HOAQ: JURNAL TEKNOLOGI INFORMASI, Volume 16 Nomor 2, Desember 2025, p241-246
p-ISSN: 2337-5280, e-ISSN: 26207427
DOI: https://doi.org/10.52972/hoaq.vol16no2

PENERAPAN ALGORITMA *DECISION TREE* UNTUK MEMPREDIKSI RISIKO KREDIT PADA NASABAH BANK

Marsianus Gerlian Eka Mbete[1], Nathanael Nyala Bintang[2], Victor Noviandus[3]

[1,2,3]Program Studi Teknik Informatika, Universitas Widya Dharma Pontianak
Jl. HOS Cokroaminoto, Pontianak – Kalimantan Barat, Indonesia
Email: [1]ekamarsianus3@gmail.com, [2]nathanaelnyalabintang@gmail.com, [3]victornoviandus@gmail.com

ABSTRAK

Risiko terjadinya kredit macet merupakan salah satu permasalahan utama yang dihadapi oleh lembaga keuangan dalam aktivitas pemberian pinjaman. Untuk mengurangi potensi risiko tersebut, diperlukan suatu sistem prediktif yang mampu mengidentifikasi calon debitur dengan tingkat risiko gagal bayar secara dini. Penelitian ini bertujuan untuk merancang dan membangun model prediksi risiko kredit pada nasabah bank dengan menerapkan algoritma *Decision Tree*. Data yang digunakan mencakup informasi historis nasabah, antara lain pendapatan, besaran cicilan, uang muka, usia, tagihan listrik dan telepon, keberadaan rekening tabungan, serta jangka waktu pinjaman. Metodologi penelitian meliputi tahap prapemrosesan data, pelatihan model *Decision Tree*, dan penerapan teknik pruning serta *sampling* guna mengatasi permasalahan *overfitting* dan ketidakseimbangan kelas data. Hasil yang diperoleh menunjukkan bahwa model yang dikembangkan mampu mengklasifikasikan risiko kredit secara akurat dan menyajikan hasil yang mudah diinterpretasikan. Penelitian ini diharapkan dapat memberikan kontribusi bagi lembaga keuangan dalam meningkatkan akurasi, efisiensi, dan objektivitas proses penilaian kelayakan kredit.

Kata kunci: *credit scoring*, *decision tree*, klasifikasi, *overfitting*, risiko kredit

ABSTRACT

Credit default risk is one of the major challenges faced by financial institutions in the loan disbursement process. To mitigate this risk, a predictive system is required to identify potential borrowers with a high risk of default at an early stage. This study aims to design and develop a credit risk prediction model for bank customers using the Decision Tree algorithm. The dataset used includes historical customer information such as income, installment amount, down payment, age, utility bills (electricity and telephone), savings account status, and loan term. The research methodology involves data preprocessing, model training using the Decision Tree algorithm, and the application of pruning and sampling techniques to address issues related to overfitting and class imbalance. The results demonstrate that the developed model is capable of accurately classifying credit risk and provides easily interpretable outcomes. This study is expected to contribute to enhancing the accuracy, efficiency, and objectivity of creditworthiness assessments in financial institutions.

Keywords: credit scoring, decision tree, classification, overfitting, credit risk

1. PENDAHULUAN

Kredit merupakan fasilitas penyediaan dana atau tagihan yang dipersamakan dengannya, yang diberikan berdasarkan suatu perjanjian atau kesepakatan pinjam-meminjam antara bank dan pihak lain. Dalam perjanjian tersebut, pihak peminjam memiliki kewajiban untuk melunasi utangnya dalam jangka waktu yang telah ditentukan, disertai dengan pembayaran bunga sesuai ketentuan yang berlaku [1]. Risiko terjadinya kredit macet menjadi salah satu tantangan utama dalam proses pemberian kredit oleh lembaga keuangan.

Prediksi terhadap status kredit diperlukan sebagai langkah preventif untuk meminimalkan risiko gagal bayar. Dengan melakukan prediksi secara akurat, lembaga keuangan dapat mengidentifikasi calon debitur yang berisiko tinggi dan mengambil tindakan mitigasi sebelum terjadinya tunggakan pembayaran yang telah melewati batas jatuh tempo.

