

EARLY WARNING SYSTEM KELULUSAN OF STUDENT GRADUATION USING THE RANDOM FOREST ALGORITHM

Elcky Mardiantho Tedut¹⁾, Wahyuni Wahyuni²⁾, dan Pitrasacha Adytia³⁾

^{1,2}Teknik Informatika, Sekolah Tinggi Manajemen Informatika dan Komputer Widya Cipta Dharma
^{1,2,3}alamat, kota, kodepos
2143045@wicida.ac.id¹⁾, wahyuni@wicida.ac.id²⁾, pitra@wicida.ac.id³⁾

ABSTRACT

On-time graduation is a key indicator of success for higher education institutions. At STMIK Widya Cipta Dharma, the average student study period is 4.7 years, indicating many students do not graduate on time. This situation can affect the institution's reputation and program accreditation.

This study aims to develop an Early Warning System (EWS) using the Random Forest algorithm to predict student graduation status: On Time, Delayed, or Drop Out (DO). The data includes GPA, semester GPA from the first to sixth semester, total completed credits, current semester, study load, and payment status.

Data was collected through literature review, interviews, and documentation to obtain academic records. The resulting system can classify student graduation status with 99% accuracy. Additionally, rule extraction from the Random Forest model provides logical interpretations of the system's decisions, serving as a basis for academic policy-making.

The system is available as a website, enabling both manual and bulk classification via Excel files, with results presented interactively. Despite its high accuracy, the results may indicate possible data leakage.

Keywords: words that serve as keys to facilitate the understanding of the scientific article, 3-6 keywords

Example: Information System, Employee Selection, OpenGL. [Times New Roman 10, italic, left aligned]

EARLY WARNING SYSTEM KELULUSAN MAHASISWA MENGGUNAKAN ALGORITMA RANDOM FOREST

ABSTRAK

Kelulusan tepat waktu merupakan salah satu indikator penting dalam menilai keberhasilan suatu institusi pendidikan tinggi. Di STMIK Widya Cipta Dharma, rata-rata masa studi mahasiswa mencapai 4,7 tahun, yang menunjukkan bahwa masih banyak mahasiswa yang belum lulus tepat waktu. Kondisi ini berpotensi memengaruhi reputasi institusi serta akreditasi program studinya.

Penelitian ini bertujuan mengembangkan *Early Warning System* (EWS) menggunakan algoritma Random Forest untuk memprediksi status kelulusan mahasiswa, apakah Tepat Waktu, Terlambat, atau Drop Out (DO). Data yang digunakan mencakup IPK, IP per Semester dari semester satu hingga enam, jumlah total SKS yang telah diselesaikan, semester, bobot studi, dan status pembayaran.

Pengumpulan data dilakukan melalui studi literatur, wawancara, dan dokumentasi untuk memperoleh data akademik. Hasilnya, sistem yang dikembangkan mampu mengklasifikasikan status kelulusan mahasiswa dengan tingkat akurasi mencapai 99%. Selain itu, dilakukan ekstraksi aturan dari model Random Forest untuk memperoleh interpretasi logis dari keputusan sistem, yang dapat menjadi dasar dalam perumusan kebijakan akademik.

Sistem ini tersedia dalam bentuk website yang memungkinkan klasifikasi manual maupun massal melalui file Excel, dan menyajikan hasil secara interaktif. Meski akurasinya tinggi, hal ini bisa menjadi indikasi adanya kebocoran data (*data leakage*).

Kata Kunci: Early Warning System, Random Forest, Prediksi Kelulusan, Klasifikasi, *Rule Implementation*, STMIK Widya Cipta Dharma

1. PENDAHULUAN

Kelulusan tepat waktu merupakan salah satu indikator utama keberhasilan suatu institusi pendidikan tinggi. Berdasarkan data pada laman

PDDIKTI Di STMIK Widya Cipta Dharma, rata-rata masa studi lulusan mencapai 4.7 tahun, menunjukkan bahwa sejumlah mahasiswa tidak

lulus tepat waktu. Hal ini dapat berdampak pada reputasi institusi dan akreditasi program studi. Keterlambatan kelulusan mahasiswa sering kali disebabkan oleh berbagai faktor, baik internal maupun eksternal, seperti performa akademik, motivasi, manajemen waktu, hingga kondisi pribadi mahasiswa. Mengingat pentingnya kelulusan tepat waktu, diperlukan sebuah Early Warning System yang mampu mengklasifikasikan status akademik mahasiswa secara akurat yang kemudian dapat digunakan untuk memprediksi kelulusan mahasiswa.

Early Warning System (EWS) adalah sebuah sistem yang dirancang untuk memberikan peringatan dini terkait potensi risiko atau masalah yang dapat muncul dalam suatu proses atau situasi tertentu. Dalam konteks pendidikan tinggi, EWS digunakan untuk memantau dan menganalisis data mahasiswa guna mengidentifikasi mahasiswa yang berpotensi menghadapi kesulitan akademik atau risiko kelulusan tidak tepat waktu. Dengan menggunakan EWS, institusi pendidikan dapat mengambil langkah-langkah intervensi secara proaktif untuk membantu mahasiswa menyelesaikan studi mereka tepat waktu. Dalam penelitian ini, algoritma Random Forest dan peraturan(*rule*) akademik digunakan sebagai inti dari EWS untuk mengklasifikasi status kelulusan mahasiswa.

