



Penerapan Metode Stacking Ensemble Untuk Klasifikasi Status Pinjaman Nasabah Bank

Mayesq Prameswari¹, Putri Emas Kania², I Gusti De Ayu³, Sri Namira Putri Harnoko⁴

^{1, 2, 3, 4}Program Studi Sains Data, Institut Teknologi Telkom Purwokerto

²2211110003@ittelkom-pwt.ac.id

³2211110018@ittelkom-pwt.ac.id

⁴2211110020@ittelkom-pwt.ac.id

Corresponding author email: ¹2211110002@ittelkom-pwt.ac.id

Abstract: Bank loan status prediction is a crucial aspect of the financial industry, as it determines an applicant's eligibility to receive a loan. Banks only lend to customers who can reliably repay their loans. This research aims to develop a model for predicting bank loan status using stacking machine learning algorithms. Historical borrower data is used to identify patterns related to the likelihood of an applicant's loan being approved or rejected. In this research, the Stacking Ensemble Learning technique is employed, which combines various machine learning models to improve prediction accuracy. Five algorithms are used: Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, and Random Forest. Each algorithm functions as both a meta-learner and an individual base model. The results show that the stacking ensemble model combination can predict more accurately than individual models. However, the success of stacking techniques in improving accuracy depends on the combination of individual models selected and the type of meta-learner used. For instance, while KNN achieved an individual model accuracy of 82.56%, its accuracy dropped to 81.40% after being used as a meta-learner in the stacking ensemble. On the other hand, in this research, combining the Random Forest meta-learner with other individual models achieved the highest accuracy of 90.7%.

Keywords: stacking, ensemble learning, machine learning, loan status, prediction

Abstrak: Prediksi status pinjaman bank merupakan aspek krusial dalam industri keuangan yang menentukan kelayakan seorang pemohon untuk menerima pinjaman. Bank hanya akan memberikan pinjaman kepada nasabah yang benar-benar dapat membayar kembali pinjamannya. Tujuan penelitian ini mengembangkan model prediksi status pinjaman bank menggunakan algoritma pembelajaran mesin. Data historis peminjam digunakan untuk mengidentifikasi pola yang berhubungan dengan kemungkinan pinjaman pemohon disetujui atau ditolak. Metode yang digunakan dalam penelitian ini yaitu Stacking Ensemble Learning, yaitu teknik yang menggabungkan berbagai model machine learning untuk meningkatkan akurasi model prediksi. Algoritma yang digunakan ada 5 yaitu Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, dan Random Forest. Cara kerjanya, setiap algoritma difungsikan sebagai meta learner dan pada saat yang sama lainnya sebagai base learner (model individu). Hasil menunjukkan bahwa kombinasi model stacking ensemble mampu memprediksi lebih baik dibandingkan model individu. Namun, keberhasilan teknik stacking dalam meningkatkan akurasi bergantung pada kombinasi model individu yang dipilih serta jenis meta learner yang digunakan. Ini ditunjukkan oleh KNN sebagai model individu mencapai akurasi 82,56%, tetapi setelah stacking dan menjadi meta learnernya justru turun akurasinya menjadi 81,40%. Pada penelitian ini, kombinasi meta learner Random Forest dan algoritma lain sebagai model individu mencapai akurasi tertinggi yaitu 90,7%.

Kata kunci: stacking, ensemble learning, machine learning, status pinjaman, prediksi

I. PENDAHULUAN

Prediksi status pinjaman bank merupakan aspek krusial dalam industri keuangan yang menentukan kelayakan seorang pemohon untuk menerima pinjaman. Proses ini penting untuk mengurangi risiko gagal bayar yang dapat mengakibatkan kerugian finansial bagi bank [1]. Kemajuan teknologi kecerdasan buatan terutama algoritma pembelajaran mesin memberikan cara baru untuk membantu prediksi yang akurat dan cepat [2]. Kemajuan teknologi tersebut menawarkan potensi proses penilaian dan persetujuan pinjaman yang obyektif dan lebih akurat berdasarkan pelajaran dari data-data masa lalu.