Penelitian ini menerapkan algoritma *Decision Tree* dalam membangun model *credit scoring* guna memprediksi risiko kredit nasabah bank. Objek penelitian mencakup data historis nasabah seperti penghasilan, cicilan, uang muka, jumlah periode pinjaman, rekening tabungan, umur pemohon, serta tagihan telepon dan listrik. Studi sebelumnya menunjukkan bahwa variabel penghasilan merupakan salah satu faktor dominan dalam memprediksi risiko kredit, dengan tingkat akurasi model mencapai 79,57% [2].

Beragam metode telah digunakan dalam prediksi risiko kredit, antara *lain Decision Tree, Random Forest* [3], *Support Vector Machine* (SVM) [4], dan *Neural Network* [5]. Decision Tree sendiri dikenal karena

kemudahannya dalam pemahaman dan interpretasi, kemampuannya menangani data campuran tanpa normalisasi, serta efisiensinya dalam pelatihan model berskala sedang hingga besar. Meskipun demikian, metode ini memiliki kelemahan seperti kecenderungan terhadap *overfitting*, sensitivitas terhadap perubahan data, dan ketidakseimbangan kelas [6].

Dalam konteks ini, beberapa solusi dapat diterapkan untuk mengatasi kekurangan algoritma *Decision Tree*, seperti teknik *pruning*, *sampling* (*oversampling* dan *undersampling*), penerapan metode *ensemble* seperti Random Forest dan Gradient Boosting, serta penggunaan teknik validasi silang (*cross-validation*) untuk meningkatkan generalisasi model.

2. METODE PENELITIAN

Pengertian *Decision Tree*

Decision Tree (Pohon Keputusan) adalah sebuah algoritma dalam pemodelan prediktif yang memiliki struktur menyerupai pohon bercabang. Algoritma ini digunakan untuk membantu proses pengambilan keputusan atau melakukan prediksi berdasarkan data yang diberikan sebagai input. Dalam *Decision Tree*, setiap simpul internal atau cabang merepresentasikan suatu kondisi atau pertanyaan logis terhadap satu atribut (fitur) tertentu dari data. Cabang-cabang yang keluar dari simpul tersebut menunjukkan kemungkinan hasil dari kondisi yang diuji, misalnya jawaban "ya" atau "tidak". Proses ini berlangsung secara berurutan hingga mencapai bagian akhir dari pohon yang disebut daun (*leaf node*), yang menunjukkan hasil akhir berupa keputusan atau klasifikasi tertentu. Dengan pendekatan ini, *Decision Tree* mampu menyajikan proses pengambilan keputusan secara sistematis dan mudah dipahami, baik dalam kasus klasifikasi maupun prediksi nilai.

Selain kemampuannya dalam menyajikan proses pengambilan keputusan yang intuitif, *Decision Tree* juga memiliki dasar matematis yang kuat dalam menentukan pemisahan data terbaik di setiap cabangnya. Pemilihan atribut terbaik dilakukan dengan menghitung seberapa baik suatu atribut mampu membagi data menjadi kelompok yang lebih homogen. Untuk kasus klasifikasi, pemisahan ini biasanya dihitung menggunakan metrik informasi seperti *Information Gain*, *Gain Ratio*, atau *Gini Index*. Salah satu rumus yang paling umum digunakan adalah *Information Gain*, yang didasarkan pada konsep *Entropy*.

Rumus *Entropy* untuk suatu *dataset S* adalah:

$$\textbf{Entropy}\ (\boldsymbol{S}) = -\sum_{\boldsymbol{i=1}}^{\boldsymbol{n}} \boldsymbol{p_i}\ \mathbf{log_2}(\boldsymbol{p_i})$$

dimana:

1. p_i adalah proporsi data dalam kelas ke-i,
2. n adalah jumlah kelas.

Kemudian, **Information Gain (IG)** dari atribut A terhadap dataset S dihitung sebagai:

$$\textbf{Gain}(\boldsymbol{S},\boldsymbol{A}) = \textbf{Entropy}(\boldsymbol{S}) - \sum_{\boldsymbol{v}\in\textbf{Values}(\boldsymbol{A})} \frac{|\boldsymbol{S_v}|}{|\boldsymbol{S}|}\ .\ \textbf{Entropy}(\boldsymbol{S_v})$$

dimana:

1. Values(A) adalah himpunan nilai-nilai unik dari atribut A,
2. S_v adalah subset dari S yang memiliki nilai v pada atribut A.