2. RUANG LINGKUP

Dalam penelitian ini permasalahan mencakup:

1. Cakupan permasalahan
Penelitian ini mencakup permasalahan kelulusan mahasiswa yang tidak tepat waktu, dengan fokus pada upaya prediksi status kelulusan menggunakan algoritma Random Forest.
2. Batasan-batasan penelitian
Penelitian ini memiliki beberapa batasan agar ruang lingkupnya lebih terfokus dan terarah. Pertama, penelitian hanya dilakukan kepada mahasiswa yang terdaftar di STMIK Widya Cipta Dharma. Data yang digunakan dalam penelitian ini terbatas pada mahasiswa reguler, yaitu mereka yang bukan mahasiswa alih jenjang dan tidak pernah mengambil cuti akademik selama masa studi. Meskipun data yang digunakan berasal dari beberapa program studi di STMIK Widya Cipta Dharma, fokus utama

evaluasi dan implementasi model akan diarahkan secara khusus pada Program Studi Teknik Informatika (TI). Atribut atau variabel yang digunakan dalam proses analisis meliputi IPK (Indeks Prestasi Kumulatif), jumlah total SKS (Satuan Kredit Semester) yang telah ditempuh, bobot nilai, IP (Indeks Prestasi) dari semester 1 hingga semester 6, semester aktif, serta status pembayaran. Dalam penelitian ini, algoritma yang digunakan untuk membangun model prediksi adalah algoritma Random Forest.

3. Rencana hasil yang didapatkan

Hasil yang diharapkan dari penelitian ini adalah terbentuknya sistem Early Warning System berbasis web yang mampu mengklasifikasikan status kelulusan mahasiswa (tepat waktu, terlambat, atau *Drop Out*) berdasarkan data historis menggunakan algoritma Random Forest.

3. BAHAN DAN METODE [Huruf Kapital]

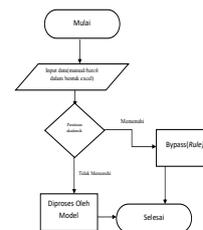
Adapun bahan dan metode yang digunakan sebagai berikut:

3.1 Cross Industry Standard Process for Data Mining

CRISP-DM (*Cross-Industry Standard Process for Data Mining*) adalah kerangka kerja yang sering digunakan dalam proyek *data mining* dan *machine learning*. CRISP-DM terdiri dari enam tahap yang sistematis, membantu mengarahkan proyek dari awal hingga selesai, yaitu *Business understanding*, *Data understanding*, *Data preparation*, *modeling*, *evaluation* dan *deployment*. **International Business Machines Corporation. IBM SPSS Modeler CRISP-DM guide (Versi 18, Rilis 3). IBM Corp.**

3.2 Tahapan Pengembangan Sistem

Tahap pengembangan sistem dilakukan berdasarkan kerangka kerja CRISP-DM (*Cross-Industry Standard Process for Data Mining*) yang mencakup enam langkah utama. Langkah-langkah ini memastikan proses penelitian berjalan secara sistematis dan terstruktur, dimulai dari memahami kebutuhan hingga implementasi sistem berbasis *machine learning* yang tepat sasaran. Setiap langkah dalam kerangka kerja ini dirancang untuk saling terhubung dan memungkinkan proses iteratif apabila ditemukan kekurangan dalam tahapan sebelumnya.



Gambar 1. Gambaran Sistem

4. PEMBAHASAN

Menampilkan aplikasi yang dibagun, baik dalam bentuk software, hardware, jaringan komputer, dan lain-lain.

Sertakan data pendukung yang berupa desain/perancangan, tabel, grafik, gambar, atau alat penolong lain seperlunya untuk memperjelas dan mempersingkat uraian yang harus diberikan.

4.1 Business Understanding

Tujuan utama dari penelitian ini adalah untuk memberikan insight berbasis data kepada pihak kampus, khususnya bagian akademik, dalam rangka menyusun kebijakan pembinaan mahasiswa yang lebih efektif. Dengan model prediksi yang akurat, diharapkan kampus dapat melakukan pendampingan lebih dini kepada mahasiswa yang terindikasi mengalami risiko keterlambatan kelulusan, sehingga dapat meningkatkan efisiensi proses pendidikan secara keseluruhan

4.2 Data Understanding

1. Sumber Data

Data yang digunakan oleh peneliti untuk keperluan analisis dan pelatihan model Random Forest berasal dari institusi pendidikan tempat penulis menempuh studi, yaitu STMIK Widya Cipta Dharma. Data tersebut diperoleh melalui metode dokumentasi, yang mana pihak Pusat Komputer kampus memberikan data kepada penulis tanpa penulis mengakses langsung database internal. Adapun data yang disediakan mencakup informasi seperti NIM, angkatan, kode program studi, kode mata kuliah, nama mata kuliah, jumlah SKS per mata kuliah, nilai mutu, nilai huruf, bobot, tahun akademik, semester, judul skripsi, tanggal lulus mahasiswa. Dan Status Pembayaran(Approved) mahasiswa. Seluruh data ini digunakan sebagai dasar dalam membangun model prediksi kelulusan mahasiswa

2. Jumlah Data

Data memiliki 122633 baris dan 12 kolom.

