BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Penelitian mengenai prediksi nilai akademik siswa telah banyak dilakukan oleh para peneliti dengan menggunakan berbagai algoritma machine learning. Penelitian yang membandingkan kinerja algoritma Naive Bayes, Support Vector Machine (SVM), dan Random Forest dalam memprediksi ketidakhadiran di tempat kerja menunjukkan bahwa Random Forest memiliki kinerja yang lebih unggul dalam hal akurasi [4]

Penelitian yang dilakukan oleh beberapa peneliti membahas prediksi nilai akhir mahasiswa menggunakan metode K-Means Clustering dan Naïve Bayes Classifier. Hasil penelitian mereka menunjukkan bahwa kombinasi metode ini dapat membantu dalam memodelkan kecenderungan nilai akademik [7].

Penelitian yang membandingkan metode klasifikasi Naive Bayes dan K-Nearest Neighbor dalam memprediksi prestasi siswa menunjukkan bahwa metode K-Nearest Neighbor memiliki ketepatan yang lebih tinggi dalam konteks data yang digunakan [8].

Selain itu, penelitian mengenai performa seleksi atribut untuk menentukan potensi mahasiswa putus studi menunjukkan pentingnya seleksi fitur dalam meningkatkan performa model prediktif [6].

Berbagai penelitian tersebut menunjukkan bahwa pemanfaatan algoritma klasifikasi sangat potensial dalam dunia pendidikan, terutama untuk melakukan prediksi terhadap performa akademik siswa. Namun, keterbatasan dari beberapa algoritma, seperti sensitivitas terhadap fitur tidak relevan dan risiko overfitting, mendorong kebutuhan untuk mengeksplorasi algoritma yang lebih stabil dan akurat seperti Random Forest.

2.2 Landasan Teori

A. Prediksi Nilai Akhir Semester Siswa Menggunakan Algoritma Random Forest

Dalam penelitian ini, fokus utama adalah memprediksi nilai akhir semester siswa dengan memanfaatkan kemampuan algoritma Random Forest. Pembahasan ini akan menguraikan konsep dasar dari setiap elemen judul, teori-teori pendukung yang lebih luas, serta alasan mengapa Random Forest adalah pilihan yang relevan untuk tujuan prediksi ini.

B. Konsep Prediksi Nilai Akhir Semester Siswa

Prediksi merupakan proses memperkirakan atau memprakirakan suatu kejadian atau nilai di masa depan berdasarkan data dan informasi yang tersedia saat ini [9]. Prediksi melibatkan analisis pola dan tren dari data historis untuk membangun model yang dapat menggeneralisasi perilaku masa depan. Dalam ranah pendidikan, prediksi ini sangat penting untuk mengantisipasi kinerja akademik siswa.

Nilai akhir semester sendiri adalah capaian akademik kumulatif seorang siswa di akhir periode pembelajaran, yang merupakan hasil agregasi dari berbagai komponen penilaian seperti tugas, kuis, partisipasi, serta ujian tengah dan akhir semester [10]. Nilai ini berfungsi sebagai indikator kunci keberhasilan siswa dalam memahami materi pelajaran dan mencapai kompetensi yang ditetapkan.

Siswa adalah individu yang sedang menempuh pendidikan formal di suatu institusi pendidikan, seperti sekolah dasar, menengah, atau perguruan tinggi (Undang-Undang Republik Indonesia Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional).

Pentingnya prediksi nilai akhir semester siswa sangat krusial. Prediksi ini memberikan umpan balik proaktif bagi siswa, memungkinkan mereka mengidentifikasi kelemahan dan melakukan perbaikan dini [11]. Bagi pendidik dan institusi, prediksi membantu dalam identifikasi dini siswa berisiko dan siswa berprestasi, memungkinkan intervensi yang tepat waktu. Selain itu, hasil prediksi dapat menjadi dasar evaluasi efektivitas metode pengajaran atau kurikulum, memberikan wawasan tentang faktor-faktor yang memengaruhi kinerja akademik

[12]. Proses ini termasuk dalam lingkup prediksi akademik, sebuah cabang dari *educational data mining* (EDM) dan *learning analytics* (LA). EDM dan LA adalah bidang interdisipliner yang menggunakan *machine learning* dan statistika untuk mengeksplorasi data pendidikan, bertujuan untuk memahami proses belajar dan meningkatkan hasil pendidikan [13].

C. Algoritma Random Forest

Algoritma Random Forest adalah metode *ensemble learning* yang kuat, dikembangkan oleh Leo Breiman pada tahun 2001. Nama "Random Forest" secara harfiah berarti "Hutan Acak," yang merepresentasikan kumpulan besar (hutan) model prediksi individual yang disebut pohon keputusan (*decision trees*) yang dibangun secara acak. Algoritma ini tetap relevan dan banyak digunakan dalam berbagai aplikasi data *science* hingga saat ini karena akurasi serta kemampuannya menangani data kompleks [14].