Atribut dengan nilai *Information Gain* tertinggi akan dipilih sebagai cabang berikutnya dalam *Decision Tree*, sehingga secara iteratif terbentuk struktur pohon yang optimal dalam memisahkan data berdasarkan target output [7].

Prinsip kerja *Decision Tree*

Decision Tree adalah salah satu metode dalam *data mining* yang digunakan untuk tugas klasifikasi dan regresi. Cara kerjanya membentuk model berupa struktur pohon yang terdiri dari simpul (*node*) dan cabang (*branch*). Berikut adalah elemen-elemen utamanya:

1. *Root Node*
   Titik awal pohon yang mewakili seluruh data.
2. *Splitting*
   Proses membagi data berdasarkan atribut tertentu yang dianggap paling baik, biasanya dinilai menggunakan kriteria seperti information gain atau Gini index.
3. *Decision Node*
   Titik dalam pohon yang mewakili keputusan berdasarkan atribut yang telah dipilih.
4. *Leaf Node*
   Titik akhir dari pohon yang menunjukkan hasil akhir dari klasifikasi atau prediksi.
5. *Pruning*
   Proses memangkas cabang-cabang pohon yang tidak terlalu penting agar model tidak terlalu rumit dan lebih mampu melakukan generalisasi terhadap data baru [8].

Algoritma *Decision Tree*

Beberapa algoritma populer yang digunakan dalam pembuatan *Decision Tree* antara lain:

1. ID3 (Iterative Dichotomiser 3)
   Menggunakan *information gain* untuk memilih atribut terbaik pada setiap proses pemisahan data [9].
2. C4.5
   Pengembangan dari ID3 yang bisa menangani atribut kontinyu dan data yang hilang [10].
3. C5.0
   Versi lanjutan dari C4.5 dengan keunggulan dalam kecepatan dan efisiensi penggunaan memori [11].
4. CART (*Classification and Regression Trees*)
   Menggunakan Gini *index* untuk klasifikasi dan mean squared error untuk regresi, serta selalu membentuk pohon biner [12].

Pengertian *Random Forest*

Random Forest merupakan algoritma *ensemble learning* yang digunakan untuk tugas klasifikasi maupun regresi. Algoritma ini terdiri atas sejumlah *Decision Tree* yang dilatih secara independen. Setiap pohon dibentuk berdasarkan subset data yang diambil secara acak dengan pengembalian (*bootstrap sampling*), serta menggunakan subset fitur acak saat pemisahan di tiap *node*. Hasil prediksi akhir diperoleh melalui mekanisme voting mayoritas untuk klasifikasi atau rata-rata prediksi untuk regresi.

Prinsip kerja *Random Forest*

Prinsip kerja algoritma Random Forest dapat dijelaskan melalui beberapa tahap berikut:

1. *Bootstrap Aggregating* (Bagging)
   Dari keseluruhan data berjumlah N, diambil sejumlah N sampel secara acak dengan pengembalian, sehingga terbentuk beberapa subset data yang berbeda untuk melatih masing-masing pohon.
2. *Random Feature Subspace*
   Pada setiap proses pemilihan fitur di *node*, hanya dipertimbangkan sejumlah k fitur yang dipilih secara acak dari keseluruhan fitur yang tersedia. Pendekatan ini bertujuan untuk mengurangi korelasi antar pohon, sehingga meningkatkan keragaman model.
3. Pertumbuhan Tree
   Setiap pohon tumbuh hingga mencapai kedalaman maksimum tanpa dilakukan proses pemangkasan (unpruned), dengan tujuan mengoptimalkan pemisahan data pada tahap pelatihan.
4. Agregasi Output
   Dalam konteks klasifikasi, hasil akhir ditentukan berdasarkan kelas yang memperoleh suara terbanyak dari seluruh pohon (*majority voting*), sementara dalam regresi, digunakan nilai rata-rata dari seluruh prediksi pohon. Proses ini menghasilkan model yang lebih stabil dan akurat dibandingkan dengan model pohon tunggal.
5. *Estimasi Error Internal* dan *Importance* Fitur
   Karena setiap pohon tidak menggunakan sekitar sepertiga dari data pelatihan (disebut *out-of-bag* atau OOB), data ini dapat dimanfaatkan untuk memperkirakan galat model secara internal. Selain itu, *Random Forest* juga mampu memberikan estimasi mengenai kontribusi atau tingkat kepentingan masing-masing fitur terhadap hasil prediksi.

