BAB 1
PENDAHULUAN

1.1 Latar Belakang Masalah

Kemacetan *traffic* jaringan (*network congestion*) merupakan salah satu tantangan utama dalam sistem komunikasi data modern yang dapat menyebabkan penurunan kualitas layanan (*Quality of Service*/QoS). Permasalahan ini ditandai dengan *throughput* yang rendah, latensi tinggi, *jitter* yang tidak stabil, serta tingginya tingkat kehilangan paket (*packet loss*) [1]. Kondisi tersebut berdampak langsung pada performa aplikasi, menurunkan kepuasan pengguna, dan mengakibatkan inefisiensi pemanfaatan sumber daya jaringan.

Seiring dengan perkembangan teknologi serta pertumbuhan jumlah perangkat dan volume lalu lintas data yang semakin besar, kebutuhan akan solusi prediksi kemacetan jaringan yang akurat dan responsif menjadi semakin mendesak. Pendekatan tradisional seperti *buffer management* dan *congestion control* pada protokol TCP sering kali mengalami keterlambatan dalam menangani atau merespons dinamika lalu lintas yang berubah-ubah secara cepat, sehingga dapat menyebabkan tindakan mitigasi atau pencegahan yang dilakukan sering kali tidak optimal dan berpotensi menyebabkan gangguan layanan. Hal ini kemudian membuat *machine learning* dimanfaatkan untuk memperbaiki atau menjadi solusi permasalahan di atas [2, 3].

Dalam beberapa tahun terakhir, pemanfaatan teknologi kecerdasan buatan (*artificial intelligence*/AI) dan pembelajaran mesin (*machine learning*) mulai banyak dikembangkan dalam bidang manajemen jaringan. Hal ini karena algoritma *machine learning* mampu menganalisis pola lalu lintas jaringan dan memprediksi kemungkinan terjadinya kemacetan berdasarkan parameter-parameter jaringan yang dapat diukur secara langsung, seperti *throughput*, latensi, *jitter*, dan *packet loss* [4]. Dengan prediksi yang akurat, administrator jaringan dapat melakukan tindakan preventif secara proaktif untuk menjaga performa jaringan tetap optimal.

Salah satu algoritma yang banyak digunakan untuk tugas klasifikasi dalam prediksi kemacetan jaringan adalah *Naive Bayes Classifier*. Algoritma ini dikenal karena efisiensi komputasinya, kemudahan implementasi, serta performa yang kompetitif pada data berskala besar. Namun, beberapa penelitian juga menunjukkan bahwa performa *Naive Bayes* dapat dipengaruhi oleh korelasi antar fitur, sehingga

diperlukan tahap pra-proