Implementasi Algoritma Random Forest untuk Klasifikasi Tingkat Produktivitas Mahasiswa Berdasarkan Aktivitas Akademik dan Non Akademik

Alif Fahmi Syaifuddin[1*], Muhammad Rezan Arifana Putra[2], Muhammad Hadia Awwala Faradisa[3], Dias Faturrahman Wicaksono[4]

[1*]Teknik Informatika/Fakultas Ilmu Komputer
Universitas Duta Bangsa Surakarta
[1]230103088@mhs.udb.ac.id

[2]Teknik Informatika/Fakultas Ilmu Komputer
Universitas Duta Bangsa Surakarta
[2]230103111@mhs.udb.ac.id

[3]Teknik Informatika/Fakultas Ilmu Komputer
Universitas Duta Bangsa Surakarta
[3]230103110@mhs.udb.ac.id

[4]Teknik Informatika/Fakultas Ilmu Komputer
Universitas Duta Bangsa Surakarta
[4]230103170@mhs.udb.ac.id

***Abstrak*— Produktivitas mahasiswa merupakan salah satu indikator penting dalam mengevaluasi keberhasilan proses pembelajaran di perguruan tinggi. Tingkat produktivitas dipengaruhi oleh berbagai aktivitas akademik dan nonakademik sehingga diperlukan suatu metode yang mampu mengklasifikasikan tingkat produktivitas mahasiswa secara cepat dan akurat. Penelitian ini bertujuan untuk mengimplementasikan algoritma Random Forest dalam mengklasifikasikan tingkat produktivitas mahasiswa berdasarkan aktivitas akademik dan nonakademik serta mengembangkan sistem berbasis web sebagai implementasi model klasifikasi. Penelitian menggunakan Student Performance Dataset yang terdiri dari 2.392 data mahasiswa dengan atribut GPA, StudyTimeWeekly, Absences, Tutoring, ParentalSupport, Extracurricular, Sports, Music, dan Volunteering, sedangkan GradeClass digunakan sebagai variabel target. Metode penelitian mengacu pada tahapan Cross Industry Standard Process for Data Mining (CRISP-DM) yang meliputi Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, dan Deployment. Proses pemodelan dilakukan menggunakan algoritma Random Forest, sedangkan evaluasi model menggunakan metrik Accuracy, Precision, Recall, dan F1-Score. Hasil pengujian menunjukkan bahwa model Random Forest memperoleh tingkat akurasi sebesar 91,44%, sehingga mampu mengklasifikasikan tingkat produktivitas mahasiswa dengan baik. Model yang telah dibangun kemudian diimplementasikan ke dalam sistem berbasis web menggunakan Flask, sehingga pengguna dapat melakukan prediksi tingkat produktivitas mahasiswa berdasarkan data akademik dan nonakademik yang dimasukkan. Hasil penelitian menunjukkan bahwa algoritma Random Forest efektif digunakan dalam klasifikasi tingkat produktivitas mahasiswa dan dapat menjadi pendukung dalam proses evaluasi akademik di lingkungan perguruan tinggi.**
Kata kunci— Random Forest, Klasifikasi, Produktivitas Mahasiswa, CRISP-DM, Flask.

***Abstract*— Student productivity is one of the key indicators used to evaluate the effectiveness of the learning process in higher education. The level of student productivity is influenced by various academic and non-academic activities, making it necessary to develop a method capable of classifying student productivity accurately and efficiently. This study aims to implement the Random Forest algorithm to classify student productivity based on academic and non-academic activities and to develop a web-based system as the implementation of the classification model. The study uses the Student Performance Dataset, consisting of 2,392 student records, with GPA, StudyTimeWeekly, Absences, Tutoring, ParentalSupport, Extracurricular, Sports, Music, and Volunteering as input attributes, while GradeClass serves as the target variable. The research follows the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology, which consists of Business Understanding, Data Understanding, Data Preparation, Modeling, Evaluation, and Deployment stages. The classification model is developed using the Random Forest algorithm and evaluated using Accuracy, Precision, Recall, and F1-Score metrics. The experimental results show that the proposed model achieved an accuracy of 91.44%, indicating that Random Forest is effective in classifying student productivity. Furthermore, the trained model was deployed into a Flask-based web application, enabling users to predict student productivity based on academic and non-academic data. The results demonstrate that the Random Forest algorithm can effectively support the classification of student productivity and assist higher education institutions in academic evaluation.**
Keywords— Random Forest, Classification, Student Productivity, CRISP-DM, Flask**.**

I. PENDAHULUAN

Prestasi akademik mahasiswa menjadi salah satu ukuran penting untuk mengevaluasi keberhasilan proses pembelajaran di perguruan tinggi [1]. Berbagai faktor, baik yang terkait dengan kegiatan akademik maupun nonakademik, mempengaruhi tingkat prestasi akademik [2]. Misalnya, ukuran seperti *GPA*, kehadiran, durasi belajar, dan dukungan dari orang tua dapat mempengaruhi hasil belajar mahasiswa [3].

Kemajuan teknologi informasi mendorong pemanfaatan teknik *data mining* di sektor pendidikan [4]. Salah satu metode yang sering diadopsi adalah klasifikasi, yang merupakan proses pengelompokan data ke dalam kategori tertentu berdasarkan karakteristik yang ada [5]. Dengan klasifikasi, lembaga pendidikan dapat memperoleh informasi yang berharga untuk memahami kondisi akademis mahasiswa dengan lebih baik.

Random Forest adalah salah satu algoritma klasifikasi yang banyak diminati karena kemampuannya dalam menghasilkan tingkat akurasi yang tinggi [6]. Algoritma ini beroperasi dengan membuat beberapa pohon keputusan dan menggabungkan hasil prediksi dari setiap pohon untuk mencapai keputusan akhir yang lebih tepat [7]. Selain itu, *Random Forest* juga dapat meminimalkan risiko *overfitting* pada data yang kompleks [8].

Berbagai penelitian sebelumnya telah menerapkan algoritma *Random Forest* pada bidang pendidikan untuk melakukan prediksi prestasi akademik maupun klasifikasi performa belajar mahasiswa. Hasil penelitian tersebut menunjukkan bahwa *Random Forest* memiliki tingkat akurasi yang baik dalam proses klasifikasi. Namun, sebagian besar penelitian masih berfokus pada prediksi nilai akademik dan belum banyak yang mengintegrasikan faktor aktivitas akademik maupun nonakademik dalam proses klasifikasi produktivitas mahasiswa. Selain itu, implementasi model klasifikasi ke dalam sistem berbasis *web* yang dapat dimanfaatkan secara langsung sebagai media evaluasi akademik juga masih terbatas. Oleh karena itu, penelitian ini mengimplementasikan algoritma *Random Forest* untuk mengklasifikasikan tingkat produktivitas mahasiswa berdasarkan aktivitas akademik dan nonakademik serta mengintegrasikannya ke dalam sistem berbasis *web*.

Penelitian ini menggunakan *Student Performance Dataset* yang terdiri dari 2.392 data mahasiswa. Variabel target pada dataset adalah *GradeClass*, yaitu kategori hasil capaian akademik mahasiswa. Dalam penelitian ini, *GradeClass* digunakan sebagai indikator tingkat produktivitas mahasiswa karena capaian akademik merupakan hasil yang dipengaruhi oleh berbagai aktivitas akademik maupun nonakademik, seperti waktu belajar, tingkat kehadiran, bimbingan belajar, dukungan orang tua, serta keterlibatan dalam kegiatan ekstrakurikuler. Dengan demikian, klasifikasi yang dihasilkan pada penelitian ini tidak dimaksudkan untuk mengukur produktivitas secara langsung, tetapi menggunakan *GradeClass* sebagai representasi produktivitas belajar mahasiswa berdasarkan atribut-atribut yang tersedia pada dataset.

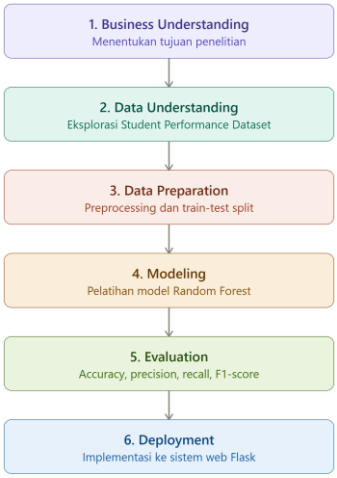
Data tersebut dianalisis dengan memanfaatkan algoritma *Random Forest* untuk menyusun model klasifikasi yang terbaik.

Penelitian ini bertujuan untuk menerapkan algoritma *Random Forest* dalam mengklasifikasikan tingkat produktivitas mahasiswa berdasarkan aktivitas akademik dan nonakademik serta mengevaluasi kinerja model menggunakan metrik *Accuracy*, *Precision*, *Recall*, dan *F1-Score*. Selain itu, model yang dihasilkan diimplementasikan ke dalam sistem berbasis *web* menggunakan *Flask* sebagai media pendukung evaluasi produktivitas mahasiswa.

II. METODOLOGI PENELITIAN

Penelitian ini menggunakan metode *Cross Industry Standard Process for Data Mining* (CRISP-DM) sebagai kerangka kerja dalam proses pembangunan model klasifikasi. Metode CRISP-DM dipilih karena memiliki tahapan yang sistematis dan terstruktur, mulai dari memahami permasalahan hingga mengimplementasikan model ke dalam system [9]. Tahapan yang digunakan pada penelitian ini

meliputi *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modeling*, *Evaluation*, dan *Deployment*.



Gambar 1. Alur CRISP-DM

A. *Business Understanding*

Tahap *Business Understanding* merupakan tahap awal yang bertujuan untuk memahami permasalahan penelitian serta menentukan tujuan yang ingin dicapai [10]. Permasalahan yang diangkat pada penelitian ini adalah bagaimana mengklasifikasikan tingkat produktivitas mahasiswa berdasarkan aktivitas akademik dan nonakademik. Produktivitas mahasiswa dipengaruhi oleh berbagai faktor, seperti indeks prestasi, waktu belajar, tingkat kehadiran, dukungan orang tua, serta keterlibatan dalam kegiatan ekstrakurikuler. Oleh karena itu, diperlukan suatu model klasifikasi yang mampu mengolah data tersebut sehingga dapat memberikan hasil prediksi tingkat produktivitas mahasiswa secara cepat dan akurat. Tujuan penelitian ini adalah mengimplementasikan algoritma *Random Forest* untuk mengklasifikasikan tingkat produktivitas mahasiswa serta mengembangkan sistem berbasis web sebagai implementasi model yang telah dibangun

B. *Data Understanding*

Pada tahap *Data Understanding*, dilakukan proses pengenalan dan eksplorasi terhadap dataset yang digunakan [11]. Dataset penelitian berasal dari *Student Performance Dataset* yang diperoleh melalui platform *Kaggle* dan terdiri dari 2.392 data mahasiswa. Dataset tersebut memuat berbagai atribut yang berkaitan dengan aktivitas akademik dan nonakademik mahasiswa.

