AnoaTIK: Jurnal Teknologi Informasi dan Komputer e-ISSN: **2987-7652**
Vol. 4, No. 1, Juni 2026 Hal. 45-57

KOMPARASI ALGORITMA XGBOOST DAN ADABOOST UNTUK ANALISIS KLASIFIKASI KEPUASAN MAHASISWA ILMU KOMPUTER FMIPA UNIVERSITAS HALU OLEO TERHADAP PEMBELAJARAN DARING

Jejen[*1], Budi Wijaya Rauf[2], Muhammad Riansyah Tohamba[3]

[1,2,3]Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Halu Oleo, Kendari
Email: [*1]jejen8238@gmail.com, [2]budiwijaya@uho.ac.id, [3]muh.riansyaht@uho.ac.id
***** *Penulis Korespondensi*

Abstrak

Penelitian ini bertujuan untuk mengetahui hasil komparasi dua algoritma *machine learning*, yaitu *Extreme Gradient Boosting* (XGBoost) dan *Adaptive Boosting* (AdaBoost), dalam mengklasifikasikan tingkat kepuasan mahasiswa Ilmu Komputer FMIPA Universitas Halu Oleo terhadap pembelajaran daring. Pembelajaran daring telah menjadi salah satu inovasi penting dalam bidang pendidikan, dengan memanfaatkan teknologi informasi dan komunikasi sebagai media utama dalam proses pembelajaran. Efektivitas pembelajaran daring sangat dipengaruhi oleh berbagai faktor, seperti pemahaman materi, kemudahan interaksi, kemudahan pengumpulan tugas, pemanfaatan *e-learning*, fasilitas mengikuti kuliah daring, dan koneksi jaringan. Data diperoleh melalui kuesioner yang diisi oleh 80 mahasiswa Program Studi Ilmu Komputer FMIPA Universitas Halu Oleo. Pengujian kinerja algoritma menggunakan sistem analisis klasifikasi kepuasan mahasiswa. Parameter utama yang diuji berupa *learning rate* dan *n_estimators*. Evaluasi performa model dilakukan menggunakan metrik klasifikasi seperti akurasi, presisi, recall, F1-score, dan *Area Under the Curve* (AUC). Hasil pengujian menunjukkan bahwa AdaBoost mencapai AUC tertinggi sebesar 0.964 pada skenario II (70:30) dengan parameter *learning rate* 0.3 dan *n_estimator* 200. Sementara, XGBoost memperoleh AUC sebesar 0.914 pada skenario II (70:30) dengan parameter *larning rate* 0.1, 0.2 dan 0.3 pada seluruh variasi jumlah *n_estimator*. Hasil AUC tersebut menunjukkan bahwa algoritma AdaBoost lebih unggul dibandingkan XGBoost..

Kata kunci: Klasifikasi, kepuasan mahasiswa, pembelajaran daring, XGBoost, AdaBoost, data mining, *machine learning*

Abstract

This study aims to compare the performance of two machine learning algorithms, namely Extreme Gradient Boosting (XGBoost) and Adaptive Boosting (AdaBoost), in classifying the level of student satisfaction in the Computer Science Department, Faculty of Mathematics and Natural Sciences, Halu Oleo University, with regard to online learning. Online learning has become one of the significant innovations in the field of education, utilizing information and communication technology as the primary medium in the learning process. The effectiveness of online learning is highly influenced by various factors, such as material comprehension, ease of interaction, ease of assignment submission, utilization of e-learning, facilities for attending online lectures, and network connectivity. Data were obtained through questionnaires filled out by 80 students of the Computer Science Study Program, FMIPA, Halu Oleo University. Algorithm performance testing was conducted using a student satisfaction classification analysis system. The main parameters tested were learning rate and n_estimators. Model performance evaluation was carried out using classification metrics such as accuracy, precision, recall, F1-score, and Area Under the Curve (AUC). The test results show that AdaBoost achieved the highest AUC of 0.964 in Scenario II (70:30) with a learning rate parameter of 0.3 and n_estimators of 200. Meanwhile, XGBoost obtained an AUC of 0.914 in Scenario II (70:30) with learning rate parameters of 0.1, 0.2, and 0.3 across all variations of n_estimators. These AUC results indicate that the AdaBoost algorithm outperforms XGBoost.

***Keywords*:** *Classification, student satisfaction, online learning, XGBoost, AdaBoost, data mining, machine learning*

1. PENDAHULUAN

Perkembangan teknologi informasi yang pesat di era digital telah membawa dampak signifikan terhadap cara manusia mengakses, menyimpan, dan mengelola informasi. Setiap aktivitas yang dilakukan secara digital menghasilkan jejak data yang terus bertambah dari waktu ke waktu. Namun, data yang melimpah tersebut tidak akan memberikan manfaat yang maksimal tanpa adanya proses pengolahan yang tepat. Akibatnya, jumlah data yang tersedia semakin besar dan kompleks, sehingga memunculkan tantangan dalam hal pengolahan serta pemanfaatannya secara efektif [3].

Data mining menjadi salah satu teknik penting dalam mengolah dan menggali informasi dari kumpulan data yang besar. Tujuan utama dari data mining adalah mengidentifikasi pola, hubungan, atau informasi yang mungkin tidak terlihat secara langsung dalam data. Teknik data mining yang digunakan untuk mengekstrak pola dan pengetahuan dari data yaitu klasifikasi (*classification*), regresi *(regression*), klastering (*clustering*), asosiasi (*association*). Proses data mining melibatkan penggunaan berbagai teknik statistik, matematis dan kecerdasan buatan. Data mining memiliki peran penting dalam membantu berbagai bidang seperti ekonomi, sosial, politik, dan pendidikan untuk pengambilan keputusan berdasarkan data yang lebih akurat dan mendalam [6].

Salah satu pendekatan utama yang digunakan dalam data mining adalah *machine learning*. Machine learning merupakan cabang dari kecerdasan buatan (*Artificial Intelligence*) yang memiliki kemampuan untuk belajar dari data tanpa diprogram secara eksplisit. Tujuan utama dari machine learning adalah untuk membangun model yang dapat memberikan prediksi atau menggambarkan pola dari data yang diberikan [4]. Dalam penelitian ini, machine learning digunakan untuk membangun model klasifikasi yang dapat mengelompokkan tingkat kepuasan mahasiswa terhadap pembelajaran daring berdasarkan data hasil kuesioner yang dikumpulkan. Dengan teknik klasifikasi ini, model dapat memberikan wawasan mengenai faktor-faktor apa saja yang memengaruhi kepuasan mahasiswa, serta sejauh mana efektivitas metode pembelajaran daring yang diterapkan.

Terdapat berbagai macam algoritma *machine learning* yang dapat digunakan untuk tugas klasifikasi, namun algoritma *Extreme Gradient Boosting* (XGBoost) dan *Adaptive Boosting* (AdaBoost) menjadi dua di antara yang paling populer dan efektif. Kedua algoritma tersebut merupakan bagian dari teknik *ensemble learning*, namun memiliki pendekatan yang berbeda dalam membangun model *ensemble*. XGBoost memiliki kinerja yang tinggi, tetapi membutuhkan penyesuaian parameter yang lebih kompleks dan biasanya bekerja efektif pada dataset yang besar. Sementara itu, AdaBoost memiliki struktur yang lebih sederhana dan tidak memerlukan pengaturan parameter yang rumit, sehingga lebih mudah diterapkan [5]. *Komparasi* antara kedua algoritma tersebut dilakukan untuk mengetahui tingkat keakuratan algoritma dan keunggulan masing-masing dalam menyelesaikan masalah klasifikasi seperti analisis kepuasan mahasiswa.

Pembelajaran daring (*e-learning*) menjadi salah satu inovasi dalam bidang pendidikan yang mengandalkan pemanfaatan teknologi informasi dan komunikasi sebagai media utama dalam proses pembelajaran. Pembelajaran *e-learning* ini mempunyai kelebihan dan kekurangan bila dibandingkan dengan pembelajaran tatap muka. Kelebihan dari pembelajaran *e-learning* antara yaitu dapat dilakukan dari jarak jauh, artinya mahasiswa dapat melakukan pembelajaran dari rumah dan tidak harus berada di lingkungan perguruan tinggi. Kekurangan pembelajaran *e-learning* yaitu membutuhkan biaya yang mahal karena setiap mahasiswa harus mempunyai perangkat keras berupa *handphone* ataupun komputer untuk dapat melakukan pembelajaran [2].

