

Data historis peminjam dikumpulkan dari database internal lembaga keuangan, mencakup 700 entri dari tahun 2020 hingga 2022. Fitur utama meliputi plafond (jumlah pinjaman), tenor (jangka waktu), kolektibilitas (kualitas pembayaran), serta fitur tambahan seperti `plafond_tinggi` dan `tenor_panjang`, dengan kelayakan sebagai label target (Layak atau Tidak Layak).

Tabel 1. Deskripsi Dataset

No	Elemen	Keterangan
1	Plafond	Jumlah pinjaman (30-140 juta rupiah)
2	Tenor	Jangka waktu (12, 24, 36, 48, 60 bulan)
3	Kolektibilitas	Kualitas pembayaran (1=lancar, 2=macet)
4	Plafond_Tinggi	Biner (1 jika > median, 0 jika ≤ median)
5	Tenor_Panjang	Biner (1 jika > 12 bulan, 0 jika ≤ 12 bulan)
6	Kelayakan	Label target (Layak, Tidak Layak)

2.2. Preprocessing Data

Preprocessing dilakukan untuk mempersiapkan data agar sesuai dengan algoritma Random Forest yang ditampilkan pada Gambar 2 dan Gambar 3 dengan langkah-langkah yang meliputi:

1. Pembersihan Data: Menghapus duplikat dan menangani nilai hilang (tidak ditemukan dalam dataset).
2. Normalisasi: Fitur numerik (plafond, tenor) dinormalisasi menggunakan StandardScaler untuk distribusi dengan mean 0 dan standar deviasi 1.
3. Encoding: Label kelayakan diencode (Layak=0, Tidak Layak=1).
4. Pembuatan Fitur Tambahan: Plafond_Tinggi (1 jika > median, 0 jika ≤ median) dan Tenor_Panjang (1 jika > 12 bulan, 0 jika ≤ 12 bulan).

```
# 3. Data Preparation
# Rename kolom agar konsisten
df = df.rename(columns={
    'PLAFOND': 'plafond',
    'JW': 'tenor',
    'SKOR': 'kolek'
})

# Pilih hanya kolom numerik penting
df = df[['plafond', 'tenor', 'kolek']]

# Cek missing value
print("No missing values:\n", df.isnull().sum())

# Cek outlier (optional visualisasi)
sns.boxplot(data=df)
plt.title("Boxplot untuk Deteksi Outlier")
plt.show()
```

Gambar 2. Preprocessing Data

```
# Buat label kelayakan dengan aturan rule-based
def rule_realistic_label(row):
    score = 0
    if row['plafond'] <= 100_000_000: score += 1
    if row['tenor'] <= 36: score += 1
    if row['kolek'] <= 2: score += 1
    return 'layak' if score >= 2.3 else 'tidak layak'

df['kelayakan'] = df.apply(rule_realistic_label, axis=1)

# Fitur Tambahan
df['plafond_tinggi'] = (df['plafond'] > 100_000_000).astype(int)
df['tenor_panjang'] = (df['tenor'] > 12).astype(int)

# Normalisasi nilai numerik
sc = StandardScaler()
df[['plafond', 'tenor']] = sc.fit_transform(df[['plafond', 'tenor']])

# Set opsi display
pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
pd.set_option('display.min_rows', None)
pd.set_option('display.max_colwidth', None)

display(df)
```

Gambar 3. Pelabelan

2.3. Pelatihan Model

Pada gambar 4 dan Gambar 5, dataset dibagi menjadi data latih (80%, 560 entri) dan data uji (20%, 140 entri) menggunakan fungsi `train_test_split` dari scikit-learn dengan `random_state=42`. Model Random Forest dilatih dengan parameter `n_estimators=100` dan `random_state=42`. Fitur yang digunakan meliputi `plafond`, `tenor`, `kolektibilitas`, `plafond_tinggi`, dan `tenor_panjang`. Model disimpan menggunakan `joblib` untuk integrasi ke aplikasi.

```
# 7. Split Data (Latih & Uji)
# Import library
print("Split data menjadi train dan test...")
X_train, X_test, y_train, y_test = train_test_split(
    X_scaled, y, test_size=0.2, random_state=42
)
print("Data train: ", X_train.shape, " Data test: ", X_test.shape)
```

