

PENGEMBANGAN SISTEM DETEKSI DINI MAHASISWA BERISIKO MENGGUNAKAN *MACHINE LEARNING* BERBASIS *DATA LEARNING MANAGEMENT SYSTEM*

(Studi Kasus: rumahilmu.org)

Wahyu Syahputra¹, Endina Putri Purwandari², Widhia KZ Oktoeberza³

^{1,2,3} Program Studi Informatika, Fakultas Teknik, Universitas Bengkulu,
Jl. WR. Supratman, Kandang Limun, Bengkulu, 3871A
Telp. (627) 3621170, Faks (627) 3622105

¹syahputrawahyu61@gmail.com

²endinaputri@unib.ac.id

³widhiakz@unib.ac.id

Abstrak: Penelitian ini bertujuan mengembangkan sistem deteksi dini mahasiswa berisiko menggunakan *machine learning* berbasis data dari *Learning Management System* (LMS) rumahilmu.org. Sistem ini dirancang untuk Program Studi Sistem Informasi Universitas Bengkulu, dengan menganalisis data dari 459 *enrollment* mahasiswa pada lima mata kuliah. Sebanyak 37–76 fitur diekstraksi dari aktivitas LMS untuk memprediksi mahasiswa yang berpotensi mendapat nilai di bawah persentil ke-30 pada tiga titik waktu strategis (25%, 50%, dan 75% semester). Penelitian ini menerapkan pendekatan optimasi per-kelas, menguji 11 algoritma untuk menemukan model terbaik bagi setiap mata kuliah. Hasil penelitian menunjukkan bahwa tidak ada satu algoritma tunggal yang *superior*; model paling efektif bervariasi untuk setiap mata kuliah, dengan *Gaussian Process*, *Logistic Regression*, dan *Voting Classifier* menjadi yang paling sering terpilih. Namun, evaluasi pada data uji menunjukkan tantangan signifikan: meskipun skor validasi silang tinggi (*F1-score* > 0.80), terjadi *overfitting* dan penurunan performa. Temuan paling krusial adalah rendahnya kemampuan model dalam mendeteksi kelas minoritas 'Berisiko', dengan metrik *Recall* (Berisiko) mencapai 0.00 pada 8 dari 15 skenario. Kinerja deteksi terbaik dicapai pada mata kuliah Statistika & Probabilitas dengan *Recall* 0.50. Sistem yang diimplementasikan dengan arsitektur *3-tier* (FastAPI dan React) menyediakan *dashboard* interaktif, namun efektivitas prediksinya untuk deteksi dini dibatasi oleh *dataset* yang kecil dan tidak seimbang.

Kata Kunci: deteksi dini mahasiswa berisiko, learning analytics, machine learning, klasifikasi biner, learning management system

Abstract: *This research aims to develop an early detection system for at-risk students using machine learning based on data from the Learning Management System (LMS) rumahilmu.org. The system was designed for the Information Systems Study Programs at the University of Bengkulu, analyzing data from 459 student enrollments across five courses. A total of 37–76 features were extracted from LMS activities to predict students likely to score below the 30th percentile at three strategic time points (25%, 50%, and 75% of the semester). This study implemented a per-class optimization approach, testing 11 algorithms to find the best model for each course. The results showed that no single algorithm was universally superior; the most effective models varied for each course, with Gaussian Process, Logistic Regression, and Voting Classifier being the most frequently chosen. However, evaluation on the test data revealed significant challenges: despite high cross-validation scores ($F1\text{-score} > 0.80$), overfitting and performance degradation occurred. The most critical finding was the model's low capability in detecting the 'At-Risk' minority class, with the Recall (At-Risk) metric reaching 0.00 in 8 out of 15 scenarios. The best detection performance was achieved in the Statistics & Probability course with a Recall of 0.50. The implemented system, featuring a 3-tier architecture (FastAPI and React), provides an interactive dashboard, but its predictive effectiveness for early detection is limited by small and imbalanced datasets.*

Keywords: *early detection of at-risk students, learning analytics, machine learning, binary classification, learning management system.*

I. PENDAHULUAN

Model pembelajaran di perguruan tinggi secara historis didominasi oleh pendekatan tradisional yang berpusat pada pengajar (*teacher-centered*) [1]. Di mana dosen berperan sebagai sumber utama pengetahuan dan mahasiswa diposisikan sebagai penerima pasif. Meskipun alat bantu mengajar telah berevolusi dari papan tulis menjadi proyektor digital, esensi peran mahasiswa sebagai audiens seringkali tidak berubah secara fundamental. Namun, paradigma pembelajaran konvensional ini menghadapi tantangan signifikan untuk beradaptasi dengan tuntutan era digital,

sehingga efektivitas dan kualitasnya semakin dipertanyakan.

Generasi mahasiswa saat ini merupakan generasi digital yang tumbuh di tengah kemudahan akses terhadap internet dan teknologi. Kondisi ini membawa dua perubahan fundamental dalam dinamika kelas. Pertama, internet menyediakan sumber pengetahuan alternatif yang tak terbatas [2], menyebabkan peran dosen sebagai satu-satunya otoritas keilmuan mulai tergeser. Ketidakpuasan mahasiswa terhadap konten, metode penyampaian, atau gaya mengajar dapat dengan mudah memicu hilangnya atensi, seperti mengantuk, tidak menyimak, hingga membolos. Kedua, perangkat seluler telah menjadi bagian tak terpisahkan dari kehidupan mahasiswa [2], yang penggunaannya secara berlebihan di dalam kelas dapat mengalihkan fokus dan mengganggu proses penyerapan materi [3].

Selain itu, tantangan ini diperburuk oleh praktik kelas berkapasitas besar yang masih umum diterapkan di banyak institusi. Model pengajaran massal tersebut menyulitkan implementasi pendekatan pembelajaran yang bersifat personal dan individual, sehingga berpotensi memberikan dampak negatif terhadap keterlibatan dan kinerja akademik mahasiswa [4]. Oleh karena itu, terdapat kebutuhan mendesak untuk mentransformasi metode pengajaran agar lebih selaras dengan interaksi antara teknologi, karakteristik mahasiswa modern, dan tujuan pembelajaran yang efektif.

Performa akademik mahasiswa dalam setiap mata kuliah merupakan tolok ukur fundamental bagi keberhasilan proses pembelajaran dan menjadi perhatian utama bagi pengelola program studi [5]. Pencapaian nilai yang optimal pada setiap mata kuliah tidak hanya merefleksikan pemahaman mahasiswa terhadap materi perkuliahan, tetapi juga mempengaruhi kelancaran studi dan kesuksesan akademik mereka. Di Program Studi Sistem

Informasi dan Informatika Universitas Bengkulu, mahasiswa [12] [7] [13] [14]. Penelitian terkini upaya untuk memastikan mahasiswa dapat mencapai nilai terbaik di setiap mata kuliah menjadi sebuah prioritas. Ketua Program Studi, sebagai salah satu pihak yang bertanggung jawab atas kualitas dan kelancaran proses akademik, memerlukan alat bantu yang efektif untuk memantau dan mengidentifikasi secara dini mahasiswa yang berpotensi mendapat nilai rendah pada mata kuliah yang sedang berjalan.