Pembelajaran daring memberikan kemudahan bagi mahasiswa dalam mengakses materi ajar, berinteraksi dengan dosen, serta mengikuti aktivitas perkuliahan melalui berbagai platform digital. Namun, meskipun menawarkan sejumlah kemudahan, efektivitas pembelajaran daring tidak terlepas dari berbagai faktor yang saling berkaitan. Beberapa faktor yang memengaruhi keberhasilan pembelajaran daring antara lain adalah keaktifan dosen, media pembelajaran, jaringan internet, waktu belajar, dan metode pembelajaran. Apabila salah satu faktor tersebut tidak terpenuhi dengan baik, maka kemungkinan besar akan berdampak pada menurunnya tingkat kepuasan dan hasil belajar mahasiswa [1].

2. METODE PENELITIAN

2.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan mulai bulan Juni 2025 sampai dengan bulan September 2025. Tempat penelitian ini dilakukan di Program Studi Ilmu Komputer, Fakultas Matematikan dan Ilmu Pengetahuan Alam, Universitas Halu Oleo.

2.2 Instrumen Penelitian

Instrumen penelitian yang digunakan dalam penelitian ini yaitu perangkat keras (*hardware*) dan perangkat lunak (*software*). Adapun instrumen penelitian yang digunakan untuk penelitian ini disebutkan pada Tabel 1.

Tabel 1. Instrumen Penelitian

Perangkat Keras	Perangkat Lunak
Laptop Acer Processor AMD Ryzen 3 3250U with Radeon Graphics 2.60 GHz RAM 12.0 GB (9.94 GB usable)	*Google Colaboratory*
	Visual Studio Code versi 1.96.3
	Python
	Streamlit

2.3 Variabel Penelitian

1. Variabel Independen (X)

Variabel independen dalam penelitian ini yaitu variabel yang mempengaruhi atau menjadi penyebab perubahan pada variabel lain. Variabel ini disebut juga variabel bebas, karena variabel ini dapat dimanipulasi, dikontrol, atau dipilih untuk melihat dampaknya terhadap variabel dependen (terikat). Variabel independen pada penelitian ini dapat dilihat pada Tabel 2.

Tabel 2. Variabel Independen (X)

Variabel	Skala Likert
Pemahaman Materi (X1)	Sangat Tidak Paham (1)
	Tidak Paham (2)
	Cukup Paham (3)
	Paham (4)
	Sangat Paham (5)
Kemudahan Interaksi (X2)	Sangat Tidak Setuju (1)
	Tidak Setuju (2)
	Netral (3)
	Setuju (4)
	Sangat Setuju (5)
Kemudahan Pengumpulan Tugas (X3)	Sangat Sulit (1)
	Sulit (2)
	Cukup (3)
	Mudah (4)
	Sangat Mudah (5)
Pemanfaatan *E-learning* (X4)	Tidak Pernah (1)
	Jarang (2)
	Kadang-kadang (3)
	Sering (4)
	Selalu (5)
Fasilitas Mengikuti Kuliah Daring (X5)	Sangat Tidak Memadai (1)
	Tidak Memadai (2)
	Cukup Memadai (3)
	Memadai (4)
	Sangat Memadai (5)
Koneksi Jaringan (X6)	Sangat Sering Terjadi (1)
	Sering (2)
	Kadang-kadang (3)
	Jarang (4)
	Tidak Pernah (5)

2. Variabel Dependen (Y)

Variabel dependen dalam penelitian ini yaitu variabel yang dipengaruhi atau dijelaskan oleh variabel lain. Variabel ini disebut juga variabel terikat, karena nilainya bergantung pada perubahan yang terjadi pada variabel independen (variabel bebas). Variabel dependen pada penelitian ini dapat dilihat pada Tabel 3.

Tabel 3. Variabel Dependen (Y)

Variabel	Skala Likert
Kepuasan Mahasiswa (Y)	Tidak Puas (0)
	Puas (1)

2.4 Pengolahan dan Analisis Data

1. *Data Cleaning*

Data cleaning atau pembersihan data dalam penelitian ini berfungsi untuk membersihkan data mentah (*raw data*) dari kesalahan dan ketidaksesuaian sebelum dianalisis atau digunakan dalam algoritma klasifikasi seperti *Extreme Gradient Boosting* dan *Adaptive Boosting*. Data mentah yang diperoleh dari hasil survei kepuasan mahasiswa sering kali mengandung *missing value*. Oleh karena itu, proses pembersihan data mencakup penanganan nilai kosong sangat penting untuk memastikan bahwa data yang digunakan dalam pelatihan model benar-benar bersih, representatif, dan konsisten.

2. Trasformasi Data

Penelitian ini terdapat beberapa fitur atau atribut dari data kepuasan mahasiswa terhadap pembelajaran daring yang berisi nilai *non-numerik* (huruf/kategori) diubah menjadi bentuk numerik agar dapat diproses oleh algoritma XGBoost dan AdaBoost. Teknik yang digunakan untuk transformasi ini adalah *Label Encoding*, di mana setiap nilai unik dari variabel diubah menjadi angka.

3. *Split Data*

Split data dalam penelitian ini berfungsi membagi dataset menjadi beberapa bagian terpisah sebelum melakukan pelatihan (*training*) dan pengujian (t*esting*) model *machine learning*. Pembagian data dalam penelitian ini untuk masing-masing algoritma dengan rasio 60:40, 70:30 dan 80:20 di mana sebagian besar data digunakan untuk melatih model, sementara sisanya digunakan untuk menguji performa model. Proses ini bertujuan agar model tidak hanya menyesuaikan diri dengan data yang tersedia, tetapi juga mampu mengenali pola pada data yang belum pernah digunakan dalam proses pelatihan.

4. Perbandingan Kedua Model Machine Lerning

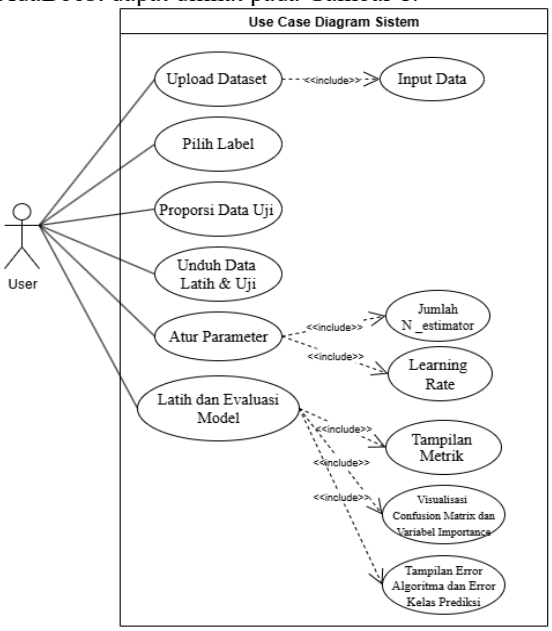
Penelitian ini dilakukan proses klasifikasi untuk menganalisis kepuasan mahasiswa terhadap pembelajaran daring menggunakan dua algoritma *machine learning* berbasis *ensemble*, yaitu XGBoost dan AdaBoost. Proses analisis dimulai dengan pembagian dataset menjadi dua bagian, yaitu data pelatihan (*training*) dan data pengujian (*testing*). Data pelatihan digunakan untuk membangun model masing-masing algoritma, sedangkan data pengujian digunakan untuk mengevaluasi kinerja model yang telah dibentuk. Pada tahap pemodelan menggunakan data pelatihan, dilakukan perhitungan kesalahan prediksi model yang berbeda untuk masing-masing algoritma, yaitu *log loss* untuk XGBoost dan *exponential loss* untuk AdaBoost. Selain itu, untuk menganalisis lebih dalam tingkat kesalahan prediksi masing-masing kelas, digunakan *class-wise error rate* yang menghitung kesalahan prediksi secara spesifik.