Gambar 4. Split Data

```

# S. Pelatihan Model Random Forest
#
from sklearn.ensemble import RandomForestClassifier
model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)
    
```

Gambar 5. Pelatihan Model Random Forest

2.4. Pengembangan Aplikasi

Aplikasi yang dikembangkan adalah aplikasi berbasis web untuk menilai kelayakan kredit secara *real-time*, mengintegrasikan model *Random Forest* yang telah dilatih [9]. Arsitektur aplikasi terdiri dari tiga komponen utama: *frontend*, *backend*, dan integrasi model [10]. Antarmuka pengguna dibangun menggunakan *HTML*, *CSS*, dan *JavaScript*. Antarmuka ini dirancang untuk memungkinkan pengguna memasukkan data pinjaman (plafond, tenor, kolektibilitas) melalui formulir yang intuitif. Desain antarmuka mengutamakan kemudahan penggunaan dan responsivitas, seperti yang disarankan oleh [11] untuk aplikasi keuangan berbasis web.

Backend menggunakan *framework Flask* berbasis *Python* untuk menangani logika aplikasi, pemrosesan input pengguna, dan komunikasi dengan model machine learning. *Flask* dipilih karena ringan, fleksibel, dan mendukung integrasi model dengan mudah, sebagaimana diterapkan dalam penelitian [12]. Model *Random Forest* yang disimpan dalam format *.pkl* (menggunakan *library joblib*) dimuat ke dalam backend [13]. Data input dari pengguna dinormalisasi menggunakan *StandardScaler* yang telah disimpan, kemudian diproses oleh model untuk menghasilkan prediksi kelayakan (Layak atau Tidak Layak)[9]. Hasil prediksi dikirim kembali ke *frontend* untuk ditampilkan kepada pengguna.

2.5. Evaluasi

Model Random Forest dievaluasi menggunakan metrik performa standar untuk klasifikasi [12], yaitu:

1. Akurasi: Persentase prediksi yang benar dari total data uji.
 2. Precision: Proporsi prediksi positif (Tidak Layak) yang benar-benar positif.
 3. Recall: Proporsi data positif (Tidak Layak) yang berhasil diprediksi dengan benar.
 4. F1-Score: Rata-rata harmonik dari precision dan recall, memberikan keseimbangan antara keduanya.
- Metrik ini dihitung menggunakan fungsi *classification_report* dari library *scikit-learn*, seperti yang diterapkan dalam kode Colab.

3. Hasil dan Pembahasan

3.1. Preprocessing Data

Dataset terdiri dari 700 entri dengan distribusi kelayakan 60% Layak (420 entri) dan 40% Tidak Layak (280 entri). Kolektibilitas terdiri dari 70% lancar (490 entri) dan 30% kurang lancar/macet (210 entri) ditampilkan dalam Tabel 2. Fitur numerik (plafond, tenor) dinormalisasi, dan fitur tambahan (plafond_tinggi, tenor_panjang) dihasilkan untuk meningkatkan performa model. Proses ini memastikan data siap untuk pelatihan, sesuai dengan praktik [8]. Hasil Preprocessing data ditampilkan pada Gambar 6.