Identifikasi dini mahasiswa berisiko menjadi krusial karena memungkinkan dilakukannya intervensi yang lebih cepat dan tepat sasaran [6] [7]. Keterlambatan dalam mendeteksi potensi kegagalan dapat berakibat pada penurunan motivasi mahasiswa, kesulitan mengejar ketertinggalan materi, hingga risiko mendapat nilai di bawah standar kelulusan mata kuliah. Pendekatan tradisional yang mengandalkan pemantauan nilai akhir semester seringkali bersifat reaktif, di mana tindakan baru diambil setelah mahasiswa sudah gagal. Oleh karena itu, diperlukan sebuah sistem yang lebih proaktif untuk deteksi dini, sering disebut sebagai sistem peringatan dini (Early Warning System - EWS) [8] [9].

Seiring dengan kemajuan teknologi informasi, institusi pendidikan semakin banyak menghasilkan data digital terkait aktivitas pembelajaran mahasiswa [5].

Learning Management System (LMS) telah menjadi sumber data yang sangat berharga karena merekam jejak aktivitas pembelajaran mahasiswa secara daring dengan detail [10] [11]. Di Universitas Bengkulu, LMS *rumahilmu.org* telah digunakan oleh mahasiswa Program Studi Sistem Informasi dan Informatika, menghasilkan data log aktivitas belajar yang sangat kaya dan komprehensif. Data dari LMS ini mencakup berbagai aspek pembelajaran seperti frekuensi login, jumlah akses materi, partisipasi dalam forum diskusi, skor kuis, ketepatan waktu pengumpulan tugas, dan pola aktivitas temporal

menunjukkan bahwa data aktivitas LMS saja sudah memiliki kemampuan prediktif yang kuat terhadap performa akademik mahasiswa tanpa memerlukan data eksternal [14] [15]. Hal ini membuka peluang untuk mengembangkan sistem prediksi yang lebih sederhana namun efektif dengan hanya memanfaatkan data yang sudah tersedia di LMS.

Penelitian sebelumnya telah menunjukkan bahwa teknik *Educational Data Mining* (EDM) dan *machine learning* (ML) dapat secara efektif digunakan untuk menganalisis data pendidikan dan memprediksi berbagai aspek kinerja mahasiswa [5] [8] [9] [16]. Berbagai algoritma ML telah dibandingkan untuk tujuan prediksi kinerja, dan banyak penelitian fokus pada pengembangan EWS untuk mengidentifikasi mahasiswa berisiko [8] [11] [17]. Model prediktif yang dihasilkan tidak hanya bertujuan untuk akurasi, tetapi juga diharapkan dapat memberikan insight yang dapat ditindaklanjuti (actionable) oleh pihak terkait [18]. Pemodelan mahasiswa berdasarkan data pembelajaran juga menjadi area kajian yang penting dalam learning analytics [18] [19]. Penggunaan data log dari LMS untuk memprediksi kinerja juga telah banyak dieksplorasi [20].

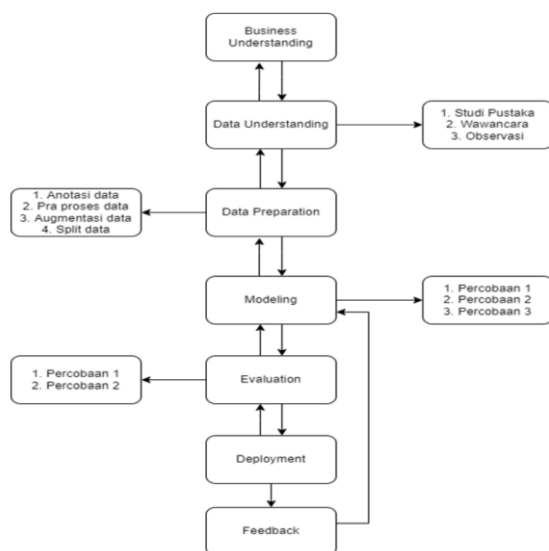
Meskipun demikian, penelitian yang secara spesifik menggunakan data aktivitas LMS untuk memprediksi nilai mata kuliah mahasiswa dengan pendekatan klasifikasi biner di konteks Program Studi Sistem Informasi dan Informatika Universitas Bengkulu masih perlu dikembangkan. Sistem yang memanfaatkan data LMS saja tanpa memerlukan integrasi kompleks dengan sistem akademik lain dapat memberikan solusi yang lebih praktis dan mudah diimplementasikan. Dengan menggunakan threshold dinamis berdasarkan persentil ke-30 dari distribusi nilai kelas, sistem dapat mengidentifikasi mahasiswa yang berisiko mendapat nilai rendah pada

mata kuliah yang sedang berjalan. Pendekatan ini memungkinkan Ketua Program Studi untuk melakukan evaluasi, pengambilan keputusan, dan perencanaan intervensi yang lebih terarah dan tepat waktu bagi mahasiswa yang teridentifikasi berisiko, sebagaimana pentingnya pemahaman terhadap faktor-faktor prediktor dari aktivitas LMS yang telah diidentifikasi oleh berbagai penelitian sebelumnya [10] [7].

II. METODOLOGI PENELITIAN

A. Rancangan Penelitian

Penelitian ini dilaksanakan menggunakan metode CRISP-DM dari IBM. CRISP-DM merupakan singkatan dari *Cross-Industry Standard Process for Data Mining* yang digunakan oleh industri global dalam mengembangkan suatu proyek untuk *data science*. Alur metode ini dapat dilihat pada diagram alir yang digambarkan pada gambar 1



Gambar 1. Alur Metode CRISP-DM

Metode ini diawali dengan *business understanding* yang merupakan proses pemahaman terhadap masalah yang akan diselesaikan. Selanjutnya adalah kegiatan pemahaman terhadap data (*data understanding*) yang meliputi penentuan kebutuhan data, pengumpulan data dan eksplorasi data.

Tahapan berikutnya adalah *data preparation* yang dilakukan untuk memperbaiki kualitas data agar sesuai dengan proses data agar sesuai dengan proses *modeling*. Pada tahapan *modeling*, data yang sudah siap akan dilakukan dilakukan proses pelatihan berdasarkan model yang akan akan dibuat berdasarkan kasus yang diinginkan. Tahapan berikutnya adalah evaluasi sebelum model yang dihasilkan akan dilakukan proses *deployment* agar menjadi sistem operasional. Tahapan terakhir diakhiri dengan proses *feedback* dan pelaporan. Urutan fase tidak ketat. Faktanya, sebagian besar proyek bergerak bolak-balik antar fase sesuai kebutuhan [21].