Fokus perbandingan antara XGBoost dan AdaBoost dilakukan dengan mengamati pengaruh konfigurasi dari dua parameter, yaitu *learning rate* dan *n_estimators* terhadap kinerja model. Parameter *learning rate* berfungsi untuk mengontrol kontribusi

setiap model dalam proses *boosting*, sementara *n_estimators* menentukan jumlah pohon atau *iterasi* yang digunakan dalam membangun model *ensemble.* Dengan variasi nilai pada kedua parameter ini, dilakukan pengujian terhadap kinerja masing-masing algoritma menggunakan metrik evaluasi seperti akurasi, precision, recall, F1-score dan AUC, sehingga dapat diketahui algoritma mana yang memberikan performa terbaik.

5. Sistem Analisis

Tahapan perhitungan pada penelitian ini adalah dengan mengunakan sistem analisis klasifikasi kepuasan mahasiswa. Tahapan-tahapan dalam mengunakan sistem perbandingan XGBoost dan AdaBoost dapat dilihat pada Gambar 1.



Gambar 1. Use Case Diagram Sistem

Penjelasan mengenai tahapan-tahapan dalam menggunakan sistem analisis yaitu:

a. *User* (Pengguna) yaitu individu yang berinteraksi langsung dengan sistem untuk menjalankan fungsionalitas yang tersedia.
b. Upload dataset yaitu mengunggah file CSV berisi data kepuasan mahasiswa. Setelah file berhasil diunggah, sistem secara otomatis melakukan verifikasi terhadap struktur dan kebersihan data tersebut. Proses ini mencakup pemeriksaan apakah dataset tidak kosong, tidak mengandung nilai yang hilang (*missing value*), serta memastikan bahwa setiap kolom memiliki tipe data yang valid dan dapat diproses lebih lanjut oleh model klasifikasi.
c. Pilih label yaitu pengguna memilih kolom hasil klasifikasi dalam dataset yang telah diupload akan menjadi tujuan klasifikasi prediksi model *machine learning*. Selanjutnya, sistem akan menampilkan visualisasi distribusi label target dalam bentuk diagram batang.
d. Proporsi data uji yaitu pengguna menentukan persentase data uji berdasarkan proporsi yang ditentukan. Penelitian ini membagi data latih dan data uji menjadi 60:40, 70:30 dan 80:20.
e. Unduh data latih dan uji yaitu setelah pembagian data, pengguna dapat mengunduh file CSV untuk data latih dan uji.
f. Atur parameter yaitu pengguna mengatur nilai *learning rate* dan jumlah *n_estimators* sebagai parameter model. Penelitian ini menggunakan *learning rate* 0.1, 0.2 dan 0.3 dan jumlah *n_estimators* 50, 100, 200.
g. Latih dan Evaluasi model yaitu sistem melatih model XGBoost dan AdaBoost berdasarkan data latih dan parameter yang dipilih, kemudian melakukan evaluasi terhadap data uji. Dilatih dan evaluasi model pengguna dapat melihat *log loss* XGBoost, *exponential loss* AdaBoost, tampilan metrik evaluasi seperti akurasi, presisi, recall, F1-Score, AUC dan visualisasi *confusion matrix*, serta *variabel importance*.

3. HASIL DAN PEMBAHASAN

3.1 Sistem Analisis

Sistem klasifikasi kepuasan mahasiswa terhadap pembelajaran daring dibangun menggunakan *Streamlit*, sebuah *framework* berbasis Python yang memungkinkan pembuatan antarmuka pengguna (UI) secara interaktif dan responsif untuk kebutuhan analisis data dalam *machine learning*. Sistem ini dirancang untuk menjalankan proses analisis klasifikasi mulai dari mengunggah dataset, memilih algoritma klasifikasi (XGBoost atau AdaBoost), mengatur parameter model, melatih model, serta melihat hasil evaluasi model secara langsung dan interaktif.

1. Tampilan Antarmuka Sistem

Antarmuka sistem dirancang sederhana dan interaktif, dengan menu navigasi yang ditempatkan di *sidebar*. Pada *sidebar*, pengguna dapat mengatur parameter penting seperti jumlah *n_estimator*, nilai *learning rate,* dan proporsi data uji. Penggunaan slider ini sangat membantu karena memudahkan pengguna dalam melakukan penyesuaian parameter model secara langsung tanpa perlu mengedit kode program. Tampilan antarmuka dan navigasi dapat dilihat pada Gambar 2.

Gambar 2 Tampilan Antarmuka Sistem

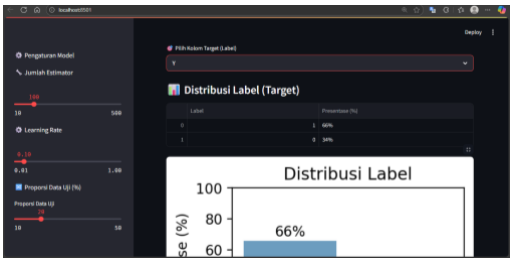
2. Proses Unggah dan Validasi Dataset

Langkah awal penggunaan sistem dimulai dengan mengunggah dataset berformat CSV. Setelah file diunggah, sistem secara otomatis melakukan validasi terhadap isi dataset. Validasi ini mencakup tiga aspek penting yaitu memastikan dataset tidak kosong atau hanya terdiri dari satu kolom, mendeteksi adanya nilai kosong (*missing values*), serta mengidentifikasi keberadaan nilai teks alfabet yang tidak sesuai dengan kebutuhan algoritma. Jika ditemukan masalah, aplikasi akan memberikan peringatan berupa pesan *error* yang menjelaskan letak dan jenis kesalahan, seperti baris dan kolom yang bermasalah. Tampilan proses unggah dan validasi dataset dapat dilihat pada Gambar 3.

Gambar 3. Tampilan Proses Unggah dan Validasi Dataset

3. Eksplorasi dan Visualisasi Data

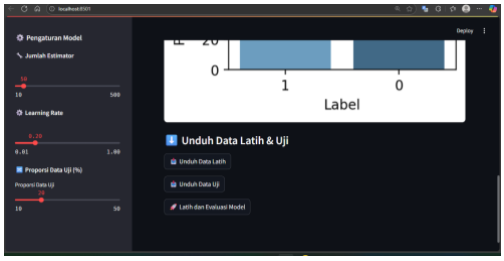
Setelah proses validasi berhasil, pengguna dapat melihat tampilan awal dari dataset dalam bentuk tabel. Kemudian, pengguna diminta untuk memilih kolom target (*label*) yang akan digunakan sebagai variabel klasifikasi. Kolom-kolom selain target secara otomatis akan dianggap sebagai fitur. Selanjutnya, sistem menampilkan distribusi label dalam bentuk tabel dan diagram batang. Tampilan eksplorasi dan visualisasi data dapat dilihat pada Gambar 4.



Gambar 4. Tampilan Eksplorasi dan Visualisasi Data

4. Pembagian Data Latih dan Uji

Setelah proses pemilihan target selesai, data akan dibagi menjadi dua bagian yaitu data latih dan data uji. Proses pembagian ini menggunakan fungsi *train_test_split* dari *library scikit-learn*. Besarnya proporsi data uji dapat ditentukan langsung oleh pengguna melalui komponen slider yang tersedia di sidebar. Sistem kemudian menyediakan tombol unduh untuk masing-masing bagian data, yaitu data latih dan data uji, dalam format CSV. Tampilan pembagian data latih dan uji dapat dilihat pada Gambar 5.

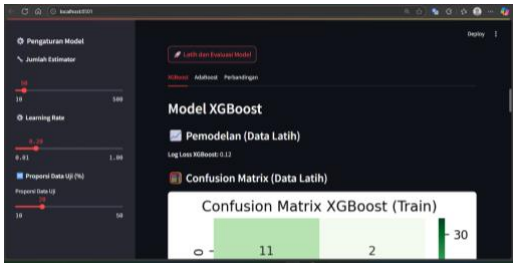


Gambar 5. Tampilan Pembagian Data Latih dan Uji

5. Pelatihan dan Evaluasi Model

Proses pelatihan model dilakukan setelah pengguna menekan tombol "Latih dan Evaluasi Model". Sistem akan membangun model menggunakan algoritma yang XGBoost dan AdaBoost, dengan parameter yang telah ditentukan sebelumnya. Setelah proses pelatihan menggunakan data latih selesai, model akan dievaluasi menggunakan data uji serta model akan melakukan perbandingan kedua algoritma.