Penerapan pembelajaran mesin dalam prediksi pinjaman kredit menawarkan berbagai keuntungan. Pertama, ini memungkinkan analisis data dalam jumlah besar secara cepat dan akurat tanpa adanya potensi kesalahan manusia. Kedua, sistem berbasis mesin dapat diperbarui dan ditingkatkan secara



berkala dengan data baru, sehingga model prediksi tetap relevan dan efektif seiring berjalannya waktu. Namun, meskipun potensi manfaatnya besar, ada beberapa tantangan yang harus diatasi. Salah satu tantangan utama adalah memastikan bahwa data yang digunakan untuk melatih model adalah representatif dan bebas dari bias yang dapat menyebabkan keputusan yang tidak adil atau tidak akurat. Selain itu, pemilihan dan pengoptimalan algoritma yang tepat sangat penting untuk mencapai kinerja prediktif yang optimal [3]. Dalam domain prediksi status pinjaman atau kredit menggunakan *machine learning*, penelitian [4] membandingkan kinerja Decision Tree dan Random Forest dalam memprediksi probabilitas gagal bayar pinjaman berdasarkan profil peminjam. Penelitian tersebut menemukan bahwa Random Forest Classifier mengungguli metode Decision Tree dengan akurasi 80% berbanding 73%. Namun, terbatasnya kasus gagal bayar dalam dataset menunjukkan perlunya data yang lebih baru untuk meningkatkan akurasi model.

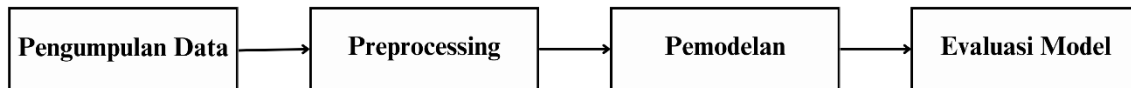
Namun penelitian-penelitian terkini menunjukkan bahwa salah satu metode mengoptimalkan machine learning adalah *stacking* [5]. Pada penelitian [6] menunjukkan bahwa penggunaan *stacking ensemble learning* secara signifikan meningkatkan kinerja model prediksi pada kelas risiko kredit. Penelitian tersebut mengintegrasikan beberapa model dasar (XGBoost, Light Gradient Boosting Machine, Gradient Boosting DTREE, AdaBoost) dan menggunakan Random Forest sebagai *meta-learner*, yang terbukti meningkatkan akurasi, presisi, *recall*, dan *F1-score* dibandingkan dengan pengklasifikasi tunggal. Penelitian lalu [7] memperkirakan risiko gagal bayar pinjaman dengan menggunakan berbagai algoritma *Machine Learning* (ML) dan *Deep Learning* (DL) disektor perbankan dengan tujuan meminimalisir kerugian finansial. Algoritma yang dibandingkan antara lain Logistic Regression, Decision Tree, Random Forest, Extra Trees, Support Vector Machine, K-Nearest Neighbors, Gaussian Naive Bayes, AdaBoost, Gradient Boosting, dan deep learning (Long Short Term Memory), hasilnya Xtra Tree mencapai akurasi paling tinggi, mencapai 87,26%. Akurasi tersebut sudah melibatkan penerapan tahapan pembersihan data, mengatasi data yang tidak seimbang, dan penanganan *outlier*. Penelitian menemukan kesimpulan bahwa keterbatasan seperti teknik penyeimbangan data yang belum teruji dan menyarankan penelitian di masa depan untuk mengeksplorasi metode pembelajaran tanpa pengawasan.

Berdasarkan beberapa penelitian sebelumnya [1], [2], [3], [6], [7], [8], penelitian ini bertujuan mengembangkan model prediksi status pinjaman bank dengan metode *stacking* menggunakan algoritma pembelajaran mesin. Variabel yang digunakan adalah data historis peminjam, seperti riwayat kredit, pendapatan, dan faktor demografis lainnya. Sistem ini diharapkan dapat mengidentifikasi pola dan indikator yang berhubungan dengan kemungkinan pinjaman pemohon disetujui atau ditolak. Penelitian ini akan mengeksplorasi berbagai algoritma pembelajaran mesin, seperti K-Nearest Neighbors (KNN), Support Vector Machine, Decision Tree, Logistic Regression, dan Random Forest. Berdasarkan lima model tersebut yang akan dijadikan sebagai *base model*, pendekatan *stacking ensemble learning* pun dilakukan dengan tujuan memprediksi status pinjaman bank. Menurut penelitian [8], *preprocessing data* seperti penanganan nilai yang hilang, *encoding* variabel kategorikal, *outliers*, dan keseimbangan data akan diterapkan untuk meningkatkan akurasi model. Maka penelitian ini juga menggunakan pendekatan pada penelitian [8], peneliti berharap dapat menyediakan alat yang lebih efisien dan akurat bagi bank dalam proses pemberian pinjaman, yang pada akhirnya dapat meningkatkan stabilitas finansial dan kepercayaan dalam sistem perbankan.



II. METODOLOGI PENELITIAN

Penelitian ini berfokus pada pembuatan model prediksi dengan algoritma klasifikasi pada status persetujuan pinjaman bank berdasarkan fitur-fitur independen (prediktor) dalam dataset dengan model yang dipilih berdasarkan karakteristik data dan tujuan analisis. Metodologi penelitian ini mencakup beberapa tahapan yang ditunjukkan oleh Gambar 1.