Atribut yang digunakan dalam penelitian meliputi *GPA*, *StudyTimeWeekly*, *Absences*, *Tutoring*, *ParentalSupport*, *Extracurricular*, *Sports*, *Music*, *dan Volunteering*, *sedangkan GradeClass* digunakan sebagai variabel target dalam proses klasifikasi. Pada tahap ini juga dilakukan analisis awal terhadap struktur data, tipe data setiap atribut, distribusi data, serta pemeriksaan kondisi dataset sebelum memasuki tahap pengolahan data.

C. *Data Preparation*

Tahap *Data Preparation* bertujuan untuk mempersiapkan dataset agar siap digunakan pada proses pemodelan menggunakan algoritma *Random Forest*. Pada tahap ini dilakukan beberapa proses, yaitu pemilihan atribut, pembersihan data, pembagian dataset, serta analisis distribusi kelas.

Proses pertama adalah menghapus atribut *StudentID* karena atribut tersebut hanya berfungsi sebagai identitas unik setiap mahasiswa dan tidak memiliki pengaruh terhadap proses klasifikasi. Selanjutnya dipilih sembilan atribut yang digunakan sebagai variabel masukan, yaitu *GPA*, *StudyTimeWeekly*, *Absences*, *Tutoring*, *ParentalSupport*, *Extracurricular*, *Sports*, *Music*, *dan Volunteering*, sedangkan *GradeClass* digunakan sebagai variabel target yang dalam penelitian ini dijadikan indikator tingkat produktivitas mahasiswa.

Sebelum proses pelatihan model dilakukan, dataset diperiksa untuk memastikan tidak terdapat missing value. Berdasarkan hasil pemeriksaan, seluruh atribut memiliki data yang lengkap sehingga tidak diperlukan proses penanganan data hilang maupun transformasi data tambahan.

Selanjutnya dilakukan pembagian dataset menggunakan metode Train-Test Split dengan rasio 80:20, yaitu sebanyak 80% data digunakan sebagai data pelatihan (training data) dan 20%

sisanya digunakan sebagai data pengujian (*testing data*). Pembagian data dilakukan menggunakan nilai random_state = 42 agar hasil pembagian data bersifat konsisten dan dapat direplikasi pada penelitian selanjutnya.

Hasil analisis menunjukkan bahwa distribusi data pada setiap kelas *GradeClass* tidak sepenuhnya seimbang. Namun demikian, penelitian ini tidak menerapkan teknik penanganan *imbalanced data* seperti *SMOTE*, *oversampling*, maupun *undersampling*. Hal tersebut karena algoritma *Random Forest* memiliki kemampuan yang relatif baik dalam menangani distribusi kelas yang tidak seimbang melalui mekanisme pembentukan banyak *decision tree* dan proses *majority voting*. Untuk memperoleh evaluasi yang lebih representatif, penelitian ini menggunakan metode *10-Fold Stratified Cross Validation*, sehingga proporsi setiap kelas tetap dipertahankan pada setiap *fold*.

D. *Modeling*

Tahap *Modeling* merupakan proses pembangunan model klasifikasi menggunakan algoritma *Random Forest* [12]. Algoritma ini bekerja dengan membangun sejumlah pohon keputusan menggunakan teknik *bootstrap sampling* [6]. Setiap pohon menghasilkan prediksi terhadap data masukan, kemudian seluruh hasil prediksi digabungkan menggunakan metode *majority voting* untuk memperoleh hasil klasifikasi akhir [13].

Proses pelatihan model dilakukan menggunakan bahasa pemrograman *Python* pada platform *Google Colab* dengan memanfaatkan pustaka *Scikit-Learn*. Setelah proses pelatihan selesai, model disimpan dalam format .pkl menggunakan *library Joblib* sehingga dapat digunakan kembali pada tahap implementasi sistem.

Pada penelitian ini algoritma Random Forest diimplementasikan menggunakan pustaka Scikit-Learn. Parameter yang digunakan meliputi n_estimators = 100, yang menunjukkan jumlah decision tree yang dibangun dalam model, criterion = gini sebagai fungsi untuk menentukan kualitas pemisahan (split) pada setiap simpul pohon, max_depth = None sehingga kedalaman pohon tidak dibatasi dan akan berkembang hingga memenuhi kondisi penghentian bawaan, serta random_state = 42 untuk memastikan proses pelatihan dan pembagian data dapat direproduksi dengan hasil yang konsisten.

Setelah parameter ditentukan, model *Random Forest* dilatih menggunakan data pelatihan (*training data*) yang diperoleh dari proses *Train-Test Split*. Model kemudian membentuk sejumlah *decision tree* berdasarkan sampel hasil *bootstrap sampling*. Selanjutnya, hasil prediksi dari setiap *decision tree* digabungkan menggunakan mekanisme *majority voting* untuk menghasilkan keputusan klasifikasi akhir.

Penelitian ini tidak melakukan proses *hyperparameter tuning* seperti *Grid Search* maupun *Random Search*. Model dibangun menggunakan konfigurasi parameter yang telah ditentukan sebelumnya karena telah menghasilkan performa klasifikasi yang baik pada dataset yang digunakan.

E. *Evaluation*

Tahap *Evaluation* dilakukan untuk mengukur performa model *Random Forest* dalam mengklasifikasikan tingkat produktivitas mahasiswa. Evaluasi dilakukan menggunakan *Confusion Matrix* dengan beberapa metrik pengujian, yaitu *Accuracy, Precision, Recall, dan F1-Score.*

Accuracy digunakan untuk mengetahui tingkat ketepatan model dalam melakukan klasifikasi terhadap seluruh data pengujian [14]. *Precision* digunakan untuk mengukur ketepatan prediksi pada setiap kelas, *Recall* digunakan untuk mengetahui kemampuan model dalam mengenali data pada kelas yang sebenarnya, sedangkan *F1-Score* digunakan untuk memberikan nilai keseimbangan antara *Precision* dan *Recall* [15].