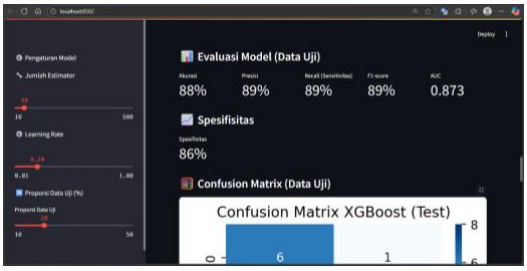
Evaluasi pada data latih untuk model AdaBoost, sistem menghitung *exponential loss*, yaitu sebagai ukuran kerugian yang memandu proses pembelajaran dengan menghitung rata-rata penalti atas kesalahan prediksi berdasarkan hubungan antara label sebenarnya dan gabungan keluaran seluruh model lemah. Sementara itu, untuk model XGBoost menggunakan *log loss* sebagai ukuran ketepatan prediksi probabilistik. Selain itu, sistem juga menampilkan *confusion matrix* untuk data latih dalam bentuk visualisasi *heatmap*, yang memudahkan pengguna dalam mengidentifikasi kesalahan klasifikasi antar kelas. Tidak hanya itu, sistem juga menyajikan tabel prediksi vs aktual yang memperlihatkan data asli dan hasil prediksi model, serta menandai apakah hasil prediksi tersebut benar atau salah. Analisis *error* per kelas juga disediakan untuk menunjukkan tingkat kesalahan klasifikasi pada masing-masing label. Fitur ini sangat membantu dalam memahami kelemahan model terhadap kelas tertentu. Fitur pemodelan dapat dilihat pada Gambar 6.



Gambar 6. Fitur Pemodelan

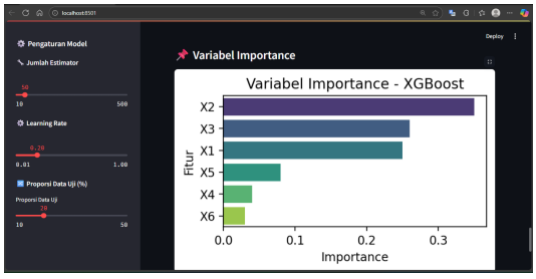
Setelah model dievaluasi pada data latih, langkah selanjutnya adalah evaluasi pada data uji. Proses ini bertujuan untuk mengetahui seberapa baik model dapat melakukan generalisasi terhadap data baru yang belum pernah dilihat sebelumnya. Evaluasi dilakukan menggunakan metrik seperti akurasi, presisi, recall, f1-score, dan AUC. Hasil evaluasi ini

ditampilkan secara lengkap untuk memudahkan perbandingan performa antar algoritma. *Confusion matrix* juga disajikan untuk data uji, yang dapat membantu pengguna mengevaluasi pola kesalahan prediksi model terhadap data yang lebih representatif. Sistem juga menyediakan tabel prediksi vs aktual untuk data uji ditampilkan serupa dengan data latih, termasuk penandaan hasil prediksi yang benar dan salah. Fitur evaluasi model dapat dilihat pada Gambar 7.



Gambar 7. Fitur Evaluasi Model

Sebagai tambahan sistem juga menampilkan visualisasi pentingnya masing-masing fitur terhadap hasil klasifikasi. Visualisasi ini disajikan dalam bentuk diagram batang *horizontal*, yang memperlihatkan fitur mana saja yang memberikan kontribusi paling besar terhadap keputusan model. Tampilan variabel *importance* dapat dilihat pada Gambar 8.



Gambar 8. Tampilan Variabel Importance

Bagian evaluasi perbandingan XGBoost dan AdaBoost menampilkan hasil evaluasi kedua model dalam bentuk tabel maupun grafik. Pada bagian tabel, menampilkan seluruh metrik evaluasi seperti akurasi, presisi, recall, dan F1-score, serta AUC. Selain tabel, sistem juga menampilkan grafik perbandingan yang memperlihatkan perbedaan nilai akurasi, presisi, recall, F1-score, serta AUC antara XGBoost dan AdaBoost. Visualisasi ini memberikan gambaran yang lebih jelas mengenai keunggulan dan kelemahan masing-masing algoritma. Selanjutnya, sistem secara otomatis menampilkan kesimpulan dengan menentukan model yang memiliki AUC terbaik. Berdasarkan hasil perhitungan, sistem juga memberikan keterangan apakah XGBoost memiliki performa yang lebih baik, AdaBoost lebih unggul, atau jika keduanya menunjukkan performa yang setara. Sistem juga menampilkan grafik dan heatmap perbandingan variabel *importance* pada kedua model dan memberikan kesimpulan variabel mana yang paling berpengaruh dalam prediksi model. Evaluasi perbandingan XGBoost dan AdaBoost dapat dilihat pada Gambar 9.

Gambar 9. Evaluasi Perbandingan XGBoost dan AdaBoost

3.2 *Data Preparation*

Data preparation penting dalam proses analisis data. Data yang disiapkan berupa data *excel* yang diperoleh dari responden mahasiswa Program Studi Ilmu Komputer FMIPA Universitas Halu Oleo. Data akan dibersihkan secara manual dengan mengambil hasil survei mahasiswa. Data yang sudah dimasukkan kedalam program akan dibersihkan dari nilai yang hilang dan mentransformasikan data dari teks menjadi numberik serta mengisi nilai yang kosong dengan nilai *modus*.

3.3 *Data Cleaning*

Hal yang dilakukan pertama pada tahap ini adalah mengatasi *missing value. Missing value* yaitu informasi yang tidak tersedia pada suatu atribut data. Pada data hasil survei kepuasan mahasiswa Ilmu Komputer terhadap pembelajaran daring FMIPA UHO, ditemukan beberapa *missing value*. Data hasil survei kepuasan mahasiswa Ilmu Komputer terhadap pembelajaran daring FMIPA UHO ditunjukan pada Gambar 10.

Kadang-kadang	Kadang-kadang	Kadang-kadang	Cukup Memadai	Cukup Memadai	Tidak Memadai
Sering	Sering	Sering	Sangat Memadai	Sangat Memadai	Memadai
Kadang-kadang	Sering	Sering	Sangat Memadai	Sangat Memadai	Sangat Memadai
Sering	Sering	Sering	Memadai	Cukup Memadai	Cukup Memadai
Sering	Sering	Sering	Sangat Memadai	Sangat Memadai	Sangat Memadai
Sering	Sering	Sering	Memadai	Memadai	Cukup Memadai
Kadang-kadang	Jarang	Sering	Sangat Memadai	Memadai	Cukup Memadai
Sering	Sering	Sering	Memadai	Cukup Memadai	Memadai
Sering	Sering	Sering	Cukup Memadai	Cukup Memadai	Cukup Memadai

Gambar 10. Data Penelitian

Data yang ditampilkan merupakan beberapa data yang mencakup hasil survei kepuasan mahasiswa Ilmu Komputer terhadap pembelajaran daring FMIPA UHO. Perlu diperhatikan data tersebut masih terdapat nilai yang hilang, untuk mengatasi hal tersebut, langkah yang dilakukan yaitu mengisi nilai yang kosong dengan nilai modus. Solusi untuk mengatasi permasalahan tersebut ditunjukkan pada Gambar 11.

