Implementasi Machine Learning untuk Mitigasi Risiko Gagal Bayar dalam Prediksi Risiko Kredit

Saiful Anwar

Project Based Internship Program
ID/X Partners – Data Scientist
December 2025

Informasi Video Project

HERE TO CONNECT

Outline

Urgensi dan Sasaran Strategis Project

Latar Belakang (The Problem)

- Tingginya Risiko Gagal Bayar: Munculnya nasabah Bad Loan (10,93%) dapat mengancam stabilitas arus kas dan menyebabkan kerugian finansial yang signifikan bagi perusahaan.
- Kompleksitas Evaluasi Kredit: Penilaian manual terhadap ratusan ribu data nasabah tidak lagi efisien dan rentan terhadap kesalahan manusia (human error).
- Kebutuhan Sistem Prediktif: Perusahaan memerlukan instrumen yang mampu mengenali pola perilaku nasabah secara akurat untuk memitigasi risiko sebelum pinjaman diberikan.

Tujuan Project (The Solution)

- Otomatisasi Penilaian Kredit: Membangun model Machine Learning yang dapat mengklasifikasikan nasabah menjadi kategori Good Loan atau Bad Loan secara otomatis.
- Optimasi Deteksi Risiko: Menghasilkan model dengan tingkat deteksi (Recall) yang tinggi (86%) guna menangkap potensi gagal bayar sedini mungkin.
- Rekomendasi Berbasis Data: Memberikan wawasan strategis mengenai faktor-faktor utama yang mendorong risiko kredit sebagai dasar pengambilan kebijakan perusahaan.

Metodologi Pengumpulan dan Pembersihan Data

DATA SOURCING

- Sumber Dataset: Menggunakan data historis pinjaman (Lending Club) periode 2007 - 2014.
- Volume Data: Total awal sebanyak 466.285 baris data dengan 75 kolom fitur awal.

```
# Load data - menggunakan low_memory=False karena dataset memiliki banyak kolom
df = pd.read_csv('loan_data_2007_2014.csv', low_memory=False)

# Cek 5 baris pertama
df.head()
```

	Unnamed: 0	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment	grade
0	0	1077501	1296599	5000	5000	4975.0	36 months	10.65	162.87	B
1	1	1077430	1314167	2500	2500	2500.0	60 months	15.27	59.83	C
2	2	1077175	1313524	2400	2400	2400.0	36 months	15.96	84.33	C
3	3	1076863	1277178	10000	10000	10000.0	36 months	13.49	339.31	C
4	4	1075358	1311748	3000	3000	3000.0	60 months	12.69	67.79	B

5 rows × 75 columns

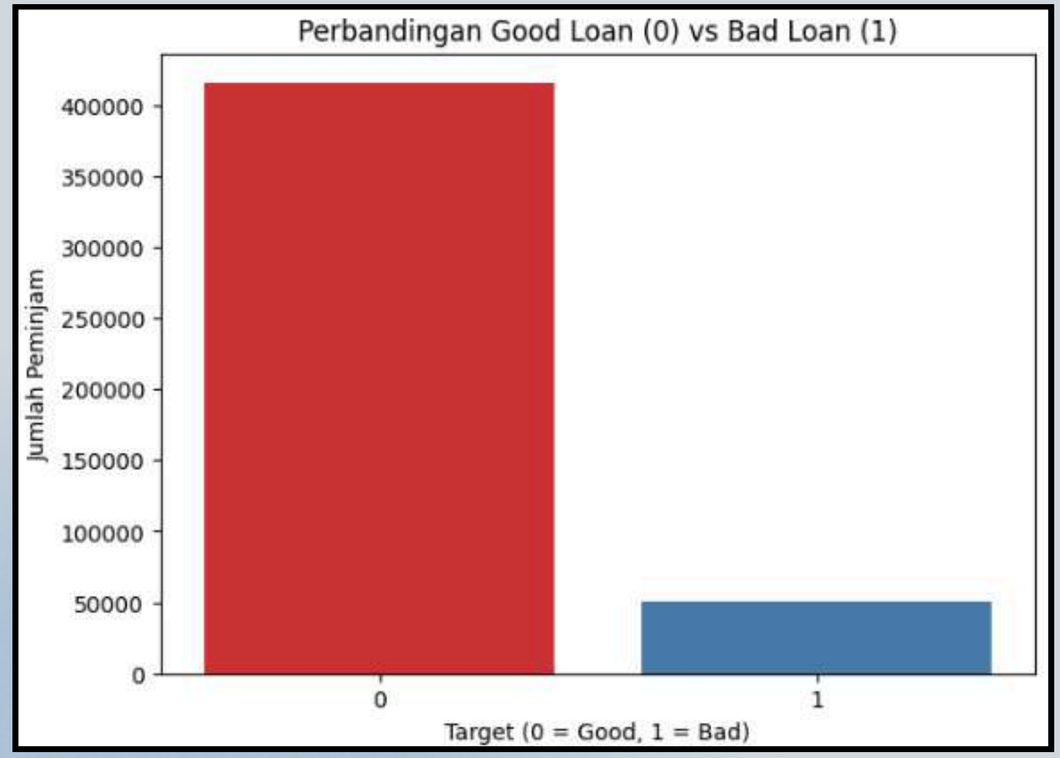
LANGKAH PEMBERSIHAN (DATA CLEANING)