Gambar 1. Alur Penelitian

Gambar 1. menunjukkan alur penelitian yang terdiri dari empat tahapan utama. Pertama, pengumpulan data dilakukan dengan mengunduh dataset Loan Bank dari Kaggle yang mencakup informasi terstruktur mengenai pinjaman bank. Tahap kedua adalah *preprocessing data*, di mana dataset disiapkan untuk analisis dengan langkah-langkah seperti pembersihan data untuk mengatasi *missing value* dan *outlier*, serta transformasi data jika diperlukan seperti *encoding* variabel kategori. Tahap ketiga melibatkan pelatihan model, di mana model *machine learning* dipilih dan dilatih menggunakan dataset yang telah diproses sebelumnya. Pada penelitian ini, model akan dipilih berdasarkan karakteristik data dan tujuan analisis, dengan fokus pada prediksi status persetujuan pinjaman (Loan_Status) berdasarkan fitur-fitur lain dalam dataset. Tahap terakhir adalah evaluasi model, di mana kinerja model dievaluasi menggunakan metrik yang sesuai, seperti akurasi, presisi, *recall*, dan *F1-score*. Evaluasi ini penting untuk menentukan seberapa baik model dapat memprediksi status persetujuan pinjaman, serta untuk menilai kecocokan model terhadap data yang digunakan dalam penelitian ini.

2.1. Pengumpulan Dataset

Dataset yang digunakan dalam penelitian ini adalah data Loan Bank yang diperoleh dari *platform Kaggle* (kaggle.com/datasets/bhavikjickadara/loan-status-prediction/data). Dataset ini terdiri dari informasi yang terstruktur mengenai pinjaman bank, seperti variabel-variabel yang dijelaskan pada Tabel 1.

Tabel 1. Variabel dalam Dataset

No.	Nama Atribut	Kriteria
1	Loan_ID	Unique load id
2	Gender	Male/ Female
3	Married	True/ False
4	Dependent	0 / 1/ Other
5	Education	Graduate/Not Graduate
6	Self_Employed	True/ False
7	ApplicantIncome	Pendapatan bulanan pemohon
8	CoapplicantIncome	Pendapatan bulanan (rekan pemohon)
9	LoanAmount	Jumlah Pinjaman
10	Loan_Amount_Term	Durasi Pinjaman (bulan)
11	Credit History	0: Bad History 1: Good History
12	Property_Area	Urban/Rural/Semi Urban
13	Loan_Status	No / Yes

Dataset penelitian ini dipilih karena relevansinya terhadap variabel-variabel yang berhubungan dengan faktor-faktor yang menjadi pertimbangan persetujuan pinjaman bank. Dataset berjumlah 381



baris dengan 13 kolom. Dengan memanfaatkan dataset ini, diharapkan dapat menghasilkan model prediksi yang relevan dalam konteks analisis risiko kredit dan keputusan pemberian pinjaman di institusi keuangan.

2.2. Preprocessing

Pada bagian *preprocessing*, terdapat langkah-langkah krusial seperti penanganan *missing value*, penanganan *outliers*, *encoding*, penanganan *imbalanced data*, dan normalisasi data.

2.2.1. Handling Missing Value

Missing value menunjukkan data yang tidak lengkap. Bisa jadi dalam satu baris ada satu atau dua variabel yang hilang. Nilai hilang ditunjukkan oleh nilai NaN atau *Null* [9]. Pada penelitian ini, penanganan nilai *null* melibatkan serangkaian langkah untuk mengatasi nilai yang hilang dalam dataset. Terdapat variabel seperti Credit History dan Dependents yang awalnya berupa angka saja, perlu dikembalikan *value* kategorinya dengan penambahan nilai *string* untuk mencerminkan sifat kategorikalnya.

Pada proses mencari *missing value*, ditemukan *missing value* pada variabel Gender sebanyak 5 nilai *null*, Dependents sebanyak 8 nilai *null*, Self_Employed sebanyak 21 nilai *null*, Loan_Amount_Term sebanyak 11 nilai *null*, dan Credit_History sebanyak 30 nilai *null*. Pendekatan yang digunakan meliputi pengisian nilai rata-rata (mean) untuk variabel Loan_Amount_Term, serta penggunaan nilai modus untuk Self_Employed dan Credit History. Sisa baris data yang masih memiliki *missing value* pada variabel lainnya dihapus untuk memastikan konsistensi data yang digunakan dalam analisis.

2.2.2. Handling Imbalanced Data dengan Oversampling

Dataset yang digunakan pada penelitian ini mengalami *imbalanced data*, dengan proporsi kelas True (pinjaman disetujui) pada Gambar 1. Untuk mengatasi ketidakseimbangan kelas pada variabel target, dilakukan teknik *balancing data* dengan teknik SMOTE (*Synthetic Minority Over-sampling Technique*). Ketidakseimbangan kelas terjadi ketika jumlah sampel dari setiap kelas tidak seimbang antara kelas mayoritas dan minoritas [10]. Ketika ketidakseimbangan kelas tidak diatasi, model dapat memiliki bias yang signifikan terhadap kelas mayoritas, lalu mengakibatkan kurang sensitif terhadap kasus-kasus yang penting dalam kelas minoritas. Meskipun akurasi model bisa tinggi karena dominasi kelas mayoritas, prediksi untuk kelas minoritas akan menjadi sangat rendah [4]. Rumus pada teknik SMOTE yang akan digunakan seperti persamaan (1).

$$Synthetic = x_i + \lambda(x_j - x_i) \quad (1)$$

Di mana:

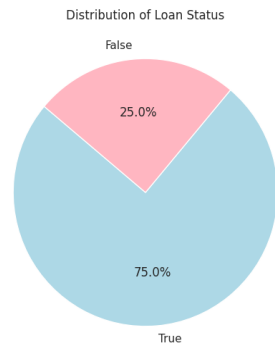
Synthetic : Data Sintesis

x_i : Data asli yang akan direplikasi

x_j : Data yang memiliki jarak terdekat dengan data x_i

λ : Nilai random antara 0 dan 1

Data sintesis yang dibuat untuk mengatasi ketimpangan kelas dengan cara menghasilkan data baru di antara data minoritas yang ada. x_i adalah data minoritas yang akan direplikasi, x_j adalah salah satu tetangga terdekat dari x_i , dan λ adalah nilai acak yang dipilih antara 0 dan 1. Metode ini bertujuan untuk meningkatkan jumlah sampel dalam kelas minoritas dengan menciptakan sampel sintetis berdasarkan perbedaan antara x_i dan x_j yang akan membantu meningkatkan keseimbangan dataset tanpa *overfitting* pada data latih [11].



Gambar 2. *Imbalanced* Loan Status

Teknik SMOTE membantu False (pinjaman tidak disetujui) agar seimbang dengan True (pinjaman disetujui) yang awalnya didominasi 75% oleh True menjadi 50% untuk masing-masing False dan True. Keseimbangan data akan memastikan bahwa model *machine learning* yang dibangun lebih adil dan akurat dalam memprediksi berbagai kelas dalam data.

2.2.3. Normalisasi Data

Tahapan selanjutnya dalam *preprocessing* adalah normalisasi data. Teknik yang digunakan adalah *scaling* dengan teknik MinMaxScaler. *Scaling* fitur-fitur numerik bertujuan untuk mengubah nilai-nilai fitur pada rentang yang sama [0-1]. Hal ini berguna untuk membuat model tahan pada pencilan. Rumus *scaling* menggunakan persamaan (2).

$$X_{scaled} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (2)$$

Di mana:

X = nilai asli dari fitur

X_{min} = nilai minimum dari fitur tersebut dalam dataset

X_{max} = nilai maksimum dari fitur tersebut dalam dataset

Proses ini membantu meningkatkan konsistensi performa model dengan memastikan bahwa perbedaan skala antar fitur-fitur tidak mempengaruhi hasil akhir. Misalnya, jika sebuah fitur memiliki rentang nilai yang sangat luas dibandingkan dengan fitur lainnya, normalisasi ini dapat membantu dalam menyeimbangkan kontribusi masing-masing fitur terhadap model. Penerapan teknik MinMaxScaler membuat nilai-nilai pada fitur-fitur numerik akan diubah sedemikian rupa sehingga nilai minimum setiap fitur akan menjadi 0, dan nilai maksimumnya menjadi 1, dengan nilai-nilai lainnya tersebar proporsional di antara kedua nilai tersebut [11], [12].

2.2.4. Outliers

Setelah identifikasi nilai *outliers*, metode Interquartile Range (IQR) digunakan untuk mengatasi masalah *outlier* pada variabel ApplicantIncome sebanyak 13 *outliers*, CoapplicantIncome sebanyak 5 *outliers*, LoanAmount sebanyak 7 *outliers*, dan Loan_Amount_Term sebanyak 66 *outliers*. Dengan menggunakan IQR, batas bawah dan batas atas ditentukan untuk mendeteksi *outliers*, dan nilai-nilai yang berada di luar batas ini ditangani sesuai dengan prosedur yang telah ditetapkan, seperti penghapusan atau transformasi data.