3. Tipe Data

Tipe data yang tersedia di dalam dataset adalah sebagai berikut

```
[9]: print(data.columns)
print(data.dtypes)

Index(['nim', 'angkatan', 'kdprodi', 'kdek', 'namamatakuliah', 'nilaimutu',
       'nilaihuruf', 'bobot', 'ta', 'smt', 'judulskripsi', 'tglulus'],
      dtype='object')

nim          object
angkatan    int64
kdprodi      int64
kdek         object
namamatakuliah  object
nilaimutu   float64
ilaihuruf    object
bobot       float64
ta           object
smt         float64
judulskripsi  object
tglulus     datetime64[ns]
dtype: object
```

4.3 Data Preparation

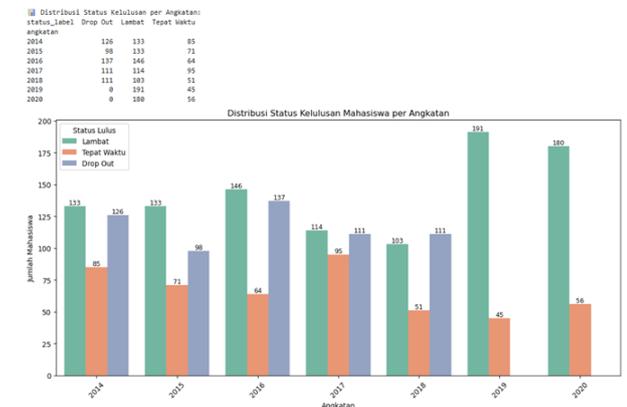
1. Pembersihan Data (Data Cleaning)

Data mentah yang diperoleh dari sumber awal sering kali mengandung nilai yang hilang (missing values), duplikasi, atau kesalahan input. Oleh karena itu, dilakukan proses pembersihan data

seperti menghapus data duplikat, mengisi atau menghapus nilai kosong, serta memastikan konsistensi format data..

2. Transformasi Data (Data Transformation)

Data yang telah dibersihkan kemudian ditransformasikan agar sesuai dengan kebutuhan algoritma yang akan digunakan. Beberapa proses yang dilakukan antara lain: Penggabungan atau pemisahan fitur berdasarkan kebutuhan analisis. Contohnya, beberapa variabel dapat digabungkan menjadi satu fitur baru yang lebih informatif, atau fitur yang kurang relevan diubah agar lebih bermakna dalam konteks model. Yakni hanya variabel IPK,IP Semester 1-6,Bobot,jumlah sks,semester dan status bayar. Adapun Semester dibiarkan actual karena jika diratakan maka akan menyebabkan performa model yang turun hingga 20-25%.



Gambar di atas menampilkan distribusi status kelulusan mahasiswa berdasarkan angkatan dari tahun 2014 hingga 2020. Tiga kategori status kelulusan yang ditampilkan adalah Drop Out, Lambat, dan Tepat Waktu, yang masing-masing direpresentasikan dengan warna berbeda dalam grafik batang.

Berdasarkan visualisasi tersebut, dapat disimpulkan beberapa hal penting:

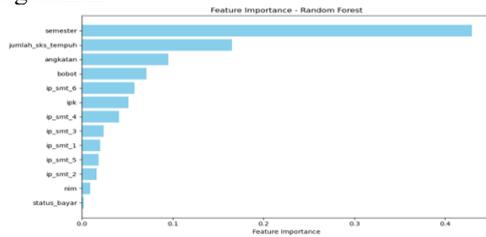
1. Jumlah kelulusan tepat waktu tertinggi terjadi pada angkatan 2017, yakni sebanyak 95 mahasiswa, disusul oleh angkatan 2014 dengan 85 mahasiswa. Jumlah ini cenderung menurun pada angkatan-angkatan berikutnya.
2. Kelulusan lambat paling dominan terjadi pada angkatan 2019 sebanyak 191 mahasiswa, diikuti oleh angkatan 2020 dengan 180 mahasiswa. Ini menunjukkan bahwa pada dua angkatan tersebut, sebagian besar mahasiswa menyelesaikan studi melebihi waktu normal.
3. Angka Drop Out tertinggi terdapat pada angkatan 2016 (137 mahasiswa) dan 2014 (126 mahasiswa), serta angka yang signifikan juga terlihat pada angkatan 2017 dan 2018 (masing-masing 111 mahasiswa).
4. Angkatan 2019 dan 2020 menunjukkan tren berbeda, di karena keterbatasan informasi dapat

diasumsikan dengan nama jika mahasiswa tidak memiliki tanggal lulus maka mahasiswa tersebut terlambat.

Secara keseluruhan, grafik ini menggambarkan bahwa sebagian besar mahasiswa mengalami keterlambatan dalam menyelesaikan studi, dan angka putus studi juga cukup tinggi di beberapa angkatan. Oleh karena itu, institusi perlu melakukan evaluasi dan intervensi, seperti sistem peringatan dini atau pembimbingan akademik yang lebih ketat, untuk meningkatkan angka kelulusan tepat waktu dan menurunkan tingkat putus studi.

3. Seleksi Fitur (*Feature Selection*)

Berdasarkan hasil pemodelan menggunakan algoritma *Random Forest*, diketahui bahwa fitur semester memiliki kontribusi paling dominan dalam menentukan prediksi kelulusan mahasiswa. Hal ini terlihat dari nilai importance yang paling tinggi dibandingkan fitur lainnya, yaitu di atas 0.4. Artinya, semester aktif mahasiswa sangat mempengaruhi apakah mahasiswa tersebut akan lulus tepat waktu, terlambat, atau bahkan tidak lulus. Selanjutnya, fitur jumlah_sks_tempuh menempati posisi kedua dalam hal pengaruh terhadap output model. Hal ini menunjukkan bahwa jumlah SKS yang telah diambil oleh mahasiswa juga merupakan indikator penting untuk memprediksi kelulusan. Fitur-fitur seperti angkatan, bobot, dan IP Semester (ip_smt_1 hingga ip_smt_6) juga berkontribusi meskipun tidak sebesar dua fitur sebelumnya. Menariknya, fitur status_bayar dan nim memiliki pengaruh yang sangat rendah, hampir mendekati nol, yang menandakan bahwa keduanya kurang relevan dalam menentukan prediksi berdasarkan data yang digunakan.