- Eliminasi Kolom Irrelevant: Menghapus fitur yang tidak memiliki nilai prediktif seperti ID nasabah, URL, dan deskripsi teks bebas.
- Penanganan Missing Values: * Menghapus kolom yang memiliki nilai kosong (null) lebih dari 50%.
- Menghapus kolom dengan nilai tunggal (constant values) yang tidak memberikan informasi variasi.
- Seleksi Fitur Akhir: Dari 75 fitur awal, dipilih 33 fitur paling relevan yang akan digunakan untuk proses pemodelan.
- Data Imputation: Mengisi sisa nilai yang kosong menggunakan nilai median untuk menjaga integritas distribusi data.

```
# 1. Hapus kolom yang memiliki missing values lebih dari 50%
# Dataset ini punya banyak kolom yang benar-benar kosong di tahun tersebut
df_clean = df.dropna(thresh=len(df) * 0.5, axis=1)

# 2. Hapus kolom identitas atau teks yang tidak bisa diolah model
# 'Unnamed: 0' adalah indeks lama, 'id' & 'member_id' hanya nomor urut
cols_to_drop = ['Unnamed: 0', 'id', 'member_id', 'url', 'title', 'zip_code', 'policy_code', 'application_type']
df_clean = df_clean.drop(columns=[c for c in cols_to_drop if c in df_clean.columns])

print(f"Jumlah kolom awal: {df.shape[1]}")
print(f"Jumlah kolom setelah dibersihkan: {df_clean.shape[1]}")
```

```
Jumlah kolom awal: 75
Jumlah kolom setelah dibersihkan: 46
```

Eksplorasi Karakteristik Target




```
Proporsi Target:
target
0    89.069346
1    10.930654
Name: proportion, dtype: float64
```

Definisi Target:

- Good Loan (0): Nasabah yang status pinjamannya Fully Paid atau sedang berjalan lancar.
- Bad Loan (1): Nasabah yang gagal bayar (Charged Off, Default, Late).

Distribusi Data:

- Jumlah nasabah yang dikategorikan aman mencapai 89,07%.
- Jumlah nasabah yang dikategorikan berisiko (gagal bayar) sebesar 10,93%.

Analisis Ketidakseimbangan (Imbalanced Data):

- Terdapat dominasi nasabah kategori Good Loan yang signifikan.
- Tantangan: Model cenderung akan lebih pintar mengenali nasabah baik daripada nasabah buruk jika tidak ditangani dengan parameter khusus pada tahap pemodelan.

Transformasi Data Kategorikal

01 Konversi Variabel Teks

Mengubah fitur bertipe objek/teks (seperti grade, sub_grade, dan term) menjadi format numerik agar dapat diolah oleh algoritma matematika.

02 Fitur Durasi (Term)

Menghapus satuan "months" pada kolom durasi pinjaman sehingga menyisakan angka murni (36 atau 60).

03 Peringkat Kredit (Grade)

Memberikan nilai urutan (Encoding) pada kualitas kredit nasabah dari skala terbaik hingga terendah.

04 Lama Bekerja (Emp Length)

Membersihkan karakter non-numerik (seperti "+ years" atau "< 1 year") menjadi representasi angka tahun pengalaman kerja.

```python
# 1. Mengubah 'term' (jangka waktu) menjadi angka (misal: ' 36 months' -> 36)
df_clean['term'] = df_clean['term'].str.extract('(\d+)').astype(int)

# 2. Mengubah 'emp_length' (lama bekerja) menjadi angka
# Kita asumsikan '< 1 year' sebagai 0 dan '10+ years' sebagai 10
df_clean['emp_length'] = df_clean['emp_length'].str.extract('(\d+)').fillna(0).astype(int)

# 3. Mengubah 'grade' menjadi angka agar memiliki urutan risiko (A paling rendah risiko, G paling tinggi)
grade_map = {'A':1, 'B':2, 'C':3, 'D':4, 'E':5, 'F':6, 'G':7}
df_clean['grade'] = df_clean['grade'].map(grade_map)

print("Berhasil mengolah kolom term, emp_length, dan grade.")
df_clean[['term', 'emp_length', 'grade']].head()
```

Output

```
Berhasil mengolah kolom term, emp_length, dan grade.
```

	term	emp_length	grade
0	36	10	2
1	60	1	3
2	36	10	3
3	36	10	3
4	60	1	2

Pengolahan Data Temporal (Waktu)

- Tantangan Data Tanggal: Kolom tanggal (seperti issue_d, last_pymnt_d) tidak bisa diproses langsung oleh model karena formatnya berupa teks bulan-tahun (Contoh: "Dec-2011").
- Transformasi Durasi: Mengonversi data tanggal menjadi nilai numerik yang merepresentasikan jumlah bulan yang telah berlalu sejak peristiwa tersebut terjadi hingga titik referensi tertentu.

- Fitur yang Dihasilkan:
  1. Months Since Issue Date: Menghitung sudah berapa lama pinjaman tersebut berjalan sejak diterbitkan.
  2. Months Since Last Payment: Mengukur jarak waktu sejak nasabah terakhir kali membayar angsuran.