Kadang-kadang	Kadang-kadang	Kadang-kadang	Cukup Memadai	Cukup Memadai	Tidak Memadai
Sering	Sering	Sering	Sangat Memadai	Sangat Memadai	Memadai
Kadang-kadang	Sering	Sering	Sangat Memadai	Sangat Memadai	Sangat Memadai
Sering	Sering	Sering	Memadai	Cukup Memadai	Cukup Memadai
Sering	Sering	Sering	Sangat Memadai	Sangat Memadai	Sangat Memadai
Sering	Sering	Sering	Memadai	Memadai	Cukup Memadai
Sering	Sering	Sering	Memadai	Memadai	Memadai
Kadang-kadang	Jarang	Sering	Sangat Memadai	Memadai	Cukup Memadai
Sering	Sering	Sering	Memadai	Cukup Memadai	Memadai
Sering	Sering	Sering	Cukup Memadai	Cukup Memadai	Cukup Memadai

Gambar 11. Pembersihan Data

3.4 Transformasi Data

Setelah tahap penanganan missing value dilakukan untuk memastikan tidak ada data yang kosong atau hilang, langkah berikutnya adalah melakukan transformasi data teks menjadi angka. Proses ini sangat penting karena algoritma XGBoost dan AdaBoost hanya dapat memproses data numerik, bukan data dalam bentuk teks. Data hasil transformasi dapat dilihat pada Gambar 12.

X1			X2			X3			X4			X5			X6		
3	3	2	4	4	5	4	5	3	3	3	3	3	3	2	2	3	2
3	3	3	3	4	4	4	4	4	4	4	4	5	5	4	3	2	3
3	2	3	4	3	4	5	4	5	3	4	4	5	5	5	4	3	4
4	3	3	3	4	4	4	4	4	4	4	4	4	3	3	3	3	3
3	4	2	3	4	4	5	4	4	4	4	4	5	5	5	5	5	3
3	3	3	3	3	3	3	4	3	4	4	4	4	4	3	2	2	1
1	1	2	3	3	3	1	4	3	4	4	4	4	4	4	3	3	3
3	4	2	3	4	4	4	4	3	3	2	4	5	4	3	2	1	2
4	4	3	4	4	4	4	4	4	4	4	4	4	3	4	3	3	3
3	3	2	3	3	4	3	3	3	4	4	4	3	3	3	2	2	3

Gambar 12. Hasil Transformasi Data

Transformasi data dilakukan agar semua nilai variabel berada dalam bentuk numerik, kemudian dilanjutkan dengan perhitungan nilai rata-rata pada masing-masing variabel. Langkah ini membantu memastikan bahwa data yang digunakan memiliki kualitas yang baik dan siap diproses lebih lanjut. Hasil rata-rata nilai pada masing-masing variabel dapat dilihat pada Gambar 13.

X1	X2	X3	X4	X5	X6
2.666667	4.333333	4	3	2.666667	2.333333
3	3.666667	4	4	4.666667	2.666667
2.666667	3.666667	4.666667	3.666667	5	3.666667
3.333333	3.666667	4	4	3.333333	3
3	3.666667	4.333333	4	5	4.333333
3	3	3.333333	4	3.666667	1.666667
1.333333	3	2.666667	4	4	3
3	3.666667	3.666667	3	4	1.666667
3.666667	4	4	4	3.666667	3
2.666667	3.333333	3	4	3	2.333333

Gambar 13. Hasil Rata-rata Nilai Variabel

3.5 Pelabelan Data

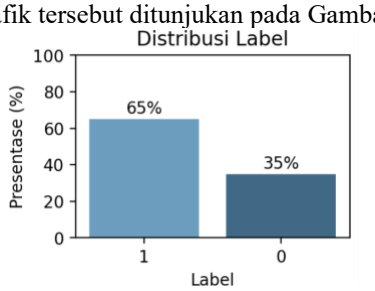
Pembelahan data dilakukan dengan perubahan nama kolom dengan nama kolom "Y" yang berisi nilai rata-rata dari setiap baris dalam data yang dibulatkan dan dikonversi jadi integer. Selain itu atribut lainnya juga diganti dengan nama kolom menjadi X1 (Pemahaman Materi), X2 (Kemudahan Interaksi), X3 (Kemudahan Pengumpulan Tugas), X4 (Pemanfaatan *E-learning*), X5 (Fasilitas Mengikuti Kuliah Daring), X6 (Koneksi Jaringan) yang masing-masing merepresentasikan variabel dalam data tersebut. Tahap ini hasilnya ditunjukan pada Gambar 14.

X1	X2	X3	X4	X5	X6	Y
3	4	4	3	3	2	0
3	4	4	4	5	3	1
3	4	5	4	5	4	1
3	4	4	4	3	3	1
3	3	3	4	4	2	0
1	3	3	3	3	3	0
3	4	4	3	4	2	1
4	4	4	4	4	3	1
3	4	5	4	3	3	1
3	3	4	3	4	3	0

Gambar 14. Pembelahan Data

3.6 Eksplorasi Data

Grafik dibawah ini menujukan distribusi presentase dari variabel Y, yang menampilkan dua kategori 0 dan 1. Pada grafik tersebut, dapat dilihat bahwa kategori 1 mendominasi dengan presentase sebesar 65%, sementara kategori 0 hanya sebesar 35%. Grafik tersebut ditunjukan pada Gambar 15.



Gambar 15. Presentase Variabel Y

3.7 *Split Data*

Tahap ini, dilakukan proses pembagian data menjadi dua bagian, yaitu data latih dan data uji. Dalam konteks analisis klasifikasi kepuasan mahasiswa Ilmu Komputer terhadap pembelajaran daring FMIPA UHO, data latih digunakan untuk mengajarkan algoritma XGBoost dan AdaBoost bagaimana mengenali pola-pola dalam data yang menujukan tingkat kepuasan mahasiswa. Serta data uji membantu mengevaluasi seberapa baik model dapat memprediksi data yang balum pernah dilihat sebelumnya. Pada tahap ini dilakukan pembagian data dengan rasio 60:30 (skenario I), 70:30 (skenario II), 80:20 (skenario III). Data latih dan uji ditunjukan pada Tabel 4.

Tabel 4. Skenario Split Data

Skenario	Data	
	Training	Testing
I	48	32
II	56	24
III	64	16

3.8 Perbandingan Algoritma XGBoost dan AdaBoost

Perbandingan untuk menganalisis kinerja algoritma XGBoost dan AdaBoost dilakukan dengan beberapa skenario dan serangkaian percobaan dengan berbagai konfigurasi parameter *learning rate* dan *n_estimators*. Percobaan ini bertujuan untuk mengevaluasi sejauh mana variasi parameter tersebut memengaruhi kualitas prediksi yang dihasilkan oleh XGBoost dan AdaBoost.

1. Pemodelan dan Evaluasi Kinerja XGBoost

Hasil pemodelan XGBoost terdapat beberapa skenario yang menunjukan variasi dalam nilai parameter. Setiap skenario, model dievaluasi berdasarkan prediksi yang dihasilkan dan dibandingkan dengan data aktual serta *log loss* yang digunakan untuk mengukur kualitas probabilistik prediksi model. Semakin kecil nilai *log loss*, maka

semakin baik performa model dalam memberikan prediksi dengan *probabilitas* yang akurat. Hasi pemodelan XGBoost dapat dilihat pada Tabel 5.