B. Jenis Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan menerapkan metode *Educational Data Mining* (EDM). Jenis penelitian yang dilakukan adalah penelitian terapan dengan tujuan untuk membangun model klasifikasi biner guna memprediksi mahasiswa yang berisiko mendapat nilai di bawah persentil ke-30 pada setiap mata kuliah. Penelitian ini juga memiliki aspek deskriptif dalam menganalisis faktor-faktor dari aktivitas LMS yang mempengaruhi prediksi nilai. Pendekatan machine learning dengan teknik supervised learning dan per-class optimization akan menjadi inti dari proses pemodelan. [18] membahas metodologi pemodelan mahasiswa berbasis ML yang relevan dengan pendekatan yang digunakan dalam penelitian ini. Selain itu, penelitian ini juga akan melibatkan pengembangan aplikasi web dengan arsitektur 3-tier sebagai implementasi dari model yang dihasilkan.

C. Metode Pengumpulan Data

Pengumpulan data dalam penelitian ini akan dilakukan dari satu sumber utama:

Learning Management System (LMS) rumahilmu.org. Data aktivitas pembelajaran

mahasiswa akan diekstraksi dari database LMS untuk lima mata kuliah yang menjadi objek penelitian. Data yang dikumpulkan mencakup:

1. Log aktivitas mahasiswa dari tabel `activity_log` (frekuensi login, durasi akses)
2. Skor dan pola pengerjaan kuis dari tabel `nilai_kuis_materis`
3. Partisipasi diskusi dari tabel `diskusi` dan `diskusi_respons`
4. Ketepatan waktu pengumpulan tugas dari tabel `pengumpulan_tugas_individus`
5. Data penilaian kelompok dari tabel `penilaian_kelompok`
6. Nilai akhir mata kuliah (`rata_rata_nilai`) sebagai target prediksi

Ekstraksi data dilakukan pada tiga titik waktu semester (25%, 50%, dan 75%) untuk memungkinkan prediksi dini, menghasilkan 37-76 fitur tergantung pada titik waktu prediksi.

D. Desain Penelitian

Desain penelitian ini mengikuti tahapan standar dalam proyek data mining dan pengembangan sistem prediktif, yang dapat diadaptasi dari kerangka kerja seperti CRISP-DM (*Cross-Industry Standard Process for Data Mining*) atau metodologi pengembangan model *machine learning* pada umumnya. Tahapan utama penelitian adalah sebagai berikut:

1. Pemahaman Bisnis dan Data (Business and Data Understanding)
 - a. Mendefinisikan secara jelas tujuan penelitian, yaitu memprediksi mahasiswa yang berisiko mendapat nilai di bawah persentil ke-30 pada mata kuliah untuk mendukung Ketua Program Studi Sistem Informasi dan Informatika.
 - b. Mengidentifikasi dan memahami

struktur data LMS rumahilmu.org, termasuk format, kualitas, dan potensi variabel yang relevan dari 37-76 fitur yang akan diekstrak.

2. Pengumpulan dan Integrasi Data (Data Collection and Integratio)}

- a. Mengekstraksi data yang diperlukan dari LMS rumahilmu.org untuk 459 enrollment mahasiswa dari lima mata kuliah yang menjadi objek penelitian.
- b. Mengorganisir data ke dalam dataset per mata kuliah dengan fitur-fitur yang diekstrak pada titik waktu prediksi (25%, 50%, 75% semester).

3. Pra-pemrosesan Data (Data Preprocessing)

- a. Pembersihan Data: Menangani nilai yang hilang (missing values) dengan teknik imputasi yang sesuai (misalnya, mean, median, atau modus imputation), mendeteksi dan menangani pencilan (outliers) jika diperlukan.
- b. Transformasi Data: Mengubah format data jika diperlukan (misalnya, konversi timestamp menjadi durasi aktivitas, agregasi log per periode waktu).
- c. Encoding Variabel Kategorikal: Mengubah variabel kategorikal (misalnya, jenis kelamin, program studi, status SMA) menjadi representasi numerik yang dapat diproses oleh algoritma machine learning (misalnya, one-hot encoding atau label encoding).

4. Rekayasa Fitur (*Feature Engineering*)

- a. Membuat fitur-fitur baru yang mengacu pada metodologi yang divalidasi oleh [22]. lebih informatif dari data mentah, Dalam penelitiannya, mereka menggunakan persentil seperti: Jumlah log aktivitas, ke-30 sebagai salah satu tolok ukur yang valid untuk diskusi dibuat/respons, rata-rata mengukur perbaikan kinerja siswa berisiko. Hasil nilai kuis, keterlambatan tugas, studi tersebut menunjukkan bahwa ambang batas ini nilai tugas individu, peer signifikan secara statistik untuk mengidentifikasi assessment, rasio jawaban benar siswa yang berhasil keluar dari kategori risiko tinggi. level kognitif tinggi, rasio respons Oleh karena itu, pendekatan ini diadopsi sebagai integrasi, dan pola engagement metode yang berbasis bukti (evidence-based) untuk sepanjang waktu. mendefinisikan risiko secara dinamis.
- b. Seleksi Fitur (Feature Selection): Mengidentifikasi dan memilih subset fitur yang paling relevan dan prediktif untuk mengurangi dimensi data dan meningkatkan performa model (misalnya, menggunakan teknik filter seperti korelasi, atau teknik wrapper/embedded seperti RFE atau feature importance dari model tree-based).
5. Definisi Variabel Target (Label):
 - a. Mengumpulkan nilai akhir mata kuliah (rata_rata_nilai) dari LMS untuk setiap enrollment mahasiswa.
 - b. Menghitung threshold dinamis menggunakan persentil ke-30 dari distribusi nilai untuk setiap mata kuliah.
 - c. Mengkategorikan mahasiswa ke dalam dua kelas: At-Risk (nilai di bawah persentil ke-30) dan Safe (nilai pada atau di atas persentil ke-30). Pendekatan threshold dinamis ini menghindari bias dari perbedaan standar penilaian antar mata kuliah.
- Pemilihan ambang batas persentil ke-30 ini
6. Pemodelan (*Modeling*):
 - a. Pembagian Data: Membagi dataset per mata kuliah menjadi data latih (training set) dan data uji (testing set) dengan proporsi 80:20 menggunakan teknik stratified sampling untuk menjaga proporsi kelas at-risk dan safe.
 - b. Pemilihan Algoritma Klasifikasi: Menguji 11 algoritma klasifikasi: Logistic Regression, Naive Bayes, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Decision Tree, Random Forest, Gradient Boosting, XGBoost, AdaBoost, LightGBM, dan Neural Network (MLP).
 - c. Pelatihan Model: Melatih setiap algoritma klasifikasi per mata kuliah menggunakan pendekatan per-class optimization.
 - d. Optimasi Model (Hyperparameter Tuning): Melakukan penyesuaian hiperparameter untuk setiap model guna mendapatkan performa terbaik (misalnya, menggunakan Grid Search atau Random Search dengan validasi silang pada data latih).
7. Evaluasi Model (*Evaluation*):