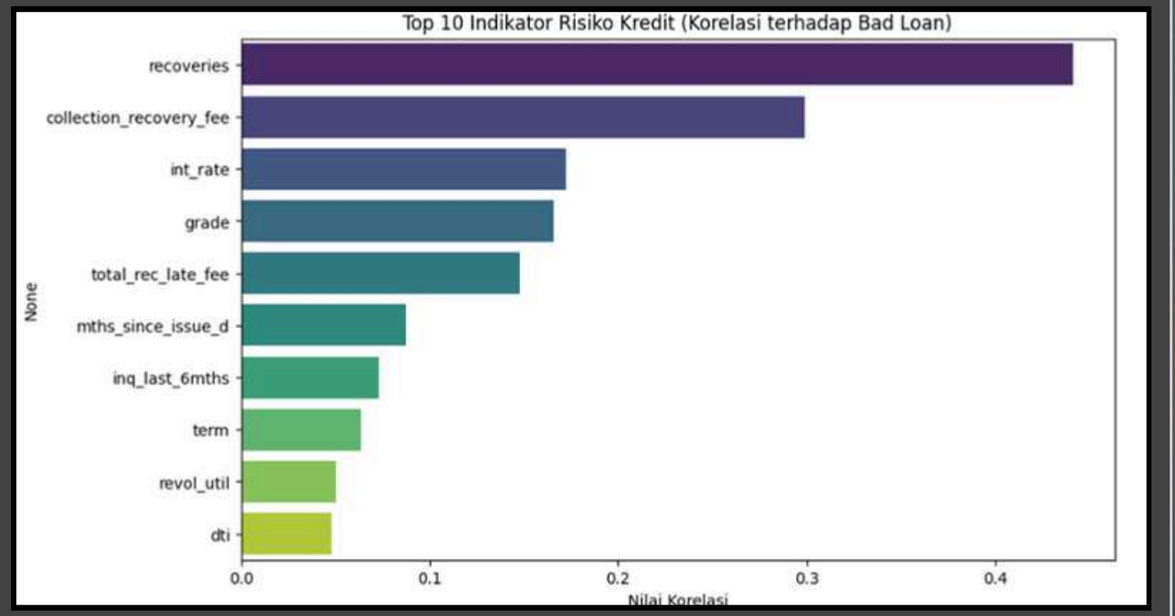
- Manfaat: Fitur ini membantu model memahami pola perilaku nasabah berdasarkan faktor kedekatan waktu, yang sangat krusial dalam mendeteksi risiko gagal bayar.

```python
# 1. Pastikan kolom issue_d sudah dalam format datetime
df_clean['issue_d'] = pd.to_datetime(df_clean['issue_d'], format='%b-%y')

# 2. Menghitung durasi bulan dengan cara yang didukung (selisih hari / 30.44)
# Kita hitung selisih hari dahulu, lalu bagi dengan rata-rata jumlah hari dalam sebulan
days_diff = (pd.to_datetime('2025-12-01') - df_clean['issue_d']).dt.days
df_clean['mths_since_issue_d'] = round(days_diff / 30.44)

# 3. Hapus kolom tanggal asli dan loan_status
cols_to_drop_date = ['issue_d', 'loan_status']
df_clean = df_clean.drop(columns=[c for c in cols_to_drop_date if c in df_clean.columns])

print("Perbaikan Berhasil! Kolom tanggal kini sudah menjadi durasi bulan.")
print(df_clean[['mths_since_issue_d']].head())
```

```
Perbaikan Berhasil! Kolom tanggal kini sudah menjadi durasi bulan.
   mths_since_issue_d
0               168.0
1               168.0
2               168.0
3               168.0
4               168.0
```

Analisis Faktor Risiko Utama dan Korelasi Variabel

- Identifikasi Indikator Risiko: Mengukur hubungan antara variabel fitur dengan variabel target (Bad Loan) menggunakan analisis korelasi.
- Korelasi Positif Tertinggi:
  1. Interest Rate (Suku Bunga): Semakin tinggi suku bunga, semakin besar kemungkinan nasabah mengalami gagal bayar.
  2. Recoveries: Nasabah yang sudah masuk tahap pemulihan dana memiliki indikasi kuat sebagai Bad Loan.
- Korelasi Negatif (Faktor Pelindung):
  1. Total Payment: Semakin besar total pembayaran yang sudah dilakukan, semakin kecil risiko gagal bayar.
- Tujuan Analisis: Memastikan bahwa fitur yang digunakan dalam model memiliki landasan logika bisnis yang kuat sebelum masuk ke tahap pelatihan.




```
Indikator Risiko Teratas (Korelasi terhadap Bad Loan):
recoveries                0.441171
collection_recovery_fee   0.299227
int_rate                  0.172361
grade                     0.165625
total_rec_late_fee        0.147750
mths_since_issue_d        0.087575
inq_last_6mths            0.073102
term                      0.063565
revol_util                0.050207
dti                       0.048102
Name: target, dtype: float64
```

Strategi Pembagian Dataset

Metode Train-Test Split:

Membagi seluruh dataset menjadi dua bagian independen untuk memastikan model diuji pada data yang belum pernah dilihat sebelumnya (unseen data).

Proporsi Pembagian:

- 80% Data Pelatihan (Training Set): Digunakan untuk melatih algoritma agar mengenali pola antara profil nasabah dan status kreditnya.
- 20% Data Pengujian (Testing Set): Digunakan sebagai simulasi dunia nyata untuk mengukur seberapa akurat model melakukan prediksi.