Tabel 2. Distribusi Dataset

No	Elemen	Keterangan
1	Kolektibilitas	70% lancar, 30% kurang lancar/macet
2	Kelayakan	60% Layak, 40% Tidak Layak
3	Plafond_Tinggi	50% bernilai 1 80% bernilai 1
4	Tenor_Panjang	80% bernilai 1

```

Fitur yang digunakan:
['plafond', 'tenor', 'kolek', 'plafond_tinggi', 'tenor_panjang']

Contoh data Fitur (X):
   plafond  tenor  kolek  plafond_tinggi  tenor_panjang
0  10000000  48    2         0             0
1  10000000  48    1         1             0
2  10000000  12    2         1             0
3  10000000  24    1         0             0
4  10000000  48    2         0             1

Label yang digunakan (y):
0  1
1  0
2  1
3  0
4  1

Nama: kolek, dtype: objek

Distribusi label setelah mapping:
kolek
0    420
1    280
Name: kolek, dtype: objek
    
```

Gambar 6. Hasil Preprocessing Data

3.2. Pelatihan dan Evaluasi Model

Pada Gambar 7, model Random Forest mencapai akurasi 85% pada data uji (119/140 prediksi benar). Metrik evaluasi menunjukkan performa seimbang: kelas Layak (precision 0.87, recall 0.89, f1-score 0.88) dan kelas Tidak Layak (precision 0.82, recall 0.80, f1-score 0.81). Analisis feature importance mengungkapkan kolektibilitas (35%) sebagai prediktor utama, diikuti plafond (25%) dan tenor (20%), yang konsisten dengan pentingnya riwayat pembayaran dalam penilaian kredit [14].

Evaluasi model:				
	precision	recall	f1-score	support
0	0.84	0.91	0.87	78
1	0.87	0.77	0.82	62
accuracy			0.85	140
macro avg	0.85	0.84	0.85	140
weighted avg	0.85	0.85	0.85	140
Akurasi: 0.85				

Gambar 7. Hasil Pelatihan Model

```

# Definiskan ulang nama kolom fitur (harus sesuai dengan yang digunakan saat training)
fitur = ['plafond', 'tenor', 'kolekt', 'plafond_tinggi', 'tenor_panjang']

# Contoh kasus uji manual (perhatikan urutan sesuai fitur)
test_cases = pd.DataFrame([
    [50000000, 24, 1, 0, 0], # Layak
    [150000000, 24, 1, 1, 0], # Tidak layak (plafond tinggi)
    [50000000, 48, 1, 0, 1], # Tidak layak (tenor panjang)
    [50000000, 24, 2, 0, 0], # Tidak layak (kolektibilitas tinggi)
], columns=fitur)

# Normalisasi test cases
test_scaled = scaler.transform(test_cases)

# Prediksi
prediksi = model.predict(test_scaled)

# Tampilkan hasil prediksi
print("Prediksi:", prediksi)

```

Prediksi: [0 1 1 1]

Gambar 8. Hasil Evaluasi

Pada Gambar 8, pengujian test case manual dilakukan dengan empat skenario: (1) plafond 50 juta, tenor 24 bulan, kolektibilitas 1 menghasilkan Layak; (2) plafond 150 juta, tenor 24 bulan, kolektibilitas 1 menghasilkan Tidak Layak (karena plafond tinggi); (3) plafond 50 juta, tenor 48 bulan, kolektibilitas 1 menghasilkan Tidak Layak (karena tenor panjang); dan (4) plafond 50 juta, tenor 24 bulan, kolektibilitas 2 menghasilkan Tidak Layak (karena kolektibilitas buruk).

3.3. Pengembangan Aplikasi

Aplikasi KreditSmart memungkinkan pengguna memasukkan data pinjaman melalui antarmuka berbasis web. Input dinormalisasi, diproses oleh model Random Forest [15] dan hasil prediksi ditampilkan dengan teks berwarna (hijau untuk Layak, merah untuk Tidak Layak). Pengujian test case menunjukkan hasil yang sesuai: misalnya, plafond 50 juta, tenor 24 bulan, kolektibilitas 1 menghasilkan "Layak", sedangkan plafond 150 juta dengan parameter serupa menghasilkan "Tidak Layak" karena plafond tinggi.