-
- | | |
|--|--|
| <p>a. Mengevaluasi performa model-model yang telah dilatih pada data uji menggunakan metrik evaluasi klasifikasi yang sesuai, seperti akurasi, presisi, recall, F1-score (per kelas dan makro/rata-rata), serta kurva ROC dan AUC (Area Under Curve).</p> <p>b. Menganalisis confusion matrix untuk memahami jenis kesalahan yang dibuat oleh model.</p> <p>c. Membandingkan performa antar algoritma untuk setiap mata kuliah dan memilih model terbaik yang berbeda untuk setiap kelas.</p> <p>d. Menganalisis feature importance dari model terbaik untuk mengidentifikasi faktor-faktor yang paling berpengaruh terhadap prediksi kategori risiko kinerja.</p> | <p>III. HASIL DAN PEMBAHASAN</p> <p>A. Hasil Analisis Data</p> <p>Pada tahap awal penelitian, dilakukan pengumpulan data terkait aktivitas pembelajaran mahasiswa dari database MySQL website_adaptif yang merupakan bagian dari Learning Management System (LMS) rumahilmu.org. Data mentah yang diperoleh kemudian dilakukan pra-pengolahan untuk membersihkan data dan mengekstraksi fitur-fitur prediktif. Jenis data yang diekstraksi mencakup jumlah log aktivitas, partisipasi diskusi, performa kuis, pengumpulan tugas, dan berbagai metrik temporal yang merekam perkembangan mahasiswa dari waktu ke waktu. Hasil pra-pengolahan data menunjukkan bahwa terdapat data yang hilang (missing values) pada beberapa fitur, terutama yang berkaitan dengan aktivitas yang tidak wajib atau belum terjadi pada titik waktu ekstraksi, serta teridentifikasinya outlier yang merepresentasikan perilaku mahasiswa yang ekstrem.</p> |
|--|--|
8. Pengembangan Prototipe Aplikasi Web: Analisis lebih lanjut terhadap data yang telah
- | | |
|--|---|
| <p>a. Merancang antarmuka pengguna (<i>dashboard</i>) untuk Ketua Program Studi Sistem Informasi.</p> <p>b. Mengimplementasikan backend untuk memuat model prediktif terbaik dan memproses data input mahasiswa baru (jika ada skenario penggunaan real-time sederhana).</p> <p>c. Mengimplementasikan frontend untuk menampilkan hasil prediksi, faktor pengaruh, dan visualisasi data.</p> | <p>bersih menunjukkan adanya beberapa pola atau tren yang signifikan. Sebagai contoh representatif, analisis pada dataset mata kuliah Pengantar Basis Data (ID 33) pada titik waktu 50% semester menunjukkan bahwa performa awal mahasiswa pada kuis memiliki korelasi positif yang kuat dengan nilai akhir, sementara keterlambatan dalam pengumpulan tugas menunjukkan korelasi negatif. Analisis statistik deskriptif seperti mean, median, dan standar deviasi digunakan untuk menggambarkan karakteristik data secara kuantitatif. Hasil analisis statistik menunjukkan bahwa, sebagai contoh, rata-rata nilai akhir mahasiswa pada kelas ini adalah 4.09 dengan standar deviasi 0.52, mengindikasikan variasi performa yang cukup signifikan di antara mahasiswa.</p> |
|--|---|
- Pendekatan ini sejalan dengan metodologi yang umum digunakan dalam penelitian EDM dan pengembangan sistem prediktif berbasis ML [18] [16].

1. Analisis Nilai Hilang (*Missing Values*)

Untuk memberikan gambaran mendalam mengenai karakteristik data,

dilakukan Analisis Data Eksploratif (EDA) pada salah satu skenario, yaitu mata kuliah Pengantar Basis Data pada pertengahan semester.

Identifikasi nilai yang hilang merupakan langkah krusial karena dapat memengaruhi kualitas dan performa model prediksi. Analisis pada dataset studi kasus (Kelas 33 pada 50% semester) menunjukkan bahwa data yang digunakan memiliki tingkat kelengkapan yang sangat tinggi.

Seperti yang dirangkum pada tabel (ringkasan nilai hilang), hanya dua fitur yang memiliki data hilang, yaitu `correct_high_level` dan `total_high_level`. Kedua fitur ini hanya memiliki 3 nilai yang hilang, yang setara dengan 2.44% dari total sampel pada kelas tersebut.

Tabel 1. Ringkasan Nilai Hilang

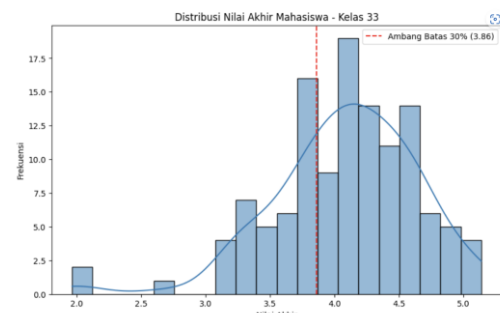
Nama Fitur	Jumlah Hilang	Persentase Hilang (%)
<code>correct_high_level</code>	3	2.44
<code>total_high_level</code>	3	2.44

Nilai yang hilang pada kedua fitur ini kemungkinan besar terjadi karena sebagian kecil mahasiswa tidak berpartisipasi dalam kuis yang mengandung soal-soal dengan tingkat kognitif tinggi. Mengingat persentase nilai hilang yang sangat rendah, dapat disimpulkan bahwa dataset untuk studi kasus ini sangat bersih dan berkualitas tinggi.

2. Distribusi Variabel Target

Variabel target dalam penelitian ini adalah nilai akhir (`final_score`), yang menjadi dasar untuk menentukan apakah seorang mahasiswa dikategorikan berisiko atau tidak. Gambar distribusi nilai akhir

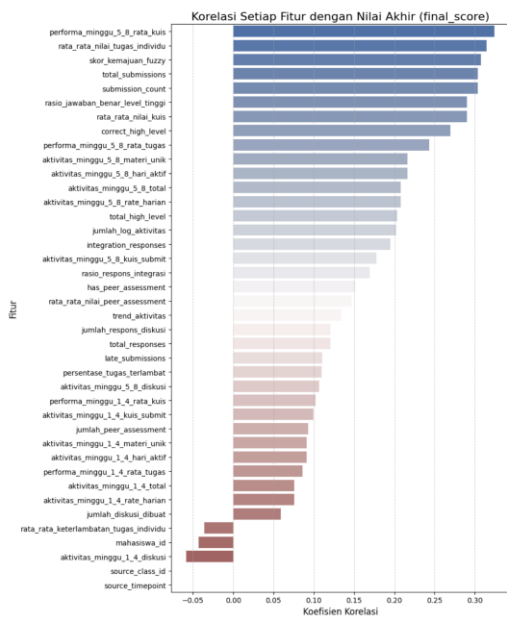
mahasiswa untuk kelas ini menunjukkan kecenderungan mendekati kurva normal, dengan mayoritas mahasiswa terkonsentrasi pada rentang nilai 3.5 hingga 4.5. Berdasarkan distribusi ini, dan sesuai dengan metodologi yang dijelaskan pada bab sebelumnya, ditetapkan ambang batas dinamis pada persentil ke-30. Untuk studi kasus ini, ambang batas tersebut jatuh pada nilai 3.86. Mahasiswa dengan nilai akhir di bawah 3.86 diklasifikasikan sebagai Berisiko (At-Risk), sementara mereka yang memiliki nilai di atas atau sama dengan 3.86 diklasifikasikan sebagai Aman (Safe). Pendekatan ini memastikan bahwa sekitar 30% mahasiswa dengan performa terendah diidentifikasi untuk potensi intervensi.