Kelebihan dan kekurangan *Random Forest*

Berikut adalah kelebihan dari metode *Random Forest*:

1. Memiliki tingkat akurasi yang tinggi karena mampu mengurangi varians model [13].
2. Relatif tahan terhadap *overfitting* dibandingkan model individual [14].
3. Dapat bekerja secara efektif pada data berdimensi tinggi maupun data yang mengandung *outlier*.
4. Mampu mengukur tingkat kepentingan fitur secara efisien.

Namun, metode *Random Forest* memiliki beberapa kelemahan. Berikut adalah kekurangan dari metode *Random Forest*:

1. Kurang interpretabel dibanding *Decision Tree* tunggal.
2. Konsumsi memori dan waktu komputasi lebih tinggi, terutama untuk banyak pohon.

Data Preprocessing

Data preprocessing merupakan tahap krusial dalam proses pemodelan data, yang bertujuan untuk mempersiapkan data mentah agar layak digunakan dalam proses analisis dan pembelajaran mesin. Tahapan ini meliputi kegiatan pembersihan data (*data cleaning*), seperti penanganan nilai hilang, penghapusan data duplikat, dan koreksi inkonsistensi; transformasi data (*data transformation*), yang mencakup normalisasi untuk menyamakan skala antar fitur numerik—misalnya dengan pendekatan *Min-Max Normalization* menggunakan rumus:

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}};$$

serta *encoding* data kategorikal, seperti *label encoding* atau *one-hot encoding*, agar fitur non-numerik dapat direpresentasikan dalam bentuk numerik. Selain itu, dilakukan pula seleksi fitur (*feature selection*) untuk mengidentifikasi atribut-atribut yang memiliki pengaruh signifikan terhadap variabel target, dengan mempertimbangkan nilai *feature importance* berdasarkan kontribusinya dalam mengurangi impurity model. Penerapan preprocessing yang tepat tidak hanya meningkatkan kualitas data, tetapi juga berkontribusi terhadap peningkatan akurasi, efisiensi pelatihan model, serta kemampuan generalisasi sistem dalam melakukan klasifikasi atau prediksi.

Normalisasi Data

Normalisasi bertujuan untuk menyamakan skala antar fitur numerik agar tidak terjadi dominasi oleh fitur dengan skala yang lebih besar. Salah satu metode normalisasi yang digunakan adalah *Min-Max Normalization*, dengan rumus sebagai berikut:

$$\boldsymbol{X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}}$$

Dimana:

1. X adalah nilai asli.
2. $X_{\min}$ dan $X_{\max}$ masing-masing adalah nilai minimum dan maksimum dari fitur tersebut.

Dengan normalisasi ini, nilai setiap fitur akan berada dalam rentang [0,1][0,1][0,1].

Encoding Data Kategorikal

Fitur kategorikal seperti status akun atau jenis pinjaman tidak bisa langsung digunakan dalam algoritma numerik. Oleh karena itu, dilakukan encoding terhadap data kategorikal tersebut. Salah satu teknik yang digunakan adalah Label *Encoding* atau *One-Hot Encoding*, tergantung pada konteks. Sebagai contoh:

Tabel 1. *Encoding* data kategorikal

Credit status	*Credit encoded*
Good	1
Bad	0

Teknik ini memungkinkan data kategorikal untuk direpresentasikan dalam bentuk numerik agar dapat diproses oleh algoritma *Decision Tree*.