4. Cek Data Leakage

Data Leakage sendiri sulit dideteksi secara otomatis, namun kita bisa tetap melakukan pencegahan dengan mengecek korelasi fitur terhadap target.

Berdasarkan analisis yang dilakukan terhadap 13 fitur dan target status_lulus, ditemukan adanya potensi *data leakage* pada fitur status_bayar. Fitur ini berpotensi memberikan informasi yang terlalu dekat atau terlalu berhubungan langsung dengan label target, sehingga dapat menyebabkan model melakukan prediksi tidak secara alami, melainkan berdasarkan informasi "bocor" dari masa depan atau hasil akhir mahasiswa. Termasuk kolom semester yang tak terdeteksi.

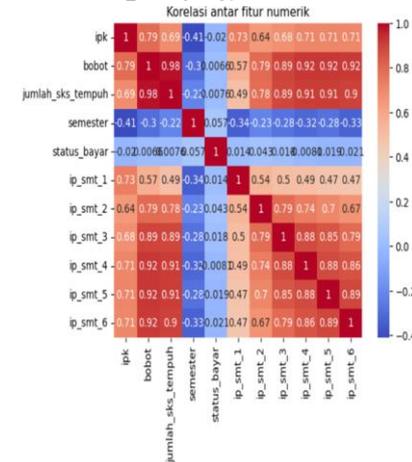
Jumlah fitur: 13
Target : status_lulus

▲ POTENSI DATA LEAKAGE DITEMUKAN di kolom-kolom berikut:
- status_bayar

📊 Korelasi Fitur Numerik terhadap Target:

semester	0.069704
status_bayar	-0.038199
angkatan	-0.269395
ip_smt_1	-0.302543
ip_smt_2	-0.335244
ip_smt_3	-0.365989
ip_smt_5	-0.429637
ip_smt_4	-0.450823
ipk	-0.474216
ip_smt_6	-0.482156
bobot	-0.512567
jumlah_sks_tempuh	-0.513996

Name: status_lulus, dtype: float64



4. Pembagian Dataset

Dataset yang telah diproses kemudian dibagi menjadi dua bagian, yaitu data latih (training set) dan data uji (testing set). Umumnya, pembagian ini dilakukan dengan perbandingan 80:20 atau 70:30 untuk memastikan model dapat dilatih dan diuji secara adil. Pada penelitian ini, data dibagi dengan proporsi 80% untuk pelatihan dan 20% untuk pengujian.

Dengan melalui tahapan data preparation ini, diharapkan data yang digunakan dalam pemodelan memiliki kualitas yang baik dan siap digunakan dalam proses klasifikasi menggunakan algoritma *Random Forest*.

4.4 Modeling

Pada tahap modeling, dilakukan pembangunan model klasifikasi menggunakan algoritma *Random Forest*. Algoritma ini dipilih karena kemampuannya dalam menangani data dengan jumlah fitur yang tidak terlalu banyak serta menghasilkan model yang cukup akurat dan tahan terhadap *overfitting*.

Langkah-langkah yang dilakukan pada tahap modeling adalah sebagai berikut:

1. Inisialisasi Model

Model *Random Forest* diinisialisasi dengan parameter tertentu, seperti jumlah pohon keputusan (*n_estimators*) sebanyak 200 dan nilai *random_state*

untuk memastikan hasil yang konsisten saat pelatihan ulang.

2. Pelatihan Model

Model dilatih menggunakan data latih (X_{train} dan y_{train}) yang telah dipersiapkan sebelumnya. Proses pelatihan ini membuat model mempelajari pola hubungan antara fitur input (misalnya IPK, bobot, jumlah SKS tempuh) dengan target output yaitu status kelulusan mahasiswa. Contoh:

```
1. model = RandomForestClassifier(n_estimators=100,
    random_state=42)
2. model.fit(X_train, y_train)
3. print(" Model berhasil dilatih dengan SMOTE!")
```

3. Evaluasi Awal Model

Setelah pelatihan selesai, model diuji menggunakan data uji (X_{test}) untuk menghasilkan prediksi. Kemudian, prediksi tersebut dibandingkan dengan data aktual (y_{test}) untuk menilai performa model. Beberapa metrik evaluasi yang digunakan antara lain accuracy, precision, recall, dan F1-score.

4. Penyimpanan Model

Setelah model Random Forest selesai dilatih, langkah berikutnya adalah menyimpan model tersebut ke dalam sebuah file menggunakan library pickle. Tujuannya agar model bisa dipakai kembali tanpa harus melatih ulang dari awal, sehingga menghemat waktu dan sumber daya saat implementasi selanjutnya.

5. Pengujian Ulang Model yang Dimuat

Model yang telah disimpan dapat dimuat ulang dan diuji kembali pada data uji untuk memastikan bahwa performa model tetap terjaga. Hal ini juga mempermudah proses deployment model dalam aplikasi nyata. Dengan tahap modeling ini, dihasilkan model klasifikasi yang siap digunakan untuk memprediksi status kelulusan mahasiswa berdasarkan fitur yang telah ditentukan, serta dapat dievaluasi performanya dengan metrik-metrik yang relevan.

4.5 Evaluation

Tahap evaluasi dilakukan untuk mengukur performa model *Random Forest* yang telah dilatih. Evaluasi ini bertujuan untuk memastikan model mampu memberikan prediksi yang akurat terhadap status kelulusan mahasiswa berdasarkan data uji.