Selain menggunakan metode *Train-Test Split*, penelitian ini juga menerapkan *10-Fold Stratified Cross Validation* sebagai metode validasi model. Metode ini bertujuan untuk mengevaluasi kestabilan performa model dengan mempertahankan proporsi setiap kelas pada setiap *fold*, sehingga hasil evaluasi menjadi lebih representatif terhadap keseluruhan dataset.

F. *Deployment*

Tahap *Deployment* merupakan tahap akhir dalam metode CRISP-DM, yaitu mengimplementasikan model yang telah dibangun ke dalam sebuah sistem berbasis web menggunakan *Flask*. Model *Random Forest* yang telah disimpan dalam format .pkl diintegrasikan ke dalam aplikasi sehingga pengguna dapat melakukan prediksi secara langsung melalui antarmuka web.

Pengguna memasukkan data berupa *GPA, StudyTimeWeekly, Absences, Tutoring, ParentalSupport, Extracurricular, Sports, Music, dan Volunteering* melalui formulir yang tersedia. Selanjutnya sistem memproses data menggunakan model *Random Forest* dan menampilkan hasil klasifikasi tingkat produktivitas mahasiswa ke dalam lima kategori, yaitu Sangat Rendah, Rendah, Sedang, Tinggi, dan Sangat Tinggi. Implementasi ini diharapkan dapat mempermudah proses klasifikasi serta menjadi media pendukung dalam evaluasi produktivitas mahasiswa.

Sistem ini dirancang sebagai media pendukung evaluasi produktivitas mahasiswa yang dapat dimanfaatkan oleh dosen maupun pihak program studi dalam melakukan pemantauan terhadap capaian akademik mahasiswa berdasarkan aktivitas akademik dan nonakademik.

III. HASIL DAN PEMBAHASAN

A. *Hasil Business Understanding*

Pada tahap *Business Understanding*, penelitian difokuskan pada permasalahan klasifikasi tingkat produktivitas mahasiswa berdasarkan aktivitas akademik dan nonakademik. Produktivitas mahasiswa dipengaruhi oleh beberapa faktor, seperti nilai *GPA*, waktu belajar mingguan, jumlah ketidakhadiran, bimbingan belajar, dukungan orang tua, serta keterlibatan dalam kegiatan ekstrakurikuler. Oleh karena itu, dibutuhkan suatu model klasifikasi yang mampu mengelompokkan tingkat produktivitas mahasiswa secara cepat dan akurat. Untuk menyelesaikan permasalahan tersebut, penelitian ini menggunakan algoritma *Random Forest* yang kemudian diimplementasikan ke dalam sistem berbasis web.

B. *Hasil Data Understanding*

Tahap *Data Understanding* dilakukan untuk memahami karakteristik dataset yang digunakan. Dataset berasal dari *Student Performance* Dataset yang diperoleh dari *Kaggle* dan terdiri dari 2.392 data mahasiswa. Dataset memiliki beberapa atribut yang berkaitan dengan aktivitas akademik dan nonakademik mahasiswa.

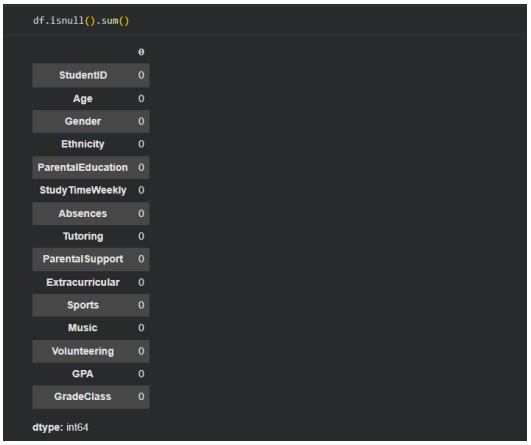
Tabel 1. Atribut Penelitian

No	Nama Atribut	Keterangan
1	*GPA*	Indeks Prestasi mahasiswa
2	*StudyTimeWeekly*	Durasi belajar per minggu
3	*Absences*	Jumlah ketidakhadiran
4	*Tutoring*	Mengikuti bimbingan belajar
5	*ParentalSupport*	Tingkat dukungan orang tua
6	*Extracurricular*	Mengikuti kegiatan ekstrakurikuler
7	*Sports*	Mengikuti kegiatan olahraga
8	*Music*	Mengikuti kegiatan musik
9	*Volunteering*	Mengikuti kegiatan sosial
10	*GradeClass*	Target klasifikasi

C. *Hasil Data Preparation*

Pada tahap Data Preparation, dilakukan proses persiapan data agar dataset siap digunakan pada proses pemodelan menggunakan algoritma Random Forest. Proses ini meliputi pemeriksaan missing value, pemilihan atribut, penghapusan atribut yang tidak digunakan, serta pembagian dataset menjadi data pelatihan dan data pengujian.

Berdasarkan hasil pemeriksaan missing value pada seluruh atribut dataset, tidak ditemukan data yang hilang (*missing value)* sebagaimana ditunjukkan pada Gambar 2 Seluruh atribut memiliki nilai 0, sehingga dataset dinyatakan lengkap dan dapat langsung digunakan pada tahap pemodelan tanpa memerlukan proses penanganan data hilang (*data cleaning*).