Feature Importance

Setelah model dilatih, penting untuk mengevaluasi fitur mana yang paling berpengaruh dalam proses klasifikasi. *Feature importance* dapat dihitung berdasarkan kontribusi fitur dalam menurunkan nilai *impurity* (seperti Gini atau Entropy) pada setiap percabangan pohon. Dalam *Random Forest*, skor *feature importance* dihitung dengan:

$$\textbf{Importance}(\boldsymbol{f}) = \sum_{\boldsymbol{t}} \Delta \boldsymbol{i_t}(\boldsymbol{f}) \; . \; \frac{\boldsymbol{N_t}}{\boldsymbol{N}}$$

Dimana:

1. $\Delta i_t(f)$ adalah pengurangan impurity yang disebabkan oleh fitur f pada node t.
2. N_t adalah jumlah sampel pada node t.
3. N adalah total sampel pada seluruh data pelatihan.

Fitur dengan nilai *importance* lebih tinggi dianggap lebih relevan dalam proses klasifikasi risiko kredit.

3. HASIL DAN PEMBAHASAN

Akurasi model

Tabel berikut menyajikan hasil evaluasi kinerja dari dua model klasifikasi:

Tabel 2. Perbandingan kinerja model

Model	Akurasi	*Precision* (Macro)	*Recall* (Macro)	F1-*Score* (Macro)
Decision Tree	66,5%	0,602	0,605	0,603
Random Forest	80,5%	0,784	0,719	0,738

Interpretasi hasil

Model *Decision Tree* menunjukkan performa yang cukup baik dalam mengklasifikasikan kelas mayoritas (kredit lancar). Namun demikian, kinerjanya menurun secara signifikan dalam mendeteksi kelas minoritas (kredit macet). Sebaliknya, model *Random Forest* mampu mengatasi kelemahan ini, ditunjukkan oleh nilai *recall* yang lebih tinggi terhadap kelas minoritas. Meskipun demikian, model Random Forest memiliki tingkat kompleksitas yang lebih tinggi dan tidak seinterpretatif *Decision Tree* dalam hal penjelasan hasil prediksi.

Analisis fitur penting

Berdasarkan analisis pentingnya fitur (*feature importance*) yang diperoleh dari model *Random Forest*, variabel-variabel yang paling berpengaruh dalam proses prediksi adalah sebagai berikut:

1. Credit amount (jumlah kredit).
2. *Status of existing checking account* (status akun pemeriksaan yang ada).
3. *Duration* (durasi).
4. *Age* (usia).
5. *Loan purpose* (tujuan peminjaman).

4. SIMPULAN

Risiko kredit macet adalah tantangan signifikan bagi lembaga keuangan dalam proses pemberian pinjaman, yang memerlukan sistem prediktif untuk mengidentifikasi calon debitur berisiko tinggi. Penelitian ini bertujuan untuk merancang model prediksi risiko kredit menggunakan algoritma *Decision Tree*, dengan memanfaatkan data historis nasabah. Proses penelitian meliputi prapemrosesan data, pelatihan model, serta penerapan teknik pruning dan *sampling* untuk mengatasi masalah *overfitting* dan ketidakseimbangan kelas. Model *Decision Tree* mencapai akurasi 66,5%, sedangkan *Random Forest* mencapai 80,5%. Model *Random Forest* lebih efektif dalam mendeteksi kredit macet dibandingkan *Decision Tree*. Variabel penting dalam prediksi risiko kredit meliputi jumlah kredit, status akun pemeriksaan, durasi pinjaman, usia, dan tujuan peminjaman. Penelitian ini diharapkan dapat meningkatkan akurasi dan objektivitas dalam penilaian kelayakan kredit di lembaga keuangan. Disarankan untuk menggunakan model yang lebih kompleks seperti *Random Forest* untuk meningkatkan akurasi, sambil mempertimbangkan interpretabilitas model. Penerapan algoritma *Decision Tree* dan *Random Forest* menunjukkan potensi besar dalam memprediksi risiko kredit, dengan perhatian pada keseimbangan antara akurasi dan interpretabilitas.

DAFTAR PUSTAKA

[1] Kementrian Keuangan, "Undang-Undang Republik Indonesia Nomor 10 Tahun 1998 Tentang Perubahan Atas Undang-Undang Nomor 7 Tahun 1992 Tentang Perbankan," *Lembaran Negara Republik Indonesia*, p. pasal 1 ayat 2, 1998, [Online]. Available: http://www.bphn.go.id/data/documents/98uu010.pdf.