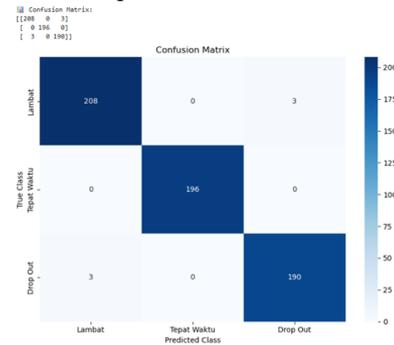
1. Prediksi Data Uji

Model yang sudah dilatih digunakan untuk memprediksi label (status_lulus)

2. Confusion Matrix

Confusion matrix dibuat untuk melihat distribusi hasil prediksi model terhadap data aktual. *Confusion matrix* memberikan informasi mengenai jumlah *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negative (FN)*. Adapun

confusion Matrix yang digunakan berukuran 3x3 untuk memperlihatkan *multiclass classification*.



3. Classification Report

Menggunakan fungsi *classification report* dari Scikit-Learn untuk menampilkan metrik evaluasi lengkap seperti precision, recall, F1-score, dan support.

Contoh:

```
Classification Report:
              precision    recall  f1-score   support

   Lambat      0.99      0.99      0.99         211
  Tepat Waktu  1.00      1.00      1.00         196
   Drop Out    0.98      0.98      0.98         193

 accuracy          0.99         600
 macro avg         0.99         600
weighted avg         0.99         600
```

Akurasi Total: 0.99

4. Accuracy Score

Menghitung akurasi keseluruhan model, yaitu proporsi prediksi yang benar dibandingkan data uji. Hasil pengujian menunjukkan angka 0.968 dibulatkan menjadi 0.99 yang berarti model memiliki akurasi 99% dan margin of error 1%

5. Perhitungan Metrik

Gambar di bawah menunjukkan hasil evaluasi model klasifikasi Random Forest yang digunakan untuk mengelompokkan mahasiswa ke dalam tiga kategori status kelulusan, yaitu Lambat, Tepat Waktu, dan Drop Out. Berdasarkan perhitungan manual terhadap metrik evaluasi, model menunjukkan performa yang sangat tinggi dan konsisten di ketiga kelas. Pada kelas Lambat, precision, recall, dan F1-score masing-masing mencapai 0.9858, dengan specificity sebesar 0.9923 dan akurasi per kelas mencapai 0.9900. Hal ini menunjukkan bahwa model sangat baik dalam mengenali mahasiswa yang lulus lambat serta minim kesalahan dalam membedakan mahasiswa dari kelas lain.

Performa model pada kelas Tepat Waktu bahkan lebih tinggi, dengan precision, recall, specificity, dan F1-score yang seluruhnya bernilai sempurna, yaitu 1.0000. Artinya, model tidak melakukan kesalahan sama sekali dalam mengklasifikasikan mahasiswa yang lulus tepat waktu, dengan akurasi per kelas mencapai 100%.

Sementara itu, pada kelas Drop Out, model juga mencatat precision, recall, dan F1-score sebesar 0.9845, dengan specificity sebesar 0.9926 dan akurasi per kelas mencapai 0.9900. Ini menunjukkan bahwa model juga sangat baik dalam mengidentifikasi mahasiswa yang tidak menyelesaikan studi.

Secara keseluruhan, hasil evaluasi menunjukkan bahwa model memiliki tingkat akurasi dan keandalan yang sangat tinggi dalam memprediksi status kelulusan mahasiswa. Namun demikian, performa sempurna pada kelas Tepat Waktu menjadi sinyal awal adanya potensi kebocoran data (data leakage). Salah satu fitur yang patut dicurigai sebagai sumber kebocoran tersebut adalah informasi mengenai semester terakhir mahasiswa.

Fitur ini sangat mungkin memiliki korelasi langsung dengan status kelulusan, sehingga perlu dilakukan evaluasi lanjutan terhadap fitur input yang digunakan agar model tetap memiliki kemampuan generalisasi yang baik terhadap data baru.

Penting dicatat bahwa akurasi model bukanlah segalanya dalam mengidentifikasi model tersebut baik atau tidak, perlu ada evaluasi lanjutan terkait semua fitur yang digunakan seta keandalan dan keabsahan dari data sumber yang digunakan untuk proses pelatihan model.

```
■ Kelas 0 (Lambat):
True Positive (TP) : 208
False Positive (FP) : 3
False Negative (FN) : 3
True Negative (TN) : 386
Precision : 0.9858
Recall (Sensitivity) : 0.9858
Specificity : 0.9923
F1-Score : 0.9858
Accuracy (Per Kelas) : 0.9900
```

```
■ Kelas 1 (Tepat Waktu):
True Positive (TP) : 196
False Positive (FP) : 0
False Negative (FN) : 0
True Negative (TN) : 404
Precision : 1.0000
Recall (Sensitivity) : 1.0000
Specificity : 1.0000
F1-Score : 1.0000
Accuracy (Per Kelas) : 1.0000
```

```
■ Kelas 2 (Drop Out):
True Positive (TP) : 190
False Positive (FP) : 3
False Negative (FN) : 3
True Negative (TN) : 404
Precision : 0.9845
Recall (Sensitivity) : 0.9845
Specificity : 0.9926
F1-Score : 0.9845
Accuracy (Per Kelas) : 0.9900
```

4.6 Deployment

Tahap deployment adalah proses menerapkan model Machine Learning yang telah dilatih dan diuji ke dalam bentuk yang dapat digunakan secara praktis oleh pengguna atau sistem lain. Pada penelitian ini, deployment dilakukan dengan cara menyimpan model klasifikasi Random Forest ke dalam format file .pkl

menggunakan pustaka pickle. File model ini nantinya dapat digunakan kembali untuk melakukan prediksi terhadap data baru tanpa perlu melatih ulang model dari awal.