Gambar 2. Distribusi Nilai Akhir Mahasiswa

Distribusi nilai menunjukkan kecenderungan mendekati kurva normal, dengan mayoritas mahasiswa terkonsentrasi pada rentang nilai 3.5 hingga 4.5. Berdasarkan distribusi ini, dan sesuai dengan metodologi yang dijelaskan pada Bab III, ditetapkan ambang batas (threshold) dinamis pada persentil ke-30. Untuk studi kasus ini, ambang batas tersebut jatuh pada nilai **3.86**, yang ditandai dengan garis putus-putus merah pada gambar. Mahasiswa dengan nilai akhir di bawah 3.86 diklasifikasikan sebagai **Berisiko (At-Risk)**, sementara mereka yang

memiliki nilai di atas atau sama dengan 3.86 diklasifikasikan sebagai **Aman** (Safe). Pendekatan ini memastikan bahwa sekitar 30% mahasiswa dengan performa terendah diidentifikasi untuk potensi intervensi.



Gambar 3. Korelasi Setiap Fitur dengan Nilai Akhir
Dari analisis korelasi pada studi kasus ini, beberapa temuan kunci dapat diidentifikasi:

1. Korelasi Positif Terkuat:

Fitur-fitur yang berkaitan dengan performa akademik secara konsisten menunjukkan korelasi positif terkuat. Fitur `performa_minggu_5_8_rata_kuis` (korelasi sekitar 0.32) dan `rata_rata_nilai_tugas_individu` (korelasi sekitar 0.31) menjadi dua prediktor teratas. Hal ini secara intuitif valid, menandakan bahwa mahasiswa yang berprestasi baik dalam asesmen selama semester cenderung mendapatkan nilai akhir yang tinggi.

2. Korelasi Positif Sedang:

Fitur yang mengukur konsistensi dan keterlibatan kognitif, seperti `skor_kemajuan_fuzzy`, `total_submissions`, dan `rasio_jawaban_benar_level_tinggi`,

juga menunjukkan korelasi positif yang signifikan (antara 0.25 hingga 0.30). Ini mengimplikasikan bahwa bukan hanya nilai, tetapi juga konsistensi dalam menyelesaikan tugas dan kualitas pemahaman materi menjadi faktor pembeda yang penting.

3. Korelasi Negatif:

Fitur `rata_rata_keterlambatan_tugas_individu` menunjukkan korelasi negatif yang paling jelas (meskipun lemah, sekitar -0.04). Ini mengindikasikan bahwa mahasiswa yang sering terlambat dalam mengumpulkan tugas memiliki kecenderungan untuk mendapatkan nilai akhir yang sedikit lebih rendah. Fitur `mahasiswa_id` juga menunjukkan korelasi negatif yang lemah, namun ini kemungkinan besar bersifat acak dan tidak memiliki makna prediktif.

4. Fitur Tidak Relevan:

Fitur `source_class_id` dan `source_timepoint` tidak menunjukkan korelasi yang berarti, yang memang diharapkan karena fitur-fitur ini hanya berfungsi sebagai identifikasi dan tidak mengandung informasi tentang perilaku mahasiswa.

Temuan-temuan dari tahap EDA ini memberikan bukti awal mengenai fitur-fitur mana yang paling relevan dan menjadi landasan penting dalam proses rekayasa fitur dan pemilihan model pada tahap selanjutnya. Fitur-fitur dengan korelasi tertinggi diharapkan akan memiliki bobot atau tingkat kepentingan yang tinggi pula dalam model machine learning yang akan dibangun.

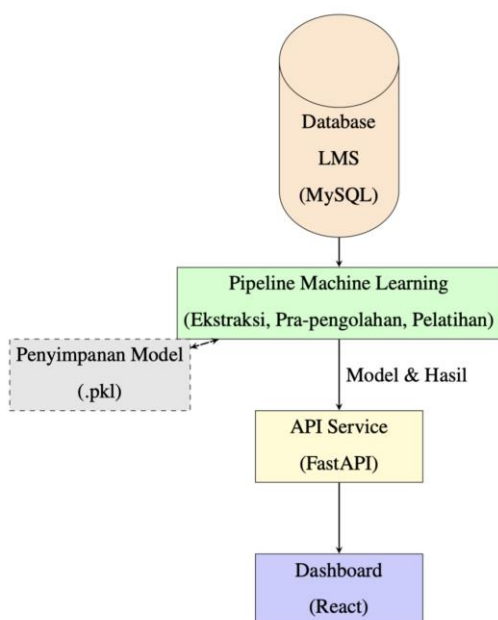
B. Hasil Perancangan Sistem

Bagian ini merinci perancangan sistem prediksi risiko mahasiswa yang dikembangkan, mulai dari analisis kebutuhan fungsional, arsitektur teknis, hingga pemilihan algoritma dan perancangan

antarmuka pengguna.

1. Perancangan Arsitektur Sistem

Arsitektur sistem yang diusulkan terdiri dari beberapa modul utama yang terintegrasi, yaitu modul ekstraksi fitur, modul pra-pengolahan data, modul pelatihan dan klasifikasi, serta modul layanan API. Modul-modul tersebut saling terhubung dan berinteraksi untuk menghasilkan prediksi yang akurat dan dapat diinterpretasikan. Diagram arsitektur sistem disajikan pada Gambar 4.



Gambar 4. Arsitektur Sistem Prediksi Kinerja Mahasiswa

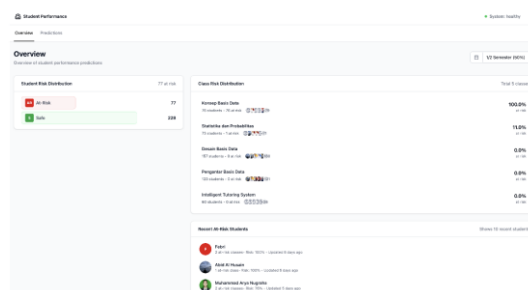
2. Perancangan Algoritma dan Model

Untuk melakukan klasifikasi data dan memprediksi mahasiswa berisiko, penelitian ini mengadopsi pendekatan optimasi per-kelas dengan menguji serangkaian algoritma machine learning. Proses ini diotomatisasi oleh *SmallDatasetTrainer*, sebuah kelas khusus yang dirancang untuk skenario *dataset* pendidikan.