2.2.5. Encoding

Selanjutnya, tahap *encoding* variabel kategorikal yang mencakup proses transformasi variabel kategori menggunakan metode One-Hot Encoding bertipe *boolean*. Dalam langkah ini, semua variabel kategori, termasuk variabel target, diubah menjadi variabel biner (*dummy variables*) dengan format *boolean* (False dan True). Setelah itu menghapus variabel Gender_Female, Married_No, Education_Not Graduate, Self_Employed_No, Loan_Status_N, Credit_History_Bad agar tidak menyebabkan terjadinya redundansi data. Berikut tabel data setelah dilakukan *encoding* yang telah dikembalikan nama variabelnya seperti pada Tabel 2.

Tabel 2. Hasil One-Hot Encoding

Applicant Income	Coapplicant Income	..	Gender	Married	Property_Area_Semiurban	Property_Area_Urban	Loan_Status	..
4583	1508.0	..	True	True	False	False	False	..
3000	0.0	..	True	True	False	True	True	..
..

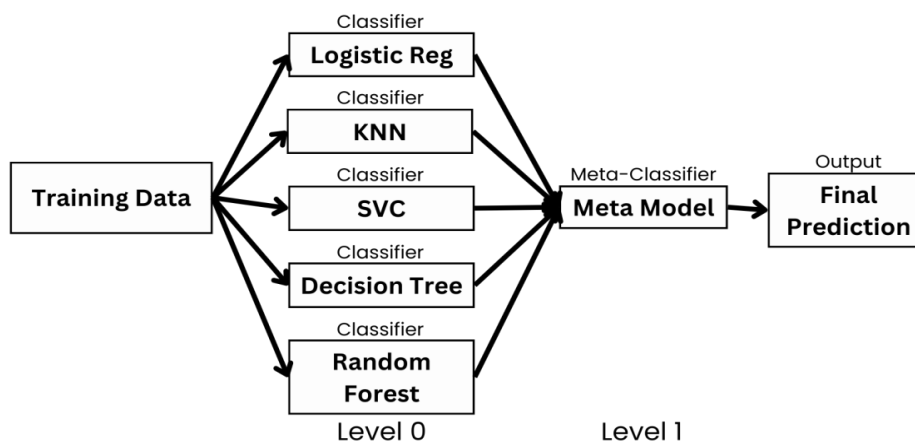
Terdapat variabel baru yang tidak dihapus karena memiliki lebih dari 2 kategori, seperti Dependents_0 Dependent, Dependents_1 Dependent, Dependents_2 Dependents, Dependents_3+ Dependents, Property_Area_Rural, Property_Area_Semiurban, dan Property_Area_Urban. Transformasi ini memastikan bahwa model *machine learning* dapat menginterpretasikan dan memproses variabel kategori secara efektif, sehingga meningkatkan akurasi dan kinerja model dalam analisis data.

2.3. Modeling

Tahapan pembangunan model *machine learning* untuk memprediksi status persetujuan pinjaman menggunakan lima algoritma *machine learning* sebagai *base model*, yaitu Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Decision Tree, dan Random Forest. Model-model tersebut dilatih menggunakan data training yang telah diproses dan dibersihkan sebelumnya.

2.3.1. Stacking Ensemble

Pendekatan *stacking ensemble* diterapkan dengan cara menggabungkan prediksi dari setiap *output base model* untuk membangun model meta. Kemudian, *output* dari *base model* tadi menjadi *input* pada *meta model* yang selanjutnya akan melakukan prediksi akhir. Teknik *stacking* ini ditunjukkan ilustrasi Gambar 3.



Gambar 3. Konsep *Stacking Ensemble*



Penelitian ini akan menguji semua algoritma *base model* yang akan dijadikan *meta learner* dalam penerapan *stacking ensemble*. Pemilihan lima *base model* tersebut berdasarkan pada keinginan untuk menggabungkan model-model dengan karakteristik yang berbeda dan kekuatan yang beragam, sehingga dapat memberikan prediksi yang lebih akurat. Berikut adalah alasan mengapa setiap model dipilih sebagai *base model*:

- Algoritma Logistic Regression adalah metode klasifikasi yang digunakan untuk memperkirakan probabilitas suatu kejadian. Algoritma ini menerapkan fungsi logit untuk memodelkan hubungan antara variabel bebas dan variabel dependen biner, sehingga mengubah hasil regresi linear menjadi probabilitas dalam rentang 0 hingga 1 [13]
- K-Nearest Neighbors (KNN) adalah algoritma yang tidak membutuhkan asumsi mengenai distribusi data dan digunakan dalam klasifikasi dan regresi. Dalam klasifikasi, algoritma ini mengklasifikasikan titik data baru berdasarkan mayoritas kelas dari K tetangga terdekat [9], [14].
- Support Vector Machine (SVM) adalah algoritma klasifikasi yang mencari *hyperplane* optimal untuk memisahkan data dari dua kelas dengan margin terbesar. Algoritma ini menggunakan titik data yang paling dekat dengan *hyperplane*, yang disebut *support vectors*, untuk menentukan posisi *hyperplane*. Untuk data non-linear, SVM menggunakan teknik kernel [15].
- Decision Tree adalah model prediktif berbentuk pohon yang memecah data ke dalam bagian-bagian homogen berdasarkan atribut-atributnya, dengan setiap node menguji atribut untuk membagi data dan setiap daun mewakili label kelas atau nilai regresi [16].
- Random Forest adalah metode *ensemble* untuk klasifikasi dan regresi yang menggunakan banyak pohon keputusan. Outputnya adalah modus kelas (untuk klasifikasi) atau rata-rata prediksi (untuk regresi) dari semua pohon yang dibangun, mengatasi masalah *overfitting* dengan menggunakan subset data dan fitur yang berbeda untuk setiap pohon [17].

Teknik ini diharapkan dapat meningkatkan akurasi prediksi dengan memanfaatkan keunggulan masing-masing *base model*.

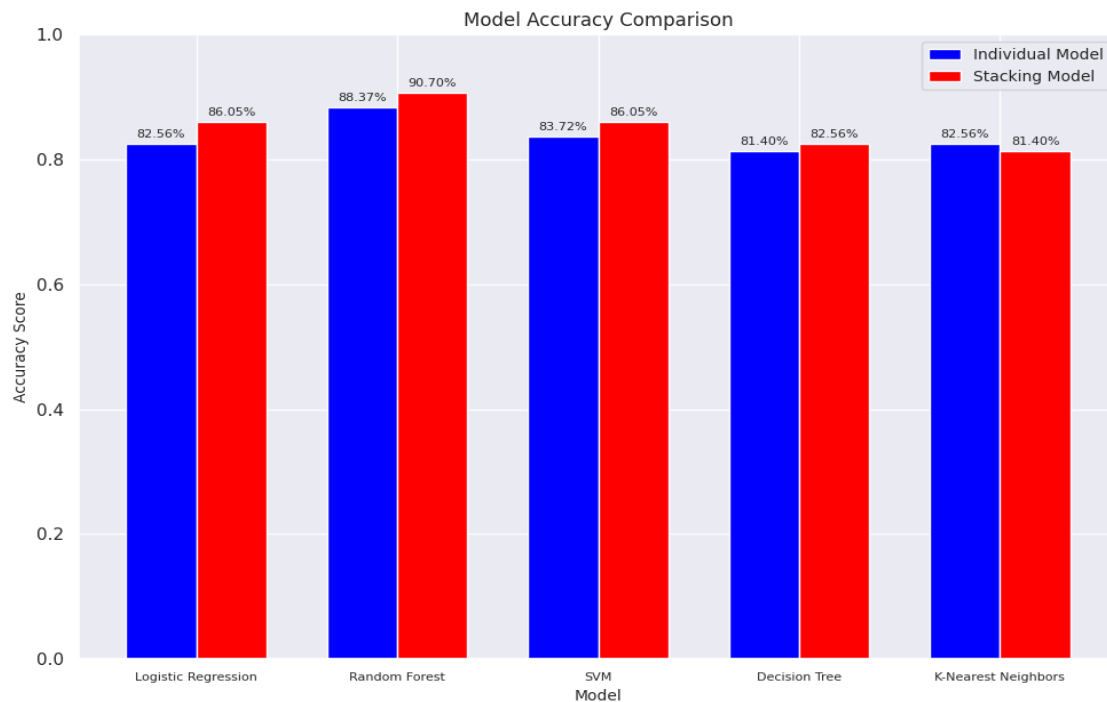
III. HASIL DAN PEMBAHASAN

Penelitian ini menerapkan algoritma Logistic Regression, Support Vector Classifier, K-Nearest Neighbors, Decision Tree, dan Random Forest sebagai *base model* atau *weak learner* sebelum diterapkannya metode *stacking ensemble*. Setelah itu menguji coba semua *stacking model* dengan lima *base model* yang sudah ditetapkan dengan *meta learner* yang juga berasal dari *base model*. Tabel 3. menunjukkan keseluruhan hasil evaluasi model menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score*.