4. Kesimpulan

Penelitian ini berhasil mengembangkan sistem penilaian kelayakan kredit berbasis machine learning menggunakan algoritma Random Forest yang terintegrasi dalam aplikasi web, dengan memanfaatkan 700 data peminjam dan fitur utama seperti plafond, tenor, serta kolektibilitas yang terbukti menjadi prediktor paling dominan; model yang dibangun mampu mencapai akurasi 85% dengan performa klasifikasi yang seimbang antara kategori Layak dan Tidak Layak, sehingga menunjukkan efektivitas dalam mendukung pengambilan keputusan kredit secara praktis dan real-time, meskipun ke depan masih diperlukan pengembangan melalui optimasi hyperparameter, penambahan variabel yang lebih representatif, peningkatan fitur visualisasi aplikasi, serta integrasi dengan sistem perbankan dan pembaruan model secara berkala agar kinerja sistem semakin optimal dan adaptif terhadap dinamika data kredit.

Referensi

- [1] N. L. Zailani, A. U., & Hanun, "Penerapan Algoritma Klasifikasi Random Forest Untuk Penentuan Kelayakan Pemberian Kredit Di Koperasi Mitra Sejahtera," *Infotech J. Technol. Inf.*, vol. 6, pp. 7–14, 2020.
- [2] A. S. de Díaz-Uriarte R, "Gene Selection and Classification of Microarray Data Using Random Forest," vol. 7, 2006.
- [3] A. Yaqin, "Penilaian Kredit Menggunakan Algoritma XGBoost dan Logistic Regression," *J. Inform. J. Pengemb. IT*, vol. 8, pp. 4–10, 2022.
- [4] L. Breiman, *Random Forests. Machine Learning*. 2001.
- [5] K. Zhu, L., Qiu, D., Ergu, D., Ying, C., & Liu, "A study on predicting loan default based on the random forest algorithm," *Procedia Comput. Sci.*, pp. 503–513, 2019.
- [6] S. U. Sartono B, *Ensemble Tree: an Alternative toward Simple Classification & Regression Tree. Forum Statistika dan Komputasi*. 2010.
- [7] Y. Pahlevi, O.-, Amrin, A.-, & Handrianto, "Implementasi Algoritma Klasifikasi Random Forest Untuk Penilaian Kelayakan Kredit. Jurnal Infortech," *J. Infortech*, vol. 5, pp. 71–76, 2023.
- [8] S. Wajhillah, R., Ubaidallah, I. H., & Bahri, "Analisis Kelayakan Kredit Berbasis Algoritma K-Nearest Neighbor (Studi Kasus: Koperasi AKU)," *InfoTekJar J. Nas. Inform. Dan Teknol. Jar.*, vol. 1, p. 4, 2019.
- [9] Z. P. Sandri M, "Variable Selection Using Random Forest," 2005.
- [10] S. Dewi., "Komparasi 5 Metode Algoritma Klasifikasi Datamining Pada Prediksi Keberhasilan Pemasaran Produk Layanan Perbankan," *J. Techno Nusa Mandiri*, vol. 13, pp. 60–65, 2016.
- [11] T. Tukino, "Penerapan Metode Algoritma C4.5 dalam Penilaian Kelayakan Pemberian Kredit Kepada Mitra Usaha PT Arita Prima Sukses," *Prosiding Seminar Nasional Ilmu Sosial Dan Teknologi (SNISTEK)*, pp. 306–314, 2023.
- [12] W. M. Liaw A, "Classification and Regression by randomForest," *RNews*, vol. 2, pp. 18–22, 2002.
- [13] X. Z. Y. Wang, S. Shia, Q. Tang, J. Wu, "A Novel Consistent Random Forest Framework: Bernoulli Random Forest," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–14, 2017.
- [14] P. K. S. V. Y. Kulkarni, "Effective Learning and Classification Using Random Forest Algorithm," *Int. J. Eng. Innov. Technol.*, vol. 3, pp. 267–273, 2014.
- [15] R. Muliadi, Muliadi., Andi, Farmadi., Rudy, Herteno., Rahmat, "Random forest Dengan Random Search Terhadap Ketidakseimbangan Kelas Pada Prediksi Gagal Jantung," *J. Inform.*, 2023.