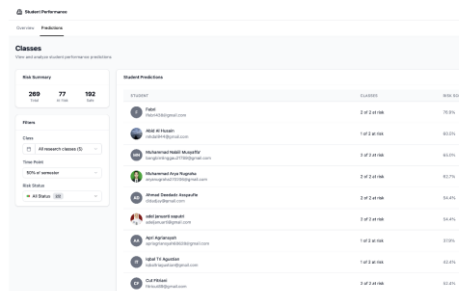
Algoritma *SmallDatasetTrainer* dipilih karena kemampuannya untuk secara sistematis menguji berbagai model, mulai dari model linear sederhana hingga ansambel yang lebih kompleks, dan secara objektif memilih yang terbaik berdasarkan performa cross-validation. Pendekatan ini sangat sesuai untuk kasus di mana ukuran dataset per kelas bervariasi dan seringkali terbatas, sehingga memerlukan pemilihan model yang adaptif. Selain itu, *SmallDatasetTrainer* juga mengintegrasikan langkah-langkah penting seperti seleksi fitur otomatis, yang meningkatkan robustitas dan keandalan model akhir.

3. Perancangan Antarmuka Pengguna

Antarmuka pengguna dirancang dengan tampilan yang sederhana dan intuitif sehingga mudah digunakan oleh pengguna yang tidak memiliki latar belakang teknis, seperti dosen atau kepala program studi. Antarmuka pengguna menyediakan fitur-fitur seperti dasbor ringkasan, daftar mahasiswa dengan indikator risiko, dan halaman detail untuk setiap mahasiswa.



Gambar 5. Tampilan Antarmuka Dasbor Utama



Gambar 6. Tampilan Antarmuka
Predictions

C. Pembahasan

Pada bagian ini, akan dianalisis secara mendalam hasil penelitian yang telah disajikan pada sub-bab sebelumnya. Tujuannya adalah untuk memberikan interpretasi yang mendalam terhadap data, menghubungkan hasil dengan teori yang relevan, dan mengevaluasi kinerja model serta sistem yang telah dibangun.

1. Analisis Kinerja Algoritma Klasifikasi

Evaluasi kinerja model dilakukan dengan menggunakan serangkaian metrik standar klasifikasi, yaitu Akurasi, Presisi, Recall, dan F1-score. Fokus utama evaluasi adalah pada F1-score sebagai ukuran keseimbangan antara presisi dan recall, serta pada Recall untuk kelas 'Berisiko' yang mengukur sensitivitas model dalam mendeteksi mahasiswa yang paling membutuhkan perhatian.

Setelah model terbaik dipilih untuk setiap skenario, performa akhirnya dievaluasi menggunakan data uji yang belum pernah dilihat sebelumnya. Tabel yang merangkum hasil evaluasi dari 15 model yang telah dioptimalkan menunjukkan metrik utama, yaitu F1-Score pada data uji (mengukur keseimbangan antara presisi dan recall) serta Recall untuk kelas 'Berisiko', yang mengukur kemampuan model dalam

mengidentifikasi mahasiswa yang benar-benar berisiko.

Tabel 2. Ringkasan Performa Terbaik

Mata Kuliah	Titik Waktu	Model Terbaik	F1-Score CV	F1-Score Uji	Recall (Berisiko)
Pengantar Basis Data	25%	gaussian_process	0.832	0.687	0.143
	50%	adaptive_knn	0.819	0.603	0.000
	75%	gaussian_process	0.809	0.562	0.000
Desain Basis Data	25%	logistic_regression	0.830	0.601	0.000
	50%	voting_classifier	0.824	0.601	0.000
	75%	voting_classifier	0.826	0.586	0.000
Konsep Basis Data	25%	gaussian_process	0.810	0.686	0.200
	50%	gaussian_nb	0.817	0.729	0.400
	75%	gaussian_nb	0.806	0.639	0.200
Intelligent Tutoring	25%	gaussian_process	0.705	0.533	0.000
	50%	bagged_logistic	0.828	0.533	0.000
	75%	logistic_regression	0.828	0.533	0.000
Statistika & Probabilitas	25%	logistic_regression	0.848	0.850	0.500
	50%	gaussian_process	0.872	0.850	0.500
	75%	bagged_logistic	0.852	0.850	0.500

Hasil evaluasi pada tabel menunjukkan adanya diskrepansi yang signifikan antara performa model pada tahap cross-validation (F1-Score CV) dan performa pada data uji (F1-Score Uji). Meskipun F1-score CV secara konsisten berada di atas 0.80, F1-score pada data uji menunjukkan penurunan yang cukup besar di beberapa skenario.

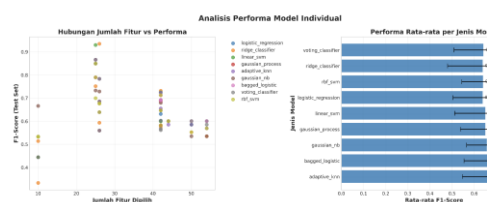
Temuan yang paling krusial adalah kinerja model pada metrik Recall (Berisiko). Pada 8 dari 15 skenario, model sama sekali gagal mengidentifikasi satu pun mahasiswa berisiko (Recall = 0.000). Kegagalan ini terutama terjadi pada mata kuliah dengan jumlah sampel yang lebih besar seperti Desain Basis Data dan pada mata kuliah terkecil Intelligent Tutoring System.

Satu-satunya mata kuliah di mana model menunjukkan performa deteksi yang menjanjikan adalah Statistika & Probabilitas, dengan Recall (Berisiko) konsisten sebesar 0.500 di semua titik waktu. Hal ini mungkin disebabkan oleh karakteristik mata kuliah tersebut yang lebih kuantitatif, di mana performa pada kuis dan tugas lebih langsung mencerminkan pemahaman mahasiswa.

2. Analisis Karakteristik Model Individual

Untuk memahami lebih mendalam bagaimana setiap algoritma pembelajaran mesin berperforma dalam konteks deteksi mahasiswa berisiko, dilakukan analisis komprehensif terhadap 11 algoritma berbeda. Analisis ini memberikan wawasan penting tentang pola pemilihan fitur dan karakteristik performa dari setiap jenis algoritma.

Hasil analisis menunjukkan bahwa Gaussian Process paling sering terpilih sebagai model terbaik (5 dari 15 skenario), diikuti oleh Voting Classifier dan Logistic Regression (masing-masing 3 skenario). Temuan ini mengkonfirmasi bahwa algoritma non-parametrik dan ensemble methods lebih efektif dalam menangani dataset kecil dan tidak seimbang. Menariknya, Gaussian Process menunjukkan konsistensi tinggi dengan kemampuan menangkap pola non-linear yang kompleks dalam data pembelajaran mahasiswa, terutama terlihat dari dominasinya pada berbagai mata kuliah dan titik waktu.



Gambar 7. Analisis Performa Model Individual

Analisis mendalam terhadap pola pemilihan fitur mengungkapkan wawasan penting tentang bagaimana setiap algoritma mengidentifikasi mahasiswa berisiko. Hanya Ridge Classifier yang melakukan seleksi fitur aktif, sementara 8 algoritma

lainnya menggunakan seluruh fitur yang tersedia dengan mekanisme penimbangan internal masing-masing.