Tabel 5. Hasil Pemodelan XGBoost

Skenario	Learning rate	N estimators	Prediksi	Aktual		Log Loss	Error Tiap Kelas
				0	1		
I	0.1	50	0	14	0	0.1069	0%
			1	0	34		0%
		100	0	13	1	0.0984	7%
			1	0	34		0%
		200	0	13	1	0.0983	7%
			1	0	34		0%
	0.2	50	0	13	1	0.0967	7%
			1	0	34		0%
		100	0	13	1	0.0967	7%
			1	0	34		0%
		200	0	13	1	0.0967	7%
			1	0	34		0%
	0.3	50	0	13	1	0.0962	7%
			1	0	34		0%
		100	0	13	1	0.0967	7%
			1	0	34		0%
		200	0	13	1	0.0962	7%
			1	0	34		0%
II	0.1	50	0	18	0	0.0896	0%
			1	0	38		0%
		100	0	18	0	0.0773	0%
			1	0	38		0%
		200	0	18	0	0.0767	0%
			1	0	38		0%
	0.2	50	0	18	0	0.0756	0%
			1	0	38		0%
		100	0	18	0	0.0750	0%
			1	0	38		0%
		200	0	18	0	0.0750	0%
			1	0	38		0%
	0.3	50	0	18	0	0.0730	0%
			1	0	38		0%
		100	0	18	0	0.0730	0%
			1	0	38		0%
		200	0	18	0	0.0730	0%
			1	0	38		0%
III	0.1	50	0	25	0	0.0773	0%
			1	0	39		0%
		100	0	25	0	0.0664	0%
			1	0	39		0%
		200	0	25	0	0.0653	0%
			1	0	39		0%
	0.2	50	0	25	0	0.0662	0%
			1	0	39		0%
		100	0	25	0	0.0658	0%
			1	0	39		0%
		200	0	25	0	0.0658	0%
			1	0	39		0%
	0.3	50	0	25	0	0.0662	0%
			1	0	39		0%
		100	0	25	0	0.0662	0%
			1	0	39		0%
		200	0	25	0	0.0662	0%
			1	0	39		0%

Tabel 5 menunjukkan hasil pemodelan XGBoost pada tiga skenario berbeda (I, II, dan III) dengan berbagai kombinasi parameter *learning rate* dan *n_estimators*. Berikut adalah *log loss* terendah dari beberapa skenario.

a. Skenario I dengan *learning rate* 0.3, diperoleh nilai *log loss* sebesar 0.0962 pada varian jumlah *n_estimators* 50 dan 200. Nilai *log loss* ini menunjukkan bahwa model masih menghasilkan kesalahan prediksi probabilistik yang relatif lebih tinggi.
b. Skenario II, pada skenario ini, dengan *learning rate* 0.3, nilai *log loss* menurun menjadi 0.0730 secara konsisten pada jumlah *n_estimators* 50, 100, dan 200. Penurunan ini mencerminkan adanya peningkatan kemampuan model dalam mempelajari pola data dibandingkan dengan Skenario I. Stabilitas nilai *log loss* pada berbagai variasi *n_estimators* juga menunjukkan bahwa model lebih adaptif dan mampu menghasilkan prediksi probabilistik yang lebih akurat.
c. Skenario III dengan *learning rate* 0.1, diperoleh nilai *log loss* terendah yaitu 0.0653 pada jumlah *n_estimators* 200. Hasil ini menjadi performa terbaik di antara ketiga skenario, karena model mampu menghasilkan prediksi probabilistik yang paling optimal dengan tingkat kesalahan yang paling kecil. Kondisi ini menegaskan bahwa kombinasi learning rate rendah dengan jumlah *n_estimators* yang cukup besar dapat memberikan keseimbangan antara proses pembelajaran dan generalisasi model.

Setelah melakukan pemodelan tahap selanjutnya yaitu model yang telah dibuat dievaluasi menggunakan *confusion matrix* dan AUC. Hasil evaluasi kinerja model XGBoost dapat dilihat pada Tabel 6.

Tabel 6. Hasil Evaluasi Kinerja Model XGBoost

Skenario	Learning rate	Nestimators	Prediksi	Aktual		Akurasi	Presisi	Recall	F1-score	AUC
				0	1					
I	0.1	50	0	11	3	84%	84%	89%	86%	0.837
			1	2	16					
		100	0	10	4	81%	80%	89%	84%	0.802
			1	2	16					
		200	0	10	4	81%	80%	89%	84%	0.802
			1	2	16					
	0.2	50	0	10	4	81%	80%	89%	84%	0.802
			1	2	16					
		100	0	10	4	81%	80%	89%	84%	0.802
			1	2	16					
		200	0	10	4	81%	80%	89%	84%	0.802
			1	2	16					

Skenario	Learning rate	Estimators	Prediksi	Aktual		Akurasi	Presisi	Recall	F1-score	AUC
				0	1					
	0.3	50	0	10	4	81%	80%	89%	84%	0.802
			1	2	16					
		100	0	10	4	81%	80%	89%	84%	0.802
			1	2	16					
		200	0	10	4	81%	80%	89%	84%	0.802
			1	2	16					
II	0.1	50	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		100	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		200	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
	0.2	50	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		100	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		200	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
	0.3	50	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		100	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		200	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
III	0.1	50	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					
		100	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					
		200	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					
	0.2	50	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					
		100	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					
		200	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					
	0.3	50	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					
		100	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					
		200	0	2	1	88%	92%	92%	92%	0.795
			1	1	12					

Tabel 6 menunjukkan hasil evaluasi XGBoost pada tiga skenario berbeda (I, II, dan III) dengan berbagai kombinasi parameter *learning rate* dan *n_estimators*. Berdasarkan tabel, skenario dengan AUC tertinggi merupakan kondisi yang paling diinginkan karena menunjukkan kinerja model yang lebih baik. Berikut adalah AUC tertinggi dari beberapa skenario.

a. Skenario I dengan *learning rate* 0.1, diperoleh nilai AUC tertinggi sebesar 0.837 pada jumlah *n_estimators* 50. Nilai ini menunjukkan bahwa model mampu melakukan klasifikasi dengan tingkat kemampuan yang cukup baik.
b. Skenario II, pada skenario ini dengan *learning rate* 0.1, 0.2, dan 0.3, nilai AUC yang dihasilkan adalah 0.914 pada seluruh variasi jumlah *n_estimators.* Hasil ini menunjukkan konsistensi performa model yang sangat baik, di mana perubahan jumlah *n_estimators* tidak memengaruhi nilai AUC. Dengan kata lain, model cukup stabil pada kombinasi parameter ini, serta mampu membedakan kelas positif dan negatif dengan tingkat yang tinggi.
c. Skenario III dengan *learning rate* 0.1, 0.2, dan 0.3, nilai AUC yang diperoleh adalah 0.795 pada seluruh variasi jumlah *n_estimators*. Nilai ini lebih rendah dibandingkan Skenario I dan II, yang mengindikasikan adanya penurunan kemampuan model dalam membedakan kelas positif dan negatif.

2. Pemodelan dan Evaluasi Kinerja AdaBoost

Hasil pemodelan AdaBoost terdapat beberapa skenario yang menunjukkan variasi dalam nilai parameter. Setiap skenario, model dievaluasi berdasarkan prediksi yang dihasilkan dan dibandingkan dengan data aktual serta *exponential error* yaitu fungsi kerugian untuk mengukur seberapa besar kesalahan prediksi model. Semakin kecil nilai *exponential error*, maka semakin baik performa model. Hasil pemodelan AdaBoost dapat dilihat pada Tabel 7.

Tabel 7. Hasi Pemodelan AdaBoost

Skenario	Learning rate	Estimators	Prediksi	Aktual		Exponential Error	Error Tiap Kelas
				0	1		
I	0.1	50	0	14	0	0.3146	0%
			1	4	30		12%
		100	0	14	0		0%
			1	4	30		12%
		200	0	14	0		0%
			1	4	30		12%
	0.2	50	0	14	0		0%
			1	4	30		12%
		100	0	14	0		0%
			1	4	30		12%
		200	0	14	0		0%
			1	4	30		12%
	0.3	50	0	14	0		0%
			1	4	30		12%
		100	0	14	0		0%
			1	4	30		12%
		200	0	14	0		0%
			1	4	30		12%