Tabel 3. Hasil Performa Model Individu dan Stacking

Algoritma	Precision	Recall	F1-Score	Accuracy
Logistic Regression	0.9024	0.7708	0.8315	0.8256
K-Nearest Neighbors	0.8293	0.8095	0.8193	0.8256
Support Vector Machine	0.9756	0.7542	0.8511	0.8372
Decision Tree	0.8293	0.7907	0.8095	0.8140
Random Forest	0.8537	0.8974	0.8750	0.8837
Stacking - Meta Learner LR	0.8780	0.8372	0.8571	0.8605
Stacking - Meta Learner KNN	0.8293	0.7907	0.8095	0.8140
Stacking - Meta Learner SVM	0.8293	0.8718	0.8500	0.8605
Stacking - Meta Learner DT	0.9024	0.7708	0.8315	0.8256
Stacking - Meta Learner RF	0.9512	0.8667	0.9070	0.9070

Gambar 4. berikut menunjukkan perbandingan akurasi model individu dan *stacking*. Secara umum model *stacking* mampu mencapai akurasi lebih optimal. Pada penelitian ini juga menunjukkan bahwa KNN justru tidak mampu optimal dengan model *stacking*. Hal ini perlu diselidiki lebih jauh penyebabnya. Namun secara konsep algoritma, KNN adalah algoritma *lazy*, bisa jadi hal ini justru menjadi kelemahan pada model *stacking*, yaitu tidak mampu bekerja optimal pada algoritma *lazy*.



Gambar 4. Perbandingan Akurasi Model Individu dan *Stacking*

Berdasarkan Gambar 4. histogram yang berwarna biru mewakili skor akurasi model individu (*base model*) dan histogram yang berwarna merah mewakili skor akurasi model *stacking* dengan *meta learner* dari masing-masing *base model*. Hasil akurasi untuk Decision Tree sebesar 81.40%, KNN sebesar 82.56%, SVM sebesar 83.72%, Logistic Regression sebesar 82.56%, Random Forest sebesar 88.37%, *stacking* dengan *meta learner* Logistic Regression sebesar 86.05%, *stacking* dengan *meta learner* KNN sebesar 81.40%, *stacking* dengan *meta learner* Random Forest sebesar 90.70%, *stacking* dengan *meta learner* SVM sebesar 86.05%, dan *stacking* dengan *meta learner* Decision Tree sebesar 82.56%. Histogram diurutkan dari *range* perubahan akurasi yang paling besar ke yang paling kecil. Secara keseluruhan, penerapan *stacking* berhasil meningkatkan akurasi model individu, kecuali pada hasil KNN yang mengalami penurunan 1.16% setelah diterapkan *stacking ensemble*.

IV. KESIMPULAN

Penelitian ini mengembangkan sistem prediksi status pinjaman bank menggunakan berbagai algoritma pembelajaran mesin, termasuk Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Decision Tree, dan Random Forest, dengan pendekatan *stacking ensemble learning*. Hasil menunjukkan bahwa kombinasi model melalui *stacking ensemble* memberikan prediksi yang lebih baik dibandingkan model individu walaupun hasil prediksinya tidak terlalu signifikan.

Penerapan teknik *stacking* mampu meningkatkan akurasi model cukup signifikan hanya pada beberapa model. Terutama pada *meta learner* dari Random Forest yang berhasil mencapai akurasi



90.70% yang naik dari 88.37% pada model Random Forest asli. Selain itu, *stacking* dengan *meta learner* SVM dan Logistic Regression juga menunjukkan peningkatan yang signifikan menjadi 86.05% dari 82.56%. Namun, tidak semua penggunaan *stacking* berhasil meningkatkan akurasi, seperti pada *meta learner* Decision Tree yang tetap memiliki akurasi 82.56%, sama dengan model Decision Tree asli. Kesimpulan ini menegaskan bahwa keberhasilan teknik *stacking* dalam meningkatkan akurasi bergantung pada kombinasi model individu yang dipilih serta jenis meta learner yang digunakan. Secara keseluruhan, hasil akurasi dari penerapan *stacking ensemble* dengan penggunaan *meta learner* dari *base model* tidak mempengaruhi model secara signifikan. Sehingga diperlukan penelitian lebih lanjut terkait kemungkinan adanya *overfitting* pada model, metode penelitian, dan pemilihan model pada *stacking ensemble*.