Ridge Classifier menunjukkan preferensi yang jelas terhadap fitur-fitur meta-statistik dan interaksi aktivitas-performa dari total 61–76 fitur yang tersedia. Dari analisis seleksi fitur, algoritma ini secara konsisten memilih:

Fitur Meta-Statistik (Prioritas Tertinggi):

- feature_range (93.3%) - mengukur rentang variabilitas perilaku mahasiswa
- feature_max (93.3%) - mendeteksi nilai ekstrem dalam pola pembelajaran
- feature_std (86.7%) - mengidentifikasi inkonsistensi behavioral
- feature_median (73.3%) - menilai tendensi sentral perilaku belajar

Fitur Interaksi Aktivitas-Performa (Prioritas Tinggi):

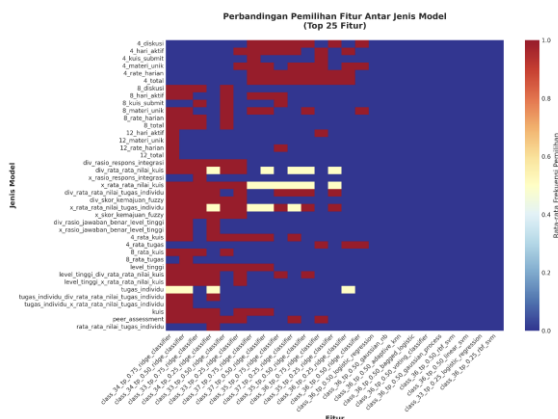
- jumlah_log_aktivitas_x_rata_rata_nilai_kuis (80%) - deteksi ketidakselarasan effort-outcome
- jumlah_log_aktivitas_x_rata_rata_nilai_tugas_individu (73.3%) - korelasi aktivitas dengan performa tugas
- jumlah_log_aktivitas_x_skor_kemajuan_fuzzy (67%) - hubungan aktivitas dengan progress pembelajaran

Fitur Behavioral dan Temporal (Prioritas Sedang):

- engagement_score (73.3%) - tingkat keterlibatan komprehensif.
- learning_depth_score (60%) - kedalaman eksplorasi materi.
- aktivitas_minggu_1_4_* - pola

pembelajaran awal semester.

Pola seleksi ini mengindikasikan bahwa Ridge Classifier mengidentifikasi mahasiswa berisiko melalui deteksi anomali behavioral: mahasiswa dengan variabilitas tinggi dalam engagement, ketidakselarasan antara aktivitas dan achievement, atau pola ekstrem dalam dimensi pembelajaran tertentu. Algoritma ini tidak hanya melihat performa akademik langsung, tetapi fokus pada konsistensi dan koherensi pola perilaku belajar secara keseluruhan.



Gambar 8. Peta Panas Perbandingan Pemilihan Fitur Antar Jenis Model

Sebelas algoritma lainnya (Logistic Regression, Linear SVM, Gaussian Process, dll.) menggunakan seluruh fitur dengan pendekatan yang berbeda:

1. **Logistic Regression & Linear SVM:** Mengandalkan kombinasi linear dari 61–76 fitur dengan koefisien yang mengindikasikan pentingnya relatif. Algoritma ini memberikan bobot tinggi pada fitur performa langsung seperti `rata_rata_nilai_kuis`, `rata_rata_nilai_tugas_individu`, dan fitur aktivitas volume seperti `jumlah_log_aktivitas`. Mahasiswa berisiko

diidentifikasi melalui kombinasi linear rendahnya nilai kuis, minimnya aktivitas LMS, dan tingginya `persentase_tugas_terlambat`.

2. **Gaussian Process:** Memanfaatkan seluruh 61 fitur melalui fungsi kernel RBF yang menangkap interaksi non-linear kompleks. Algoritma ini unggul dalam mendeteksi pola tersembunyi seperti mahasiswa dengan `jumlah_log_aktivitas` tinggi namun `learning_depth_score` rendah, atau inkonsistensi antara `aktivitas_minggu_1_4` dan `aktivitas_minggu_5_8`. Kemampuan menangkap interdependensi fitur temporal membuatnya sangat efektif untuk deteksi mahasiswa dengan pola pembelajaran yang kompleks.
3. **Adaptive KNN:** Menggunakan seluruh fitur dalam ruang 61-dimensi untuk menghitung similarity dengan normalisasi yang menekankan fitur `engagement_score`, `social_learning_score`, dan `academic_performance_index`. Mahasiswa berisiko diidentifikasi berdasarkan kedekatan Euclidean dengan cluster mahasiswa berisiko historis, terutama dalam dimensi fitur behavioral dan temporal.
4. **Bagged Logistic:** Mengombinasikan multiple model logistic yang masing-masing dilatih pada subset fitur bootstrap dari 61 fitur. Setiap model dalam ensemble fokus pada aspek berbeda: beberapa menekankan fitur kognitif (`rasio_jawaban_benar_level_tinggi`, `rasio_respons_integrasi`), lainnya pada fitur temporal atau interaksi. Agregasi voting meningkatkan robustness deteksi dengan mengurangi bias individual model terhadap subset fitur tertentu.

5. Gaussian NB: Memodelkan distribusi probabilitas setiap fitur independen per kelas. Fitur kategorikal seperti `has_peer_assessment` dan kontinyu seperti `skor_kemajuan_fuzzy` diperlakukan sebagai evidence terpisah. Mahasiswa berisiko diidentifikasi ketika kombinasi likelihood dari fitur-fitur individual menghasilkan probabilitas tinggi untuk kelas 'Berisiko'.

Berdasarkan analisis pola fitur, sistem dapat mengidentifikasi berbagai tipe mahasiswa berisiko:

1. **Tipe Inkonsisten (Ridge Classifier):** Mahasiswa dengan pola perilaku belajar yang tidak stabil, variabilitas tinggi dalam engagement, atau ketidakselarasan aktivitas-performa.
2. **Tipe Linear-Predictable (Logistic/SVM):** Mahasiswa dengan kombinasi faktor risiko yang dapat diprediksi secara linear, seperti rendahnya aktivitas forum dikombinasikan dengan nilai kuis rendah.
3. **Tipe Kompleks-Non-linear (Gaussian Process):** Mahasiswa dengan pola risiko yang melibatkan interaksi kompleks antar faktor, sulit diidentifikasi dengan metode linear tradisional.
4. **Tipe Similarity-Based (KNN):** Mahasiswa yang menunjukkan kesamaan pola dengan mahasiswa berisiko sebelumnya, meskipun tidak memiliki indikator risiko yang jelas secara individual.

Analisis ini mengonfirmasi bahwa pendekatan multi-algoritma memberikan cakupan deteksi yang lebih komprehensif, karena setiap algoritma memiliki sensitivitas terhadap tipe risiko yang berbeda. Temuan bahwa mayoritas algoritma menggunakan seluruh fitur yang tersedia menunjukkan bahwa:

1. Kompleksitas Fitur Optimal: Dengan rata-rata 30.5 fitur akhir setelah feature engineering, sebagian besar algoritma menganggap semua fitur memberikan kontribusi prediktif, yang memvalidasi proses feature engineering yang telah dilakukan.