Konsistensi Data:

Menggunakan parameter random_state untuk memastikan hasil pengujian dapat direplikasi dan tetap konsisten.

```python
from sklearn.model_selection import train_test_split

# 1. Menentukan Fitur (X) dan Target (y)
# X adalah semua kolom angka kecuali target, y adalah kolom target (0/1)
X = df_clean.select_dtypes(exclude=['object']).drop(columns=['target'])
y = df_clean['target']

# 2. Mengisi sisa nilai kosong (NaN) dengan median
# Ini langkah keamanan terakhir agar model tidak error saat membaca data
X = X.fillna(X.median())

# 3. Membagi data menjadi Training dan Testing
# stratify=y sangat penting agar proporsi 10.9% Bad Loan tetap sama di kedua bagian
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42,
    stratify=y
)

print("--- Data Berhasil Dibagi ---")
print(f"Jumlah baris Data Training (untuk belajar): {X_train.shape[0]}")
print(f"Jumlah baris Data Testing (untuk ujian): {X_test.shape[0]}")
print(f"Jumlah kolom fitur: {X_train.shape[1]}")
```

```
--- Data Berhasil Dibagi ---
Jumlah baris Data Training (untuk belajar): 373028
Jumlah baris Data Testing (untuk ujian): 93257
Jumlah kolom fitur: 33
```

Pengembangan Model Machine Learning

- Pemilihan Algoritma: Menggunakan Random Forest Classifier, model ensemble yang sangat tangguh dalam menangani data besar dan fitur yang kompleks.
- Penanganan Imbalance Data: Menerapkan parameter class_weight='balanced' untuk memastikan model memberikan perhatian yang adil pada kelas nasabah yang gagal bayar (minority class).
- Proses Training: Model mempelajari pola dari 80% data historis untuk memahami perbedaan karakteristik antara peminjam yang gagal dan yang sukses.

```python
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score

# 1. Inisialisasi Model
# n_estimators=100: Menggunakan 100 pohon keputusan
# class_weight='balanced': Memberitahu model untuk lebih teliti pada nasabah 'Bad Loan' karena jumlahnya sedikit
rf_model = RandomForestClassifier(n_estimators=100,
                                  max_depth=10,
                                  random_state=42,
                                  class_weight='balanced')

# 2. Melatih Model (Proses Belajar)
print("Sedang melatih model... (mungkin butuh waktu 1-2 menit)")
rf_model.fit(X_train, y_train)

# 3. Melakukan Prediksi (Ujian)
y_pred = rf_model.predict(X_test)

# 4. Evaluasi Hasil
print("\n--- HASIL EVALUASI MODEL ---")
print(f"Akurasi Model: {accuracy_score(y_test, y_pred)*100:.2f}%")
print("\nLaporan Klasifikasi:")
print(classification_report(y_test, y_pred))
```

Evaluasi Performa Model

Akurasi Tinggi

Model mencapai akurasi sebesar 98,23% pada data pengujian.

Matriks Recall

Berhasil mencapai 86% untuk kelas Bad Loan. Ini berarti model sangat sensitif dalam mendeteksi nasabah yang berisiko tinggi.

Analisis Confusion Matrix

Menunjukkan jumlah True Positives dan True Negatives yang dominan, membuktikan minimnya kesalahan prediksi.

```
--- HASIL EVALUASI MODEL ---
Akurasi Model: 98.23%

Laporan Klasifikasi:
              precision    recall  f1-score   support

           0       0.98      1.00      0.99     83063
           1       0.97      0.86      0.91     10194

    accuracy                           0.98     93257
   macro avg       0.98      0.93      0.95     93257
weighted avg       0.98      0.98      0.98     93257
```

Kesimpulan dan Rekomendasi Bisnis

Kesimpulan:
Model telah tervalidasi mampu memprediksi risiko gagal bayar dengan tingkat kepercayaan yang sangat tinggi.

Rekomendasi Strategis:

- Automated Approval: Gunakan model ini untuk menyetujui pinjaman nasabah Low Risk secara instan guna mempercepat layanan.
- Early Warning System: Lakukan peninjauan mendalam atau minta jaminan tambahan bagi nasabah yang terdeteksi High Risk oleh model.
- Monitoring Berkala: Melakukan pembaruan data setiap 6-12 bulan untuk menyesuaikan model dengan kondisi ekonomi terbaru.

Sumber referensi :

- Random Forest Explained: Klik disini
- Handling Imbalanced Data: klik disini

Link Submission (.zip) : Klik disini