Gambar 2. Hasil Pemeriksaan *Missing Value*

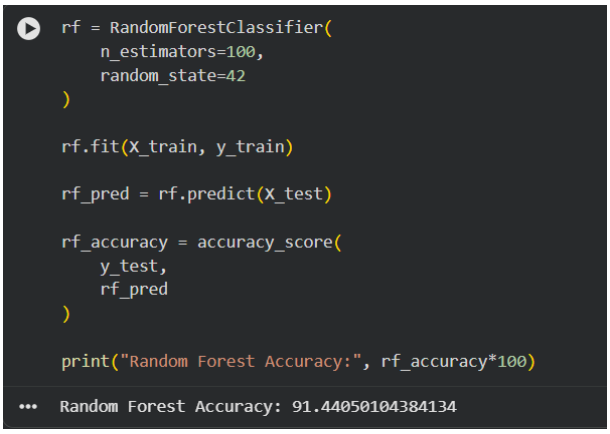
Selanjutnya dilakukan penghapusan atribut *StudentID* karena atribut tersebut hanya berfungsi sebagai identitas unik setiap mahasiswa dan tidak memberikan kontribusi terhadap proses klasifikasi. Setelah itu dipilih sembilan atribut, yaitu *GPA*, *StudyTimeWeekly*, *Absences*, *Tutoring*, *ParentalSupport*, *Extracurricular*, *Sports*, *Music*, dan *Volunteering* sebagai variabel masukan (*feature*), sedangkan *GradeClass* digunakan sebagai variabel target (*target*).

Dataset kemudian dibagi menggunakan metode *Train-Test Split* dengan rasio 80:20, sehingga diperoleh 1.913 data sebagai data pelatihan (*training data*) dan 479 data sebagai data pengujian (*testing data*) dengan nilai random_state = 42. Pembagian data tersebut bertujuan agar model dapat dilatih menggunakan sebagian besar data, sedangkan data pengujian digunakan untuk mengevaluasi kemampuan model dalam mengklasifikasikan data yang belum pernah dipelajari sebelumnya.

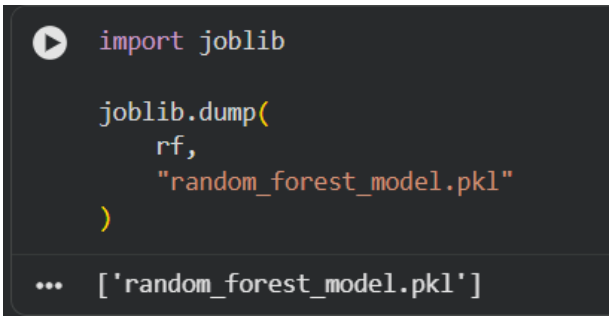
D. *Hasil Modeling*

Tahap *Modeling* dilakukan dengan membangun model klasifikasi menggunakan algoritma *Random Forest*. Model dilatih menggunakan data pelatihan yang telah dipersiapkan sebelumnya. Setelah proses pelatihan selesai, model digunakan untuk melakukan prediksi terhadap data pengujian.

Model kemudian disimpan dalam format .pkl menggunakan *library Joblib* sehingga dapat digunakan kembali pada tahap implementasi sistem berbasis web.



Gambar 3. Pelatihan Model *Random Forest*



Gambar 4. Penyimpanan Model *Random Forest*

E. *Hasil Evaluation*

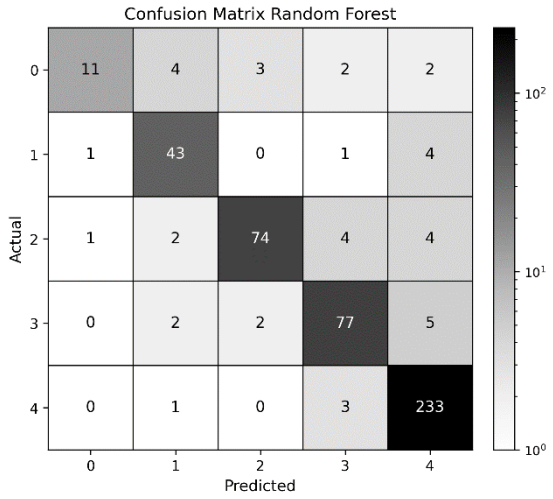
Tahap *Evaluation* bertujuan untuk mengetahui performa model *Random Forest*. Evaluasi dilakukan menggunakan *Confusion Matrix* dengan metrik *Accuracy*, *Precision*, *Recall*, dan *F1-Score*.

Tabel 2. Hasil Evaluasi Model

Metrik	Nilai
Accuracy	91,44%
Precision	91%
Recall	91%
F1-Score	91%

Hasil pengujian menunjukkan bahwa model *Random Forest* memperoleh tingkat akurasi sebesar 91,44%, sehingga mampu mengklasifikasikan tingkat produktivitas mahasiswa dengan baik.

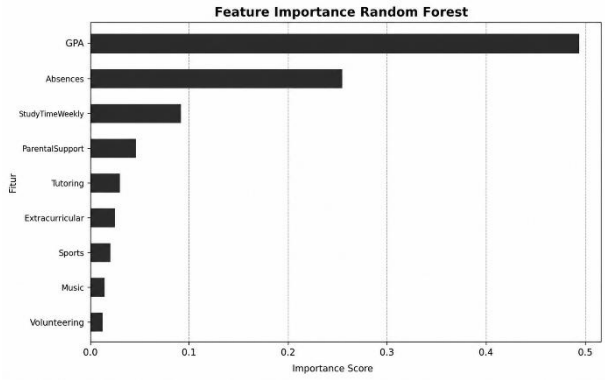
Confusion Matrix



Gambar 5. Confusion Matrix

Berdasarkan *Confusion Matrix*, sebagian besar data berhasil diklasifikasikan ke dalam kelas yang sesuai. Kesalahan klasifikasi masih ditemukan pada beberapa kelas yang memiliki jumlah data relatif lebih sedikit, khususnya kelas 0, namun secara keseluruhan model mampu memberikan hasil prediksi yang baik dengan tingkat akurasi sebesar 91%.