2. Spesialisasi Algoritma: Ridge Classifier menunjukkan perilaku unik dengan fokus pada fitur statistik agregat, yang mencerminkan bias algoritmik terhadap regularization dan interpretabilitas linear.

3. Stabilitas Ensemble: Performa superior Bagged Logistic mengonfirmasi bahwa teknik ensemble efektif dalam konteks educational data mining dengan dataset terbatas.

Analisis ini memperkuat kesimpulan bahwa pendekatan multi-algoritma dengan optimasi per-kelas memberikan fleksibilitas yang diperlukan untuk menangani heterogenitas karakteristik pembelajaran pada berbagai mata kuliah, meskipun keterbatasan data tetap menjadi tantangan fundamental.

IV. KESIMPULAN DAN SARAN

A. Kesimpulan

Penelitian ini berhasil merancang dan membangun sebuah sistem deteksi dini mahasiswa berisiko dengan arsitektur 3-tier yang fungsional. Pendekatan optimasi per-kelas yang diterapkan juga tervalidasi, menunjukkan bahwa model machine learning yang optimal bervariasi untuk setiap mata kuliah. Namun, evaluasi pada data uji mengungkap tantangan fundamental. Meskipun skor validasi silang tinggi, terjadi overfitting yang signifikan, di mana performa model menurun drastis pada data baru. Temuan paling krusial adalah kegagalan model dalam mendeteksi kelas minoritas 'Berisiko', yang tercermin dari metrik Recall (Berisiko) yang sangat rendah,

bahkan mencapai 0.00 pada 8 dari 15 skenario. Kinerja deteksi terbaik hanya dicapai pada mata kuliah Statistika & Probabilitas dengan Recall 0.50. Akibatnya, efektivitas praktis sistem untuk tujuan intervensi dini sangat terbatas karena kendala dataset yang kecil dan tidak seimbang.

B. Saran

Berdasarkan temuan penelitian, pengembangan selanjutnya sebaiknya difokuskan pada penanganan ketidakseimbangan kelas dan keterbatasan data. Teknik oversampling seperti SMOTE dapat digunakan untuk meningkatkan deteksi kelas 'Berisiko'. Pendekatan yang lebih efektif adalah membangun model umum dengan menggabungkan data dari beberapa mata kuliah, sehingga menghasilkan dataset yang lebih besar dan model yang lebih stabil. Untuk saat ini, sistem sebaiknya diterapkan secara terbatas pada mata kuliah dengan performa model yang baik, seperti Statistika & Probabilitas. Di luar itu, hasil prediksi perlu diinterpretasikan secara hati-hati—model lebih tepat digunakan untuk mengonfirmasi mahasiswa yang aman daripada menentukan secara pasti siapa yang berisiko.

DAFTAR PUSTAKA

- [1] R. Zeng, L.-R. Xiang, J. Zeng and C. Zuo, "Applying team-based learning of diagnostics for undergraduate students: Assessing teaching effectiveness by a randomized controlled trial study," *Advances in Medical Education and Practice*, p. 211–218, 2017.
- [2] M. Anshari, M. N. Almunawar, M. Shahrill, D. Wicaksono and M. Huda, "Smartphones usage in the classrooms: Learning aid or interference?," *Education and Information Technologies*, vol. 22, no. 6, p. 3063–3079, 2017.
- [3] J. S. Mendoza, B. C. Pody, S. Lee, M. Kim and I. M. McDonough, "The effect of cellphones on attention and learning: The influences of time, distraction, and nomophobia," *Computers in Human Behavior*, vol. 86, pp. 52–60, 2018.
- [4] M. Dillon and E. C. Kokkelenberg, "The effects of class size on student achievement in higher education: Applying an earnings function," 2002.
- [5] Romero, Cristobal, Ventura and Sebastian, "Educational data mining and learning analytics: An updated survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 10, p. 3, 2020.
- [6] Pelima, R. Lidya, Sukmana and Yuda, "Predicting University Student Graduation Using Academic Performance and Machine Learning: A Systematic Literature Review," in *IEEE Access*, IEEE, 2024, pp. 23451–23465.
- [7] Riestra-Gonz, "Massive LMS log data analysis for the early prediction of course-agnostic student performance," *Computers & Education*, vol. 163, 2021.
- [8] Albreiki, Balqis, Nazar and Hany, "A Systematic Literature Review of Student' Performance Prediction Using Machine Learning Techniques," *Education Sciences*, vol. 11, p. 552, 2021.
- [9] Yagci and Mustafa, "Educational data mining: prediction of students' academic performance using machine learning algorithms," *Smart Learning Environments*, vol. 9, 2022.
- [10] Bogнар and Fauszt, "Different learning predictors and their effects for Moodle Machine Learning models," in *2020 11th IEEE International Conference on Cognitive Infocommunications (CogInfoCom)*, pp. 405–410.
- [11] Pongpaichet, Siripen and Sawarin, "Early Detection At-Risk Students using Machine Learning," in *2020 International Conference on ICT and Knowledge Engineering (ICTKE)*, 2020, pp. 283–287.
- [12] Abdullah, Malak, Mahmoud, Farah and Saif, "Predicting students' academic performance using e-learning logs," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 12, pp. 831–839, 2023.
- [13] Quinn, Rory and Geraldine, "Prediction of student academic performance using Moodle data from a Further Education setting," *Irish Journal of Technology Enhanced Learning*, vol. 5, 2020.
- [14] Saqr, L. p.-P. Mohammed and Sonsoles, "The longitudinal trajectories of online engagement over a full program," *Computers & Education*, vol. 175, 2021.
- [15] Agusriandi, A. Sitanggang, I. Sukaesih and S. Hartono, "Student Performance Based on Activity Log on Social Network and e-Learning," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 10, 2020.
- [16] Pallathadka, Harikumar and Alex, "Classification and prediction of student performance data using various machine learning algorithms," in *Materials Today: Proceedings*, 2021.
- [17] Chitti, Manjari and Padmini, "Need for Interpretable Student Performance Prediction," *2020 13th International Conference on Developments in eSystems Engineering (DeSE)*, p. 2020, 269–272.
- [18] Yang, Chunsheng and Feng-Kuang, "Machine Learning-Based Student Modeling Methodology for Intelligent Tutoring Systems," *Journal of Educational Computing Research*, vol. 59, pp. 1–21, 2021.
- [19] Xu, Zhuojia, Hua and Qishan, "Student Performance Prediction Based on Blended Learning," *IEEE Transactions on Education*, vol. 64, 2020.
- [20] Mubarak, A. Ali and S. AM, "Predictive learning analytics using deep learning model in MOOCs' courses videos," *Education and Information Technologies*, vol. 26, 2020.
- [21] IBM, "Pengantar CRISP-DM," 2022. [Online]. Available: <https://www.ibm.com/docs/en/spss-modeler/SaaS?topic=guide-introduction-crisp-dm>.
- [22] Doabler, T. Christian and Ben, "Kindergarteners at Risk for Severe Mathematics Difficulties: Investigating Tipping Points of Core Mathematics Instruction," *Journal of Learning Disabilities*, vol. 54, pp. 119–132, 2020.