Skenario	Learning rate	N estimators	Prediksi	Aktual		Exponential Error	Error Tiap Kelas
				0	1		
			1	2	32	0.1507	6%
		200	0	14	0	0.0361	0%
			1	0	34		0%
	0.2	50	0	14	0	0.1406	0%
			1	2	32		6%
		100	0	14	0	0.0311	0%
			1	0	34		0%
		200	0	14	0	0.0030	0%
			1	0	34		0%
	0.3	50	0	14	0	0.0447	0%
			1	0	34		0%
		100	0	14	0	0.0064	0%
			1	0	34		0%
		200	0	14	0	0.0004	0%
			1	0	34		0%
II	0.1	50	0	17	1	0.3017	6%
			1	5	33		13%
		100	0	18	0	0.1375	0%
			1	0	38		0%
		200	0	18	0	0.0415	0%
			1	0	38		0%
	0.2	50	0	18	0	0.1159	0%
			1	0	38		0%
		100	0	18	0	0.0343	0%
			1	0	38		0%
		200	0	18	0	0.0058	0%
			1	0	38		0%
	0.3	50	0	18	0	0.0523	0%
			1	0	38		0%
		100	0	18	0	0.0115	0%
			1	0	38		0%
		200	0	18	0	0.0008	0%
			1	0	38		0%
III	0.1	50	0	24	1	0.3052	4%
			1	5	34		13%
		100	0	25	0	0.1450	0%
			1	0	39		0%
		200	0	25	0	0.0483	0%
			1	0	39		0%
	0.2	50	0	25	0	0.1305	0%
			1	0	39		0%
		100	0	25	0	0.0419	0%
			1	0	39		0%
		200	0	25	0	0.0077	0%
			1	0	39		0%
	0.3	50	0	25	0	0.0592	0%
			1	0	39		0%
		100	0	25	0	0.0131	0%
			1	0	39		0%
		200	0	25	0	0.0010	0%
			1	0	39		0%

Tabel 7 menunjukkan hasil pemodelan AdaBoost pada tiga skenario berbeda (I, II, dan III) dengan berbagai kombinasi parameter *learning rate* dan *n_estimators*. Berikut adalah *exponential error* terendah dari beberapa skenario.

a. Skenario I dengan *learning rate* 0.3 dan jumlah *n_estimators* 200, nilai *exponential error* yang diperoleh adalah 0.0004. Nilai ini tergolong sangat kecil, yang menandakan bahwa model mampu mempelajari pola data dengan baik dan menghasilkan kesalahan yang sangat minim.
b. Skenario II, pada skenario ini dengan parameter yang sama (*learning rate* 0.3 dan *n_estimators* 200), nilai *exponential error* meningkat menjadi 0.0008. Peningkatan ini menunjukkan adanya penurunan kemampuan model dalam menangkap pola data, meskipun nilainya masih relatif kecil.
c. Skenario III dengan *learning rate* 0.3 dan *n_estimators* 200, nilai *exponential error* kembali naik menjadi 0.0010. Peningkatan ini menunjukkan bahwa performa model semakin menurun dibandingkan dua skenario sebelumnya.

Setelah pemodelan dilakukan, tahap selanjutnya yaitu mengevaluasi model menggunakan *confusion matrix* dan nilai AUC. Hasil evaluasi kinerja model AdaBoost dapat dilihat pada Tabel 8.

Tabel 8. Hasil Evaluasi Kinerja Model AdaBoost

Skenario	Learning rate	Nestimators	Prediksi	Aktual		Akurasi	Presisi	Recall	F1-score	AUC
				0	1					
I	0.1	50	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
		100	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
		200	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
	0.2	50	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
		100	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
		200	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
	0.3	50	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
		100	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
		200	0	12	2	88%	89%	89%	89%	0.873
			1	2	16					
II	0.1	50	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		100	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
			0	9	1					

Skenario	Learning rate	Estimators	Prediksi	Aktual		Akurasi	Presisi	Recall	F1-score	AUC
				0	1					
		200	1	1	13	92%	93%	93%	93%	0.914
	0.2	50	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		100	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		200	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
	0.3	50	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		100	0	9	1	92%	93%	93%	93%	0.914
			1	1	13					
		200	0	10	0	96%	100%	93%	96%	0.964
			1	1	13					
III	0.1	50	0	12	2	88%	92%	92%	92%	0.795
			1	2	16					
		100	0	12	2	88%	92%	92%	92%	0.795
			1	2	16					
		200	0	12	2	88%	92%	92%	92%	0.795
			1	2	16					
	0.2	50	0	12	2	88%	92%	92%	92%	0.795
			1	2	16					
		100	0	12	2	88%	92%	92%	92%	0.795
			1	2	16					
		200	0	12	2	88%	92%	92%	92%	0.795
			1	2	16					
	0.3	50	0	12	2	88%	92%	92%	92%	0.795
			1	2	16					
		100	0	12	2	88%	92%	92%	92%	0.795
			1	2	16					
		200	0	3	0	94%	100%	92%	96%	0.962
			1	1	12					

Tabel 8 menunjukkan hasil evaluasi AdaBoost pada tiga skenario berbeda (I, II, dan III) dengan berbagai kombinasi parameter *learning rate* dan *n_estimators*. Berdasarkan tabel, skenario dengan AUC tertinggi merupakan kondisi yang paling diinginkan karena menunjukkan kinerja model yang lebih baik. Berikut adalah AUC tertinggi dari beberapa skenario.

a. Skenario I dengan menggunakan variasi *learning rate* 0.1, 0.2, dan 0.3, nilai AUC yang dihasilkan sama, yaitu sebesar 0.873 pada seluruh varian jumlah *n_estimators*. Hal ini menunjukkan bahwa pada skenario ini, perubahan nilai *learning rate* maupun *n_estimators* tidak memberikan pengaruh signifikan terhadap peningkatan performa model.
b. Skenario II, pada skenario ini digunakan *learning rate* 0.3, dan diperoleh nilai AUC tertinggi yaitu 0.964 ketika jumlah *n_estimators* adalah 200. Hasil ini lebih tinggi dibandingkan Skenario I, sehingga dapat dikatakan bahwa kombinasi *learning rate* 0.3 dengan *n_estimators* 200 memberikan pengaruh positif terhadap peningkatan kinerja model.
c. Skenario III dengan menggunakan *learning rate* 0.3 dengan jumlah *n_estimators* 200, diperoleh nilai AUC sebesar 0.962. Nilai ini sedikit lebih rendah dibandingkan Skenario II, tetapi masih menunjukkan performa yang sangat baik. Hal ini mengindikasikan bahwa meskipun terdapat perbedaan kecil, model tetap konsisten menghasilkan kinerja tinggi pada parameter tersebut.

3.9 Hasil Perbandingan XGBoost dan AdaBoost

Penerapan algoritma XGBoost dan AdaBoost pada analisis klasifikasi kepuasan mahasiswa Ilmu Komputer terhadap pembelajaran daring FMIPA UHO dengan tiga skenario dan berbagai konfigurasi parameter memberikan hasil yang berbeda-beda. Hasil evaluasi yang tertinggi pada XGBoost dan AdaBoost dengan tiga skenario berbeda dapat dilihat pada Tabel 9 dan Tabel 10.

Tabel 9. Hasil Evaluasi yang tertinggi pada XGBoost

Skenario	Learning rate	Nestimators	Train Split Data Test (%)	Akurasi	Presisi	Recall	F1-score	AUC
I	0.1	50	60:40	84%	84%	89%	86%	0.837
II	0.1, 0.2, 0.3	50, 100, 200	70:30	92%	93%	93%	93%	0.914
III	0.1, 0.2, 0.3	50, 100, 200	80:20	88%	92%	92%	92%	0.795

Tabel 10. Hasil Evaluasi yang tertinggi pada AdaBoost

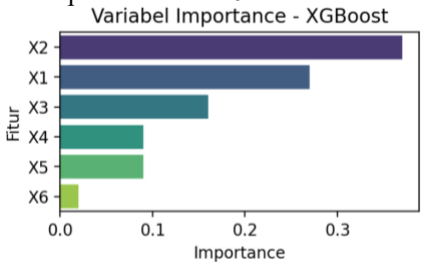
Skenario	Learning rate	Nestimators	Train Split Data Test (%)	Akurasi	Presisi	Recall	F1-score	AUC

I	0.1, 02, 03	50, 100, 200	60:40	88 %	89 %	89 %	89 %	0.873
II	0.3	200	70:30	96 %	100 %	93 %	96 %	0.964
III	0.3	200	80:20	94 %	100 %	92 %	96 %	0.962

Berdasarkan Tabel 9 hasil pengujian algoritma XGBoost dengan tiga skenario berbeda (I, II, dan III) dengan berbagai kombinasi parameter menunjukan hasil evaluasi yang tertinggi pada skenario II (70:30) dengan tingkat akurasi 92%, presisi 93%, recall 93%, f1-score 93% dan AUC 0.914, hasil AUC tersebut model (sangat baik) dalam membedakan kelas positif adan negatif. Berdasarkan Tabel 10 hasil pengujian algoritma AdaBoost dengan tiga skenario berbeda (I, II, dan III) dengan berbagai kombinasi parameter menunjukan hasil evaluasi yang tertinggi pada skenario II (70:30) dengan tingkat akurasi 96%, presisi 100%, recall 93%, f1-score 96% dan AUC 0.964, hasil AUC tersebut model (sangat baik) dalam membedakan kelas positif adan negatif.