Feature Importance



Gambar 6. *Feature Importance*

Berdasarkan hasil *Feature Importance*, atribut *GPA* memiliki nilai kepentingan paling tinggi dibandingkan atribut lainnya. Hal ini menunjukkan bahwa indeks prestasi kumulatif merupakan indikator utama dalam menentukan tingkat produktivitas mahasiswa. Mahasiswa dengan *GPA* yang tinggi umumnya memiliki konsistensi belajar, kemampuan memahami materi, serta hasil akademik yang lebih baik sehingga lebih sering diklasifikasikan ke dalam kategori produktivitas tinggi. Dominannya atribut GPA menunjukkan bahwa capaian akademik memiliki hubungan yang paling kuat terhadap variabel target *GradeClass*. Hal ini wajar karena *GradeClass* pada *Student Performance Dataset* dibentuk berdasarkan performa akademik mahasiswa, sehingga perubahan nilai GPA memberikan pengaruh yang lebih besar dibandingkan atribut lainnya dalam proses klasifikasi.

Atribut *Absences* menempati urutan kedua. Semakin tinggi tingkat ketidakhadiran mahasiswa, semakin besar kemungkinan mahasiswa memiliki produktivitas yang lebih rendah karena berkurangnya partisipasi dalam proses pembelajaran. Tingginya nilai *Feature Importance* pada atribut *Absences* menunjukkan bahwa kehadiran mahasiswa selama proses pembelajaran berkontribusi terhadap hasil klasifikasi. Mahasiswa dengan tingkat ketidakhadiran yang tinggi cenderung memiliki kesempatan lebih sedikit untuk mengikuti pembelajaran sehingga berpotensi memperoleh *GradeClass* yang lebih rendah.

Atribut *StudyTimeWeekly* juga memberikan kontribusi yang cukup besar terhadap proses klasifikasi. Hal ini menunjukkan bahwa intensitas belajar yang lebih tinggi cenderung berpengaruh terhadap peningkatan hasil akademik dan produktivitas mahasiswa. Waktu belajar mingguan (*StudyTimeWeekly*) juga memberikan kontribusi yang cukup besar karena mencerminkan intensitas belajar mahasiswa di luar kegiatan perkuliahan. Semakin tinggi waktu belajar, semakin besar peluang mahasiswa memahami materi perkuliahan sehingga dapat meningkatkan capaian akademiknya.

Atribut nonakademik seperti *Sports*, *Music*, *Volunteering*, dan *Extracurricular* memiliki nilai *Feature Importance* yang lebih rendah dibandingkan atribut akademik. Hal tersebut menunjukkan bahwa aktivitas nonakademik tidak secara langsung menentukan capaian akademik mahasiswa, namun tetap memberikan kontribusi sebagai faktor pendukung. Keterlibatan dalam kegiatan nonakademik dapat membantu meningkatkan kemampuan manajemen waktu, kerja sama, komunikasi, dan

kedisiplinan mahasiswa yang secara tidak langsung dapat memengaruhi produktivitas belajar.

Tabel 3. Interpretasi Kategori Produktivitas

GradeClass	Tingkat Produktivitas	Karakteristik
4	Sangat Tinggi	GPA tinggi, waktu belajar tinggi, ketidakhadiran rendah
3	Tinggi	GPA baik, kehadiran baik, cukup aktif belajar
2	Sedang	GPA sedang, waktu belajar cukup
1	Rendah	GPA rendah, ketidakhadiran mulai meningkat
0	Sangat Rendah	GPA sangat rendah, waktu belajar rendah, ketidakhadiran tinggi

Berdasarkan hasil klasifikasi, setiap kelas produktivitas memiliki karakteristik yang berbeda. Mahasiswa dengan kategori produktivitas tinggi umumnya memiliki nilai *GPA* yang tinggi, waktu belajar yang lebih lama, serta tingkat ketidakhadiran yang rendah. Sebaliknya, kategori produktivitas rendah didominasi oleh mahasiswa dengan *GPA* rendah, waktu belajar yang relatif sedikit, dan tingkat ketidakhadiran yang lebih tinggi. Karakteristik tersebut menunjukkan bahwa atribut akademik memiliki pengaruh yang lebih besar dibandingkan atribut nonakademik dalam menentukan hasil klasifikasi.

Hasil *10-Fold Stratified Cross Validation*

Selain menggunakan metode *Train-Test Split*, penelitian ini juga melakukan evaluasi menggunakan *10-Fold Stratified Cross Validation*. Metode ini digunakan untuk memperoleh estimasi performa model yang lebih stabil dengan mempertahankan proporsi setiap kelas pada setiap *fold*.

Tabel 4. Hasil *10-Fold Stratified Cross Validation*

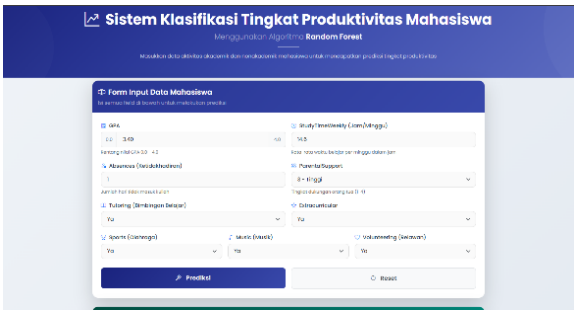
Fold	Accuracy (%)
1	92,92
2	94,17
3	93,72
4	90,79
5	90,79
6	93,31
7	90,79
8	93,72
9	93,31
10	93,72
Rata - rata	**92,73**
Standar Deviasi	**1,30**