REFERENSI

1. S. Syafudin, R. Agastya Nugraha, K. Handayani, W. Gata, and S. Linawati, “Prediksi Status Pinjaman Bank dengan Deep Learning Neural Network (DNN),” *Jurnal Teknik Komputer AMIK BSI*, vol. 7, no. 2, 2021, doi: 10.31294/jtk.v4i2.
2. A. Kurniawan, A. Rifa’i, M. A. Nafis, N. Sefrida, and H. Patria, “Pemilihan Metode Predictive Analytics dengan Machine Learning untuk Analisis dan Strategi Peningkatan Kualitas Kredit Perbankan,” *Indonesian Journal of Applied Statistics*, vol. 5, no. 1, p. 1, May 2022, doi: 10.13057/ijas.v5i1.55483.
3. M. Madaan, A. Kumar, C. Keshri, R. Jain, and P. Nagrath, “Loan default prediction using decision trees and random forest: A comparative study,” in *IOP Conference Series: Materials Science and Engineering*, IOP Publishing Ltd, Jan. 2021. doi: 10.1088/1757-899X/1022/1/012042.
4. C. L. Perera and S. C. Premaratne, “An Ensemble Machine Learning Approach for Forecasting Credit risk of Loan Applications,” *WSEAS Transactions on Systems*, vol. 23, pp. 31–46, 2024, doi: 10.37394/23202.2024.23.4.
5. M. A. Muslim *et al.*, “New model combination meta-learner to improve accuracy prediction P2P lending with stacking ensemble learning,” *Intelligent Systems with Applications*, vol. 18, May 2023, doi: 10.1016/j.iswa.2023.200204.
6. F. Zhu, X. Chen, and G. Li, “Multi-classification assessment of personal credit risk based on stacking integration,” in *Procedia Computer Science*, Elsevier B.V., 2022, pp. 605–612. doi: 10.1016/j.procs.2022.11.218.
7. N. Uddin, M. K. Uddin Ahamed, M. A. Uddin, M. M. Islam, M. A. Talukder, and S. Aryal, “An ensemble machine learning based bank loan approval predictions system with a smart application,” *International Journal of Cognitive Computing in Engineering*, vol. 4, pp. 327–339, Jun. 2023, doi: 10.1016/j.ijcce.2023.09.001.
8. J. Cahya Mestika, M. Oktavio Selan, and M. Iqbal Qadafi, “Menjelajahi Teknik-Teknik Supervised Learning untuk Pemodelan Prediktif Menggunakan Python.” [Online]. Available: <https://jurnalmahasiswa.com/index.php/biikma>
9. S. Khomsah, “Prediksi Harapan Hidup Penderita Hepatitis Kronik Menggunakan Metode-Metode Klasifikasi,” in *Seminar Nasional Informatika Medis (Snimed)*, 2018, pp. 38–45.
10. IEEE Staff, “Improving Imbalanced Dataset Classification Using Oversampling and Gradient Boosting,” in *2019 5th International Conference on Science in Information Technology (ICSITech)*, IEEE, 2019, pp. 217–223.
11. M. Persada Pulungan, A. Purnomo, A. Kurniasih, S. Tinggi Ilmu Manajemen dan Ilmu Komputer ESQ, and P. Korespondensi, “PENERAPAN SMOTE UNTUK MENGATASI IMBALANCE CLASS DALAM KLASIFIKASI KEPERIBADIAN MBTI MENGGUNAKAN NAIVE BAYES CLASSIFIER APPLICATION OF SMOTE,” *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 10, no. 7, pp. 1493–1502, 2023, doi: 10.25126/jtiik.2023107989.
12. J. Chen *et al.*, “Detecting Hypoglycemia Incidents Reported in Patients’ Secure Messages: Using Cost-Sensitive Learning and Oversampling to Reduce Data Imbalance,” *J Med Internet Res*, vol. 21, no. 3, Mar. 2019, doi: 10.2196/11990.
13. E. Agbemava, I. K. Nyarko, T. C. Adade, and A. K. Bediako, “Logistic Regression Analysis Of Predictors Of Loan Defaults By Customers Of Non-Traditional Banks In Ghana,” *European Scientific Journal, ESJ*, vol. 12, no. 1, p. 175, Jan. 2016, doi: 10.19044/esj.2016.v12n1p175.



Seminar Nasional Sains Data 2024 (SENADA 2024)

E-ISSN 2808-5841

UPN “Veteran” Jawa Timur

P-ISSN 2808-7283

14. T. M. Cover and P. E. Hart, “Nearest Neighbor Pattern Classification,” *IEEE Trans Inf Theory*, vol. 24, no. 1, pp. 335–342, 1967.
15. C. Cortes, V. Vapnik, and L. Saitta, “Support-Vector Networks Editor,” Kluwer Academic Publishers, 1995.
16. J.R. Quinlan, “Induction of Decision Trees,” *Mach Learn*, vol. 1, pp. 81–106, 1986.
17. L. Breiman, “Random Forests,” *Mach Learn*, vol. 45, pp. 5–32, 2001.