Sebagai bagian dari *ensemble learning*, Random Forest menggabungkan beberapa model prediktif untuk menghasilkan satu model yang lebih kuat dan akurat [15]. Ide utamanya adalah bahwa "kebijaksanaan kerumunan" seringkali lebih baik daripada kebijaksanaan satu individu. Setiap pohon dalam hutan adalah pohon keputusan, yang merupakan model prediksi non-parametrik yang membagi ruang data menjadi wilayah-wilayah yang lebih kecil berdasarkan serangkaian aturan keputusan (Sarker et al., 2022). Proses pembentukan pohon melibatkan pemilihan fitur dan nilai *threshold* yang paling baik untuk memisahkan data ke dalam kelompok-kelompok yang lebih homogen (berdasarkan variabel target). Meskipun pohon keputusan tunggal cenderung rentan terhadap *overfitting*, Random Forest mengatasi masalah ini dengan membangun banyak pohon yang terdiversifikasi.

D. Cara Kerja Algoritma Random Forest

Algoritma *Random Forest* bekerja dengan menerapkan dua bentuk keacakan utama untuk membangun kumpulan pohon keputusan:

1. *Bootstrap Aggregating* (Bagging): Dari dataset pelatihan, Random Forest akan membuat beberapa subset data baru secara independen dan dengan penggantian (*with replacement*). Ini berarti beberapa data mungkin muncul berkali-kali dalam satu subset, sementara data lain mungkin tidak muncul sama sekali. Setiap subset ini akan digunakan untuk melatih satu pohon keputusan individual [9].
2. Pemilihan Fitur Acak pada Setiap Pemisahan Node: Ketika membangun setiap pohon keputusan individual, pada setiap node (titik keputusan) di mana pohon perlu memisahkan data, algoritma tidak mempertimbangkan semua fitur yang tersedia. Sebaliknya, hanya subset acak dari total fitur yang dipilih sebagai kandidat untuk pemisahan terbaik. Jumlah fitur yang dipertimbangkan biasanya akar kuadrat dari total jumlah fitur untuk masalah klasifikasi, atau sepertiga dari total fitur untuk masalah regresi [15]. Keacakan ini memastikan bahwa setiap pohon dalam hutan memiliki keragaman yang tinggi, mencegah mereka menjadi terlalu mirip satu sama lain dan mengurangi korelasi antar pohon.

Setelah sejumlah pohon keputusan dibangun dengan cara ini, hasil prediksi digabungkan: untuk masalah regresi (seperti prediksi nilai numerik), output akhirnya adalah rata-rata dari prediksi yang diberikan oleh setiap pohon individual dalam hutan. Untuk masalah klasifikasi (prediksi kategori), outputnya adalah kelas mayoritas (mode) yang dipilih oleh sebagian besar pohon individual dalam hutan.

E. Kelebihan dan Kekurangan Random Forest

Kelebihan Algoritma Random Forest meliputi: akurasi tinggi berkat kombinasi banyak, ketahanan terhadap *overfitting* melalui keacakan dalam sampling data dan fitur, bahkan tanpa perlu pemangkasan pohon [9]. Kemampuan

menangani data skala besar dan kompleks dengan banyak fitur , kemampuan menangani nilai hilang secara internal, tidak memerlukan skala data (normalisasi/standardisasi tidak mutlak diperlukan), dan menyediakan estimasi pentingnya fitur (*feature importance*) yang berguna untuk interpretasi dan seleksi fitur [16].

Namun, kekurangan Algoritma Random Forest adalah: komputasi yang intensif karena harus membangun dan mengelola banyak pohon, yang bisa memakan waktu dan sumber daya, terutama untuk *dataset* yang sangat besar [15]; sifatnya yang sering dianggap "kotak hitam" sehingga sulit untuk secara langsung menginterpretasikan bagaimana keputusan akhir dibuat dibandingkan dengan satu pohon keputusan yang visualisasinya lebih sederhana dan proses prediksi yang bisa lebih lambat dibandingkan model tunggal karena melibatkan agregasi hasil dari banyak pohon.

F. Teori Pendukung dan Pembahasan Lanjut

Untuk mendukung penerapan Random Forest dalam prediksi nilai akhir semester, penelitian ini juga mengacu pada beberapa teori dari bidang *machine learning*, statistika, dan *educational data mining*.