Berdasarkan hasil *10-Fold Stratified Cross Validation*, model *Random Forest* memperoleh rata-rata *Accuracy* sebesar 92,73% dengan standar deviasi sebesar 1,30%. Nilai rata-rata akurasi yang tinggi menunjukkan bahwa model memiliki kemampuan klasifikasi yang baik pada berbagai pembagian data. Selain itu, nilai standar deviasi yang rendah menunjukkan bahwa performa model relatif konsisten pada setiap *fold*, sehingga model memiliki kemampuan generalisasi yang baik dan tidak bergantung pada satu kali pembagian data menggunakan metode *Train-Test Split.*

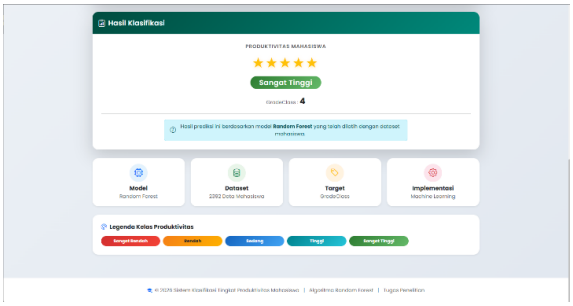
F. *Hasil Deployment*

Sebagai implementasi hasil penelitian, dibangun sebuah sistem berbasis web menggunakan *framework Flask*. Sistem ini dirancang untuk memudahkan pengguna dalam melakukan prediksi tingkat produktivitas mahasiswa berdasarkan data akademik dan nonakademik yang dimasukkan.

Pengguna cukup mengisi data seperti *GPA*, *StudyTimeWeekly*, *Absences*, *Tutoring*, *ParentalSupport*, *Extracurricular*, *Sports*, *Music*, dan *Volunteering*. Data tersebut kemudian diproses menggunakan model *Random Forest* yang telah dilatih sebelumnya, sehingga sistem dapat menghasilkan prediksi tingkat produktivitas mahasiswa dalam lima kategori, yaitu Sangat Rendah, Rendah, Sedang, Tinggi, dan Sangat Tinggi.



Gambar 7. Halaman Input Sistem



Gambar 8. Hasil Prediksi Sistem

Hasil prediksi yang ditampilkan sistem mengacu pada lima kategori produktivitas yang telah dijelaskan pada tabel 3.Interpretasi Kategori Produktivitas.

Sistem yang dikembangkan ditujukan untuk digunakan oleh dosen wali, program studi, maupun bagian akademik sebagai alat bantu dalam melakukan evaluasi produktivitas mahasiswa. Hasil klasifikasi yang diperoleh dapat dimanfaatkan sebagai informasi awal untuk mengidentifikasi mahasiswa yang memerlukan pendampingan akademik, pemberian layanan bimbingan belajar, maupun evaluasi terhadap keterlibatan mahasiswa dalam aktivitas akademik dan nonakademik. Dengan demikian, sistem tidak hanya berfungsi sebagai media prediksi, tetapi juga mendukung proses pengambilan keputusan dalam upaya meningkatkan kualitas pembelajaran di perguruan tinggi.

IV. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa algoritma Random Forest berhasil diterapkan untuk mengklasifikasikan tingkat produktivitas mahasiswa berdasarkan aktivitas akademik dan nonakademik menggunakan Student Performance Dataset yang terdiri dari 2.392 data mahasiswa. Proses klasifikasi dilakukan dengan memanfaatkan atribut *GPA, StudyTimeWeekly, Absences, Tutoring, ParentalSupport, Extracurricular, Sports, Music, dan Volunteering* sebagai variabel masukan, sedangkan *GradeClass* digunakan sebagai variabel target klasifikasi.

Hasil pengujian menunjukkan bahwa model *Random Forest* memperoleh tingkat *Accuracy* sebesar 91,44%, dengan nilai *Precision, Recall,* dan *F1-Score* yang menunjukkan performa klasifikasi yang baik. Selain itu, hasil evaluasi menggunakan *10-Fold Stratified Cross Validation* memperoleh rata-rata *Accuracy* sebesar 92,73% dengan standar deviasi 1,30%, yang menunjukkan bahwa model memiliki performa yang stabil dan kemampuan generalisasi yang baik pada berbagai pembagian data.

Selain menghasilkan model klasifikasi yang akurat, penelitian ini juga berhasil mengimplementasikan model *Random Forest* ke dalam sebuah sistem berbasis web menggunakan *Flask*. Sistem yang dikembangkan mampu menerima data aktivitas akademik dan nonakademik mahasiswa, kemudian memberikan hasil klasifikasi tingkat produktivitas ke dalam lima kategori, yaitu Sangat Rendah, Rendah, Sedang, Tinggi, dan Sangat Tinggi. Sistem ini dapat dimanfaatkan oleh dosen, program studi, maupun pihak akademik sebagai media pendukung dalam melakukan evaluasi produktivitas mahasiswa dan membantu proses pengambilan keputusan terhadap mahasiswa yang memerlukan pendampingan akademik.

Penelitian ini masih memiliki beberapa keterbatasan. Dataset yang digunakan merupakan satu dataset publik sehingga belum sepenuhnya merepresentasikan karakteristik mahasiswa pada berbagai perguruan tinggi. Selain itu, penelitian ini belum melakukan validasi menggunakan data mahasiswa dari institusi lain, sehingga kemampuan generalisasi model pada kondisi nyata masih perlu diuji lebih lanjut. Penelitian ini juga belum melakukan perbandingan performa *Random Forest* dengan algoritma klasifikasi lain, seperti *Decision Tree, Naïve Bayes,* maupun *Support Vector Machine,* sehingga belum dapat diketahui metode yang memberikan performa terbaik untuk kasus klasifikasi produktivitas mahasiswa. Oleh karena itu, penelitian selanjutnya disarankan menggunakan dataset yang lebih beragam, melakukan proses hyperparameter tuning, membandingkan performa beberapa algoritma klasifikasi, serta menguji model menggunakan data mahasiswa dari berbagai perguruan tinggi

agar diperoleh model yang lebih representatif dan memiliki kemampuan generalisasi yang lebih baik.