[2] Suryadi, Hozeng, and S. K. Aisa, "Aplikasi Data Mining dengan Menggunakan Metode Decision Tree Untuk Prediksi Penentuan Resiko Kredit,", *SISITI : Seminar Ilmiah Sistem Informasi Dan Teknologi Informasi*, vol. V, no. 2, pp. 1–10, Aug. 2016, doi: 10.36774/sisiti.v5i2.159.

[3] M. F. R. Aditya, N. Lutvi, and U. Indahyanti, "Prediksi Penyakit Hipertensi Menggunakan Metode Decison Tree dan Random Forest," *Jurnal Ilmu Komputasi*, vol. 23, no. 1, pp. 9–16, 2024, doi: 10.32409/jikstik.23.1.3503.

[4] I. Siti Aisah, B. Irawan, and T. Suprapti, "Algoritma Support Vector Machine (SVM) Untuk Analisis Sentimen Ulasan Aplikasi Al Qur'an Digital," *JATI (Jurnal Mahasiswa Teknik Informatika).*, vol. 7, no. 6, pp. 3759–3765, 2024, doi: 10.36040/jati.v7i6.8263.

[5] Y. D. Pristanti and F. Windana, "Pengembangan Metode Neural Networks untuk Menentukan Karakter Seseorang," *Jurnal STT STIKMA Internasional.*, vol. 6, no. 1, pp. 9–27, 2015. Available [online]: https://jurnal.stikma.ac.id/index.php/jssi/article/view/11.

[6] M. R. Qisthiano, P. A. Prayesy, and I. Ruswita, "Penerapan Algoritma Decision Tree dalam Klasifikasi Data Prediksi Kelulusan Mahasiswa", J*urnal Teknologi Terapan G-Tech*, vol. 7, no. 1, pp. 21–28, Jan. 2023, doi: 10.33379/gtech.v7i1.1850.

[7] A. Febriani and V. Anggraini, "Implementasi Algoritma Decision Tree (J.48) untuk Memprediksi Resiko Kredit pada BMT," *Tekinfo Jurnal Ilmiah Teknik Industri dan Informasi*, vol. 9, no. 2, pp. 91–99, 2021, doi: 10.31001/tekinfo.v9i2.904.

[8] C. N. Syahputri, M. S. Hasibuan, "Optimasi Klasifikasi Decision Tree dengan Teknik Pruning Untuk Mengurangi Overfitting," *JSiI (Jurnal Sistem Informasi)*, vol. 11, no. 2, pp. 87–96, 2024, doi: 10.30656/jsii.v11i2.9161.

[9] K. Wang, L. Wang, and J. Sun, "The Data Analysis of Sports Training by ID3 Decision Tree Algorithm and Deep Learning," *Scientific Reports.*, vol. 15, no. 1, pp. 1–11, 2025, doi: 10.1038/s41598-025-99996-5.

[10] P. Khosravi, A. Vergari, Y. Choi, Y. Liang, and G. Van den Broeck, "Handling Missing Data in Decision Trees: A Probabilistic Approach," 2020, [Online]. Available: http://arxiv.org/abs/2006.16341

[11] R. Pandya and J. Pandya, "C5. 0 Algorithm to Improved Decision Tree with Feature Selection and Reduced Error Pruning," *International Journal of Computer Applications*, vol. 117, no. 16, pp. 18–21, 2015, doi: 10.5120/20639-3318.

[12] J. M. Klusowski, "Sparse learning with CART," *Conference on Neural Information Processing Systems (NeurIPS 2020)*, pp. 1-22, Nov. 2020. doi: 10.48550/arXiv.2006.04266

[13] V. M. Herrera, T. M. Khoshgoftaar, F. Villanustre, and B. Furht, *Random forest implementation and optimization for Big Data analytics on LexisNexis's high performance computing cluster platform*, vol.

6, no. 1. Springer International Publishing, 2019. doi: 10.1186/s40537-019-0232-1.

[14] D. Bratić, P. Miljković, D. Jurečić, and T. Grabarić, "AI-Driven Random Forest Model and the Six Sigma Approach for Enhancing Offset Printing Process and Product Quality," *Applied Sciences*, vol. 15, no. 5, pp. 2266-2299, 2025, doi: 10.3390/app15052266.