1. Teori *Machine Learning* (Pembelajaran Mesin): Prediksi nilai akhir semester adalah contoh klasik dari *supervised learning*, di mana model belajar dari pasangan input-output yang diketahui (data historis siswa dan nilai akhir semesternya) untuk memprediksi output pada data baru [12]. Karena nilai akhir semester adalah variabel kontinu (misalnya 0-100, atau skala 0-4), tugas ini termasuk dalam regresi. Model regresi bertujuan untuk memprediksi nilai numerik. Metrik evaluasi yang umum digunakan untuk mengevaluasi kinerja model regresi meliputi *Mean Absolute Error* (MAE), *Mean Squared Error* (MSE), dan *Root Mean Squared Error* (RMSE) [15].
2. Evaluasi Model dalam *Machine Learning*: Untuk memastikan validitas dan kemampuan generalisasi model prediksi, evaluasi yang sistematis sangat penting. *Dataset* biasanya dibagi menjadi data latih (*training data*) untuk membangun model dan data uji (*testing data*) untuk mengevaluasinya pada

data yang belum pernah dilihat sebelumnya [9]. Selain itu, teknik validasi silang (*cross-validation*), seperti *k-fold cross-validation*, sering digunakan untuk mendapatkan estimasi kinerja model yang lebih robust dan mengurangi bias dari pembagian data tunggal. Ini melibatkan pembagian data menjadi *k* lipatan, melatih model *k* kali dengan lipatan yang berbeda sebagai data uji [17].

3. Data Mining dalam Pendidikan (*Educational Data Mining* dan *Learning Analytics*): EDM dan LA bertujuan untuk menemukan pola tersembunyi dan pengetahuan yang berguna dari data pendidikan . Dalam konteks ini, pola yang ditemukan adalah hubungan antara variabel input (misalnya nilai UTS, nilai tugas, absensi) dan nilai akhir semester siswa. Aplikasi utama EDM dan LA adalah prediksi kinerja siswa, yang memungkinkan identifikasi dini siswa berisiko, rekomendasi personalisasi, deteksi kecurangan, dan analisis efektivitas kurikulum [16]. Penerapan *machine learning* seperti Random Forest dalam EDM dan LA menyediakan alat yang objektif dan berbasis data untuk membuat keputusan yang lebih baik dalam lingkungan pendidikan. Hal ini memungkinkan intervensi yang lebih tepat waktu dan terarah, yang pada akhirnya dapat meningkatkan hasil belajar siswa secara signifikan.

G. Peran Bahasa Pemrograman Python dalam Implementasi

Dalam pengolahan data dan implementasi algoritma *Random Forest* untuk prediksi nilai akhir semester, bahasa pemrograman Python memiliki peran yang sangat sentral dan krusial. Python dikenal luas sebagai bahasa pemrograman pilihan dalam bidang ilmu data (*data science*), *machine learning*, dan *artificial intelligence* karena kesederhanaan sintaksisnya, fleksibilitas, dan ekosistem pustaka yang sangat kaya dan mendukung [18].

Beberapa alasan utama mengapa Python menjadi pilihan ideal meliputi:

1. Pustaka *Machine Learning* yang Komprehensif: Python menyediakan pustaka *machine learning* yang matang dan efisien seperti Scikit-learn. Pustaka ini secara khusus menawarkan implementasi algoritma *Random Forest* yang sudah teroptimasi (*sklearn.ensemble.RandomForestRegressor* atau *sklearn.ensemble.RandomForestClassifier*), memudahkan peneliti untuk

membangun, melatih, dan mengevaluasi model tanpa perlu membangun algoritma dari awal [19].

2. Manajemen dan Pra-pemrosesan Data: Pustaka seperti Pandas dan NumPy sangat efektif untuk mengelola dan memanipulasi *dataset*. Pandas memfasilitasi operasi seperti membaca data dari berbagai format (CSV, Excel), membersihkan data, menangani nilai yang hilang, dan melakukan transformasi data yang diperlukan sebelum model dilatih. NumPy menyediakan dukungan untuk komputasi numerik berkinerja tinggi yang menjadi dasar operasi matematika dalam *machine learning* [20].
3. Visualisasi Data: Pustaka seperti Matplotlib dan Seaborn memungkinkan visualisasi data yang efektif, baik untuk eksplorasi awal data (melihat distribusi, korelasi antar fitur) maupun untuk mempresentasikan hasil prediksi dan evaluasi model. Visualisasi ini membantu dalam memahami karakteristik data dan kinerja model [21].
4. Komunitas dan Sumber Daya: Python memiliki komunitas *data science* yang sangat besar dan aktif, yang berarti banyak sumber daya, tutorial, dan dukungan tersedia secara online. Ini sangat membantu dalam proses pengembangan dan pemecahan masalah [22].

Dengan menggunakan Python, seluruh alur kerja proyek *machine learning*, mulai dari pengumpulan data, pra-pemrosesan, pembangunan model Random Forest, hingga evaluasi dan visualisasi hasil, dapat dilakukan *dalam* satu lingkungan yang terintegrasi dan efisien. Hal ini menjadikan Python sebagai *tools* yang tidak terpisahkan dalam implementasi prediksi nilai akhir semester menggunakan algoritma Random Forest.