REFERENSI

[1] & M. 2023. Taufik, "arus jurnal pendidikan (AJUP) kegiatan ekstrakurikuler dalam meningkatkan prestasi akademik," *Taufik, ma'arif. 2023*, 2023.

[2] F. Mawaddah, N. K. Daulay, and H. Fauza, "Strategi Kepala Madrasah dalam Meningkatkan Prestasi Non Akademik Siswa melalui Kegiatan Ekstrakurikuler Di MTs Negeri 3 Medan," *Al-Tarbiyah J. Ilmu Pendidik. Islam*, vol. 1, no. 4, p. 100, 2023.

[3] J. C. Saksana, "AnalisisPengaruhMotivasiBelajar,KemampuanKognitifdan Manajemen Waktu Terhadap Prestasi Belajar Mahasisw," *J. Pendidik. dan Kebud. Nusant.*, vol. 2, no. 4, pp. 172–181, 2024.

[4] S. A. A. Kharis and A. H. A. Zili, "Learning Analytics dan Educational Data Mining pada Data Pendidikan," *J. Ris. Pembelajaran Mat. Sekol.*, vol. 6, no. 1, pp. 12–20, 2022, doi: 10.21009/jrpms.061.02.

[5] N. Hendrastuty, "Penerapan Data Mining Menggunakan Algoritma K-Means Clustering Dalam Evaluasi Hasil Pembelajaran Siswa," *J. Ilm. Inform. dan Ilmu Komput.*, vol. 3, no. 1, pp. 46–56, 2024, doi: 10.58602/jima-ilkom.v3i1.26.

[6] A. Putra Argadinata, D. Abdul Fatah, and H. Sukri, "Klasifikasi Kualitas Buah Apel Menggunakan Metode Random Forest," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 9, no. 2, pp. 2016–2022, 2025, doi: 10.36040/jati.v9i2.12854.

[7] N. I. Yaman, A. R. Juwita, S. A. P. Lestari, and S. Faisal, "Perbandingan Kinerja Algoritma Decision Tree dan Random Forest untuk Klasifikasi Nutrisi pada Makanan Cepat Saji," *J. Algoritm.*, vol. 21, no. 2, pp. 184–196, 2024, doi: 10.33364/algoritma/v.21-2.1649.

[8] I. Maulita and A. M. Wahid, "Prediksi Magnitudo Gempa Menggunakan Random Forest, Support Vector Regression, XGBoost, LightGBM, dan Multi-Layer Perceptron Berdasarkan Data Kedalaman dan Geolokasi," *J. Pendidik. dan Teknol. Indones.*, vol. 4, no. 5, pp. 221–232, 2024, doi: 10.52436/1.jpti.470.

[9] P. Aliya, S. Nurbaeti, A. Firdaus, F. Suparman, and M. Pd, "3 1,2,3," vol. 11, no. April, pp. 156–165, 2025.

[10] A. Pratama, A. C. Nurcahyo, and L. Firgia, "Penerapan Machine Learning dengan Algoritma Logistik Regresi untuk Memprediksi Diabetes," *Pros. CORISINDO 2023*, p. 117, 2023, [Online]. Available: https://stmikpontianak.org/ojs/index.php/corisindo/article/view/30%0Ahttps://stmikpontianak.org/ojs/index.php/corisindo/article/download/30/22

[11] I. Setiawan, R. Fina Antika Cahyani, and I. Sadida, "Exploring Complex Decision Trees: Unveiling Data Patterns and Optimal Predictive Power," *J. Innov. Futur. Technol.*, vol. 5, no. 2, pp. 112–123, 2023, doi: 10.47080/iftech.v5i2.2829.

[12] D. B. Saputra, V. Atina, and F. E. Nastiti, "Penerapan Model Crisp-Dm Pada Prediksi Nasabah Kredit Menggunakan Algoritma Random Forest," *IDEALIS Indones. J. Inf. Syst.*, vol. 7, no. 2, pp. 240–247, 2024, doi: 10.36080/idealis.v7i2.3244.

[13] D. Triyana, M. Muharrom Al Haromainy, and H. Maulana, "Implementasi Metode Ensemble Majority Vote Pada Algoritma Naive Bayes Dan Random Forest Untuk Analisis Sentimen Twitter Harga Tiket Pesawat Domestik," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 4, pp. 7885–7894, 2024, doi: 10.36040/jati.v8i4.10475.

[14] M. Azhari, Z. Situmorang, and R. Rosnelly, "Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, Random Forest, SVM dan Naive Bayes," *J. Media Inform. Budidarma*, vol. 5, no. 2, p. 640, 2021, doi: 10.30865/mib.v5i2.2937.

[15] S. Helmiyah, R. Pramestiawan, P. Studi Pendidikan Informatika, and S. Tinggi Keguruan dan Ilmu Pendidikan Rosalia Lampung, "Analisis Komparatif Algoritma Machine Learning dengan Metrik Akurasi, Presisi, Recall, dan F1-Score pada Dataset Kacang Kering," *J. Ilmu Komput. Dan Teknol.*, vol. 6, no. 3, pp. 152–159, 2025, [Online]. Available: https://doi.org/10.35960/ikomti.v6i3.2031%0Ahttp://creativecommons.org/licences/by/4.0/%0Ahttp://ejournal.uhb.ac.id/index.php/IKOMTI