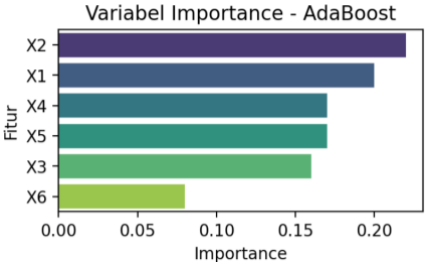
3.10 Variabel *Importance*

Hasil evaluasi algoritma XGBoost dan AdaBoost menunjukkan bahwa kedua model memperoleh nilai evaluasi tertinggi yang berbeda. Selain itu, analisis variable *importance* digunakan untuk menggambarkan seberapa besar kontribusi masing-masing variabel (fitur) terhadap kinerja model dalam melakukan prediksi. Semakin besar angka variabel importance menunjukan semakin besar juga peran variabel tersebut dalam mempengaruhi hasil prediksi. Variabel i*mportance* pada hasil evaluasi tertinggi algoritma XGBoost dapat dilihat pada Gambar 16.



Gambar 16. Hasil Variabel Importance pada XGBoost

Variabel *importance* pada hasil evaluasi tertinggi algoritma AdaBoost dapat dilihat pada Gambar 17.



Gambar 17. Hasil Variabel Importance pada AdaBoost

3.11 Interprestasi Hasil

Hasil evaluasi dari tiga skenario (I, II, dan III) kedua algoritma menunjukkan hasil yang berbeda-beda. Pengujian algoritma XGBoost menunjukan hasil akurasi, presisi, recall, dan f1-score yang paling tinggi pada skenario II (70:30) dengan parameter *learning rate* 0.1, 0.2, dan 0.3 pada *n_estimator* 50, 100, 200 dengan tingkat akurasi 92%, presisi 93%, recall 93%, f1-score 93% dan AUC 0.914. Sedangkan pengujian algoritma Adaboost menunjukan hasil akurasi, presisi, recall, dan f1-score yang paling tinggi pada skenario II (70:30) dengan parameter *larning rate* 0.3 dan *n_estimator* 200 dengan tingkat akurasi 96%, presisi 100%, recall 93%, f1-score 96% dan AUC 0.964. Berdasarkan seluruh skenario percobaan yang telah dilakukan, AUC tertinggi diperoleh pada skenario II (70:30) dengan menggunakan algoritma AdaBoost. Hasil ini menunjukkan bahwa AdaBoost memiliki kinerja yang lebih baik dibandingkan dengan algoritma XGBoost.

Hasil analisis variabel *importance* pada kedua model, yaitu XGBoost dan AdaBoost menunjukkan bahwa variabel kemudahan interaksi (X2) memiliki peranan yang sangat dominan dalam membentuk prediksi kepuasan mahasiswa terhadap pembelajaran daring. Hal ini ditunjukkan oleh nilai variabel *importance* yang paling besar pada kedua algoritma, sehingga variabel tersebut memberikan kontribusi paling signifikan dalam menentukan hasil klasifikasi. Dengan kata lain, baik XGBoost maupun AdaBoost sama-sama menilai bahwa aspek kemudahan interaksi menjadi faktor kunci yang memengaruhi kepuasan mahasiswa. Temuan ini juga menunjukkan konsistensi antar algoritma dalam mengidentifikasi faktor yang paling berpengaruh, sehingga dapat menjadi dasar pertimbangan penting dalam upaya peningkatan kualitas pembelajaran daring, khususnya pada peningkatan interaksi yang mudah dan efektif antara mahasiswa dan dosen.

4. KESIMPULAN

Berdasarkan penelitian yang telah dilakukan, dapat disimpulkan bahwa penerapan algoritma XGBoost dan AdaBoost mampu bekerja dengan baik dalam mengklasifikasikan tingkat kepuasan mahasiswa Ilmu Komputer terhadap pembelajaran daring. Hasil evaluasi dari tiga skenario (I, II, dan III) kedua algoritma ini menunjukkan hasil yang berbeda pada masing-masing metrik evaluasi, yaitu akurasi, presisi, recall, f1-score dan AUC. Algoritma AdaBoost memperoleh AUC tertinggi sebesar 0.964 pada skenario II (70:30) dengan parameter *larning rate* 0.3 dan *n_estimator* 200, sedangkan algoritma XGBoost mencapai AUC sebesar 0.914 pada skenario II (70:30) dengan parameter *learning rate* 0.1, 0.2, dan 0.3 pada seluruh variasi jumlah *n_estimator*. Hasil ini menunjukkan bahwa algoritma AdaBoost lebih unggul dibandingkan XGBoost.

Hasil analisis variabel importance pada kedua model menunjukkan bahwa kemudahan interaksi (X2) memiliki peranan yang sangat dominan dalam membentuk prediksi kepuasan mahasiswa Ilmu Komputer terhadap pembelajaran daring.

5.DAFTAR PUSTAKA

[1] Fatmawati, F., & Narti, N. (2022). Perbandingan algoritma C4.5 dan Naive Bayes dalam klasifikasi tingkat kepuasan mahasiswa terhadap pembelajaran daring. *JTIM: Jurnal Teknologi Informasi dan Multimedia*, 4(1), 1–12.

[2] Mahendro, I., & Abimanto, D. (2022). Analisa Kepuasan Mahasiswa Terhadap E-Learning Menggunakan Algoritma Support Vector Machine. *Jurnal Sains Dan Teknologi Maritim*, 23(1), 97–108.

[3] Mustari, K. A., Assiroj, P., Hartati, B., & Samuel, F. (2024). IMPLEMENTASI DATA MINING PADA INSTANSI PEMERINTAHAN. *JATI* (*Jurnal Mahasiswa Teknik Informatika*), 8(3), 3137–3142.

[4] Pratama, A., Nurcahyo, A. C., & Firgia, L. (2023). Penerapan Machine Learning dengan Algoritma Logistik Regresi untuk Memprediksi Diabetes. *Prosiding CORISINDO 2023*.

[5] Purnama, N., & Utami, N. W. (2023). Penerapan Algoritma AdaBoost untuk Optimasi Prediksi Kunjungan Wisatawan ke Bali dengan Metode Decision Tree. *JUSIM* (*Jurnal Sistem Informasi Musirawas*), 8(2), 119–126.

[6] Rahayu, P. W., et al. (2024). *Buku ajar data mining*. PT Sonpedia Publishing Indonesia.

[7] Kurniawan, H. P., & Farhatuaini, L. (2024). Identifikasi Pola Kepuasan Mahasiswa Terhadap Proses Pembelajaran Menggunakan Algoritma K-Means Clustering. *Jurnal Informatika: Jurnal Pengembangan IT*, 9(2), 164–172.

[8] Nurtiani, D. N., & Saputra, R. A. (2024). Klasifikasi Tingkat Kepuasan Mahasiswa Terhadap Fasilitas Kampus Menggunakan Metode ID3. *METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi*, 8(1), 123–128.

[9] Riswandhana, W. A. T., & Muhammad, A. H. (2024). Optimalisasi Akurasi Algoritma C4. 5 dengan Metode Adaptive Boosting Memprediksi Siswa dalam Menerima Dana Pendidikan. *G-Tech: Jurnal Teknologi Terapan*, 8(4), 2895-2902.

[10] Saputro, M. R., Mahdiyah, U., & Swanjaya, D. (2024). Perbandingan Metode Adaptive Boosting dan Extreme Gradient Boosting Untuk Prediksi Hasil Pertandingan Liga Spanyol. *Nusantara of Engineering (NOE),* 7(1), 67-73.