1) Website

Backend

Backend dari sistem ini dibangun menggunakan framework **Flask** untuk melayani permintaan prediksi/klasifikasi dari pengguna melalui website. Backend ini berfungsi untuk memuat model Random Forest yang telah dilatih, menerima data masukan baik secara manual maupun file Excel, serta mengembalikan hasil klasifikasi kepada pengguna.

Fitur Backend

2. Flask + Flask-CORS

Framework Flask digunakan karena ringan dan mudah digunakan. Ditambahkan juga Flask-CORS untuk mengizinkan akses dari frontend yang berjalan di domain berbeda (misalnya dari *framework* Next.js).

3. Load Model

Model yang telah dilatih dan disimpan dalam file model_ke7Revisi semester asli .pkl dimuat ulang saat backend dijalankan

```
14: # Load model
15: with open("model_ke7Revisi semester asli .pkl", "rb") as f:
16:     model = pickle.load(f)
```

4. Fungsi:

Prediksi/Klasifikasi Manual:

Endpoint /predict_manual menerima input dari form: IPK, Bobot, dan Jumlah SKS. Jika IPK di bawah 3, otomatis diklasifikasikan sebagai "Terlambat", sisanya diprediksi oleh model.

```
29: @app.route('/predict_manual',
methods=['POST'])
30: def predict_manual():
```

Prediksi/Klasifikasi File Excel:

Endpoint /predict_excel menerima file Excel yang di-upload oleh pengguna. File harus memiliki kolom: ipk, bobot, dan jumlah_sks_tempuh. Baris dengan IPK < 3 otomatis diklasifikasi sebagai "Terlambat", sisanya diprediksi oleh model.

Hasil ditampilkan dalam format 6able HTML:

```
50: @app.route('/predict_excel', methods=['POST'])
51: def predict_excel():
```

4. Penyimpanan File Upload

File Excel disimpan di folder uploads yang otomatis dibuat saat aplikasi dijalankan.

```
11: app.config['UPLOAD_FOLDER'] = 'uploads'
12: os.makedirs(app.config['UPLOAD_FOLDER'],
exist_ok=True)
```

5. Menjalankan Backend

Aplikasi dijalankan secara 6able untuk pengujian dan nantinya bisa di-deploy ke server.

```
Contoh
99: if __name__ == '__main__':
100:     app.run(debug=True)
```

FrontEnd

1. Input Manual Mahasiswa

Form berisi tiga input: IPK, Bobot, dan Jumlah SKS Tempuh. IP semester 1 hingga 6, semester dan status pembayaran. Kemudian Data dikirim ke endpoint Flask (/predict_manual) melalui metode POST menggunakan FormData, dan hasilnya akan ditampilkan langsung.

Hasil dikembalikan dalam bentuk teks (HTML friendly) dan ditampilkan ke pengguna secara langsung.

```
25: const handleSubmitManual = async () => {
26:   const formData = new FormData();
27:   formData.append("ipk", form.ipk);
28:   formData.append("bobot", form.bobot);
29:   formData.append("jumlah_sks_tempuh",
form.jumlah_sks_tempuh);
30:
31:   const res = await
fetch("http://localhost:5000/predict_manual", {
32:     method: "POST",
33:     body: formData,
34:   });
```

2. Upload File Excel

Pengguna dapat mengunggah file .xlsx atau .xls yang berisi data beberapa mahasiswa. File dikirim ke endpoint Flask (/predict_excel) dan hasil klasifikasi ditampilkan sebagai table HTML di halaman web.

```
40: const handleSubmitExcel = async () => {
41:   if (!file) {
42:     alert("Pilih file Excel terlebih dahulu.");
43:     return;
44:   }
45:
46:   const formData = new FormData();
47:   formData.append("file", file);
48:
49:   const res = await
fetch("http://localhost:5000/predict_excel", {
50:     method: "POST",
51:     body: formData,
52:   });
```

3. Tampilan Website

Merupakan *User Interface* sederhana dari *website* yang telah dibangun sebelumnya, meliputi laman *login*, laman klasifikasi dan laman visualisasi.

1) Halaman Login

Halaman login merupakan pintu masuk ke sistem *Early Warning System Kelulusan Mahasiswa*. Digunakan untuk memverifikasi siapa yang mengakses sistem, karena sistem hanya didesain dengan kebutuhan *single user* atau admin saja (untuk sementara ini).

Gambar 4.16 Halaman Login

Sumber : Dokumen Penulis

2) Laman Klasifikasi

Klasifikasi

Gambar 4.17 Halaman Klasifikasi manual

Sumber : Dokumen Penulis

Hasil Klasifikasi Excel										
nim	ipk	bobot	jumlah_sks_tempuh	semester	ip_smt_1	ip_smt_2	ip_smt_3	ip_smt_4	ip_smt_5	ip_smt_6
22000001	2.626667	200	127	6	3.16	3.52	2.83	3.50	2.95	3.14
22000002	3.713333	207	129	6	3.80	3.65	3.77	3.54	3.52	4.00
22000003	3.811667	245	128	6	3.90	4.00	3.36	3.80	3.90	3.91
22000004	3.096667	182	122	6	3.40	3.05	3.00	3.30	2.80	2.85
22000005	3.490000	215	126	6	3.15	3.72	3.15	3.81	3.71	3.22
22000006	3.518333	179	129	6	3.30	3.50	3.22	3.59	3.60	3.90
22000007	3.823333	227	127	6	3.55	3.54	3.31	3.63	3.71	4.00
22000008	3.683333	199	129	6	3.35	3.70	3.54	3.68	3.71	4.00
22000009	3.661667	231	128	6	3.85	3.90	3.30	3.80	3.80	3.52
22000010	3.190000	189	127	6	3.20	2.90	3.38	3.70	3.30	2.66
22000011	2.500000	180	149	6	2.71	2.84	2.73	2.83	1.80	2.83
22000012	1.528333	91	80	6	2.28	2.82	2.47	0.00	0.00	1.60
22000013	1.020000	61	65	2	3.04	3.08	0.00	0.00	0.00	0.00
22000014	2.515000	180	149	6	2.71	2.84	2.73	2.83	1.80	2.38
22000015	1.588000	127	113	6	2.98	1.08	0.96	0.20	1.45	3.26

Gambar 4.18 Halaman Klasifikasi excel
Sumber : Dokumen Penulis

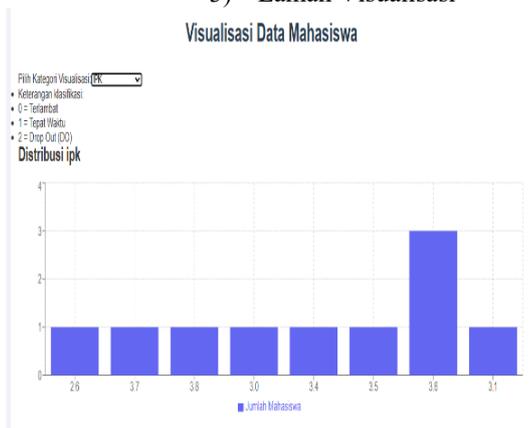
Form Manual

Fitur ini memungkinkan pengguna untuk mengisi data IPK, Bobot, dan Jumlah SKS Tempuh satu per satu, lalu mengirimkannya untuk diklasifikasikan.

Form Upload Excel

Fitur ini memfasilitasi pengunggahan file Excel yang berisi data mahasiswa dalam jumlah banyak. Sistem akan memproses semua data dan menampilkan hasil klasifikasi dalam bentuk tabel HTML. Seperti yang telah disinggung sebelumnya jika fitur memenuhi kriteria *rule* akademik maka data tidak akan diproses lebih lanjut oleh model.

3) Laman Visualisasi



Gambar 4.19 Halaman Visualisasi
Sumber : Dokumen Penulis

Fitur ini memfasilitasi visualisasi hasil klasifikasi excel. Sistem akan memproses semua data dan menampilkan hasil Visualisasi dalam bentuk *Bar Chart*. Terdapat *dropdown* untuk memilih kategori sesuai dengan format klasifikasi.

4.2.7 Rule Implementation

Pada tahap ini, dilakukan penerapan aturan berbasis logika (*rule-based*) sebagai bagian dari sistem klasifikasi

untuk memprediksi potensi kelulusan mahasiswa. *Rule* ini digunakan sebagai filter awal sebelum data diproses menggunakan model machine learning. Adapun aturan yang diterapkan mengacu pada pedoman akademik kampus, yaitu: mahasiswa dikategorikan berisiko *Drop Out*

Apabila tidak mencapai IPK minimal 2,00 dan jumlah SKS tertentu pada semester-semester kritis, yaitu 36 SKS pada akhir semester kedua, 72 SKS pada akhir semester keempat, dan 110 SKS pada akhir semester kesepuluh. Oleh karena itu, apabila data mahasiswa menunjukkan semester 2 dengan IPK di bawah 2,00 atau SKS di bawah 36, maka secara otomatis diklasifikasikan sebagai *Drop Out* tanpa melalui proses prediksi oleh model. Aturan serupa berlaku untuk semester 4 dan 10.

Selain itu, sebagai tindakan preventif, sistem juga menetapkan bahwa mahasiswa pada semester selain 2, 4, dan 10 yang memiliki IPK di bawah 2,00 akan langsung diklasifikasikan sebagai *Drop Out*, meskipun tidak memenuhi batas semester kritis. Implementasi ini bertujuan untuk meningkatkan akurasi sistem dengan mengutamakan keputusan untuk kasus-kasus yang sudah jelas bermasalah, serta mengurangi ketergantungan penuh pada model machine learning yang bersifat prediktif.

Jika data mahasiswa tidak termasuk dalam kriteria *rule-based*, maka proses klasifikasi akan dilanjutkan menggunakan model Random Forest yang telah dilatih sebelumnya.

Rule Implementation juga diharapkan dapat membuat model sesuai dengan konteks lingkungan kampus dan tidak murni pada algoritma random forest itu sendiri, hal ini penting untuk kontekstualisasi model dan relevansi model.

4.2.8 Testing

4.2.8.1 White Box

Pengujian yang dilakukan dengan melihat langsung struktur logika kode program. Pengujian ini dilakukan oleh pengembang untuk memastikan alur logika, percabangan, dan fungsi berjalan sesuai harapan.

5. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan mengenai *Early Warning System* kelulusan mahasiswa menggunakan algoritma Random Forest, dan berdasarkan rumusan masalah serta tujuan penelitian dapat disimpulkan beberapa hal sebagai berikut:

1. Sistem yang dibangun dapat mengklasifikasikan mahasiswa ke dalam tiga kategori, yaitu Tepat Waktu, Terlambat, dan DO berdasarkan data IPK, IP Semester 1 hingga 6, jumlah SKS yang ditempuh, semester, bobot dan status pembayaran.
2. Algoritma Random Forest menunjukkan performa yang baik dalam proses pelatihan model, dengan akurasi mencapai hasil akurasi

tinggi dan mampu menangani data dengan kompleksitas sedang.

3. *Website* yang telah dibuat memudahkan pengguna (admin/kampus) untuk melakukan klasifikasi baik secara manual maupun massal menggunakan file Excel, serta menyajikan hasil klasifikasi secara interaktif dan informatif.
4. *Website* telah mencakup Sebagian *rule* akademik yang telah diimplementasikan di bagian *backend* sebagai logika sebelum pemrosesan data oleh model.
5. Terdapat potensi *data leakage* di beberapa fitur latihan, antara lain IPK dan Semester.

6. SARAN

Berdasarkan hasil evaluasi terhadap model klasifikasi Random Forest, diperoleh akurasi yang sangat tinggi pada seluruh kelas kategori kelulusan mahasiswa, terutama dengan nilai akurasi total sebesar 99% dan nilai evaluasi yang sempurna (1.0000) pada kelas Tepat Waktu. Namun, tingginya performa ini justru mengindikasikan adanya potensi *data leakage*, khususnya pada fitur-fitur yang secara tidak langsung mengandung informasi target.

7. REFERENSI

Achmad, Y. F., & Yulfitri, A. (2020). PENGUJIAN SISTEM PENDUKUNG KEPU-TUSAN MENGGUNAKAN BLACK BOX TESTING STUDI KASUS E-WISUDAWAN DI INSTITUT SAINS DAN TEKNOLOGI AL-KAMAL. In *Jurnal Ilmu Komputer* (Vol. 5).

Ajay, P., Pranati, M., Ajay, M., Reena, P., & Balakrishna, T. (2020). Prediction of student performance using random forest classification technique. *International Research Journal of Engineering and Technology (IRJET)*, 7(8), [e-ISSN: 2395-0056].

Aggarwal, C. C. (2021). *Machine learning for text*. Springer.

Bartram, L., Correll, M., & Tory, M. (2021). *Untidy data: The unreasonable effectiveness of tables*. arXiv.

Dinata, R. K., & Hasdyna, N. (2020). *Machine Learning*. Unimal Press.

Esposito, M., Palma, L., Belli, A., Sabbatini, L., & Pierleoni, P. (2022). Recent advances in Internet of Things solutions for early warning systems: A review. *Sensors*, 22(6), 2124.

International Business Machines Corporation. *IBM SPSS Modeler CRISP-DM guide* (Versi 18, Rilis 3). IBM Corp.

Josso, P., Hall, A., Williams, C., Le Bas, T., Lusty, P., & Murton, B. (2023). Application of random-forest machine learning algorithm for mineral predictive mapping of Fe-Mn crusts in the World Ocean. *Ore Geology Reviews*, 162, 105671

Kementerian Kesehatan Republik Indonesia. (n.d.). *Apa itu sistem peringatan dini (Early Warning System)*.

Diakses pada [6 November 2024], dari

<https://pusatkrisis.kemkes.go.id/apa-itu-sistem-peringatan-dini-early-warning-system/>

Microsoft. (2024). *Microsoft Excel*. Microsoft.

Nusamandiri Repository. *Modul sistem basis data*.

Nusamandiri. Diakses dari

<https://repository.nusamandiri.ac.id/repo/files/227362/download/Modul-SistemBasis-Data.pdf>

Project Jupyter. (n.d.). *Project Jupyter*

Raharjo, B. (2021). *Pembelajaran mesin (Machine learning)*. Yayasan Prima Agus Teknik.

Saleh, J. S., Adrian, A. M., & Sanger, J. B. (2022).

Sistem klasifikasi kelulusan mahasiswa dengan algoritma Random Forest. Program Teknik Informatika, Fakultas Teknik, Universitas Katolik De La Salle Manado

Sie, J. B. L., Musdar, I. A., & Bahri, S. (2022). Pengujian white box testing terhadap website Room menggunakan teknik basis path. *Jurnal Kharisma Tech*, 17(2), 45–57.

<https://jurnal.kharisma.ac.id/kharimatech/>

Surya, N. D. (2021). Implementasi Naïve Bayes Classifier dan Confusion Matrix pada analisis Sentimen Berbasis Teks Pada Twitter. *Jurnal Sains Komputer dan Informatika*, 702.

STMIK Widya Cipta Dharma. (2023). *Pedoman akademik STMIK Widya Cipta Dharma 2023*.

Samarinda: STMIK Widya Cipta Dharma

Telkom University. (2023, November 3). *Next.js:*

Framework berbasis React yang SEO friendly. Telkom University Jakarta.

Universitas Medan Area. (2023, April 10). *Pengertian basis data*.

Universitas Siliwangi. (n.d.). *Bab 2: Landasan teori*.

Repositori Universitas Siliwangi. Diakses dari

<http://repositori.unsil.ac.id/233/6/bab%202.pdf>

Utomo, D. P. (2020). Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung. *Jurnal Media Informatika Budidarma*, 439.

Vercel. (2025). *Next.js: The React Framework*

Wicaksono, S. R. (Ed.). (2021). *Blackbox testing: Teori dan studi kasus*. CV. Seribu Bintang

Yuda, O. W., Tuti, D., Yee, L. S., & Susanti. (2022).

Penerapan data mining untuk klasifikasi kelulusan mahasiswa tepat waktu menggunakan metode Random Forest. *SATIN - Sains dan Teknologi Informasi*.



UCAPAN TERIMA KASIH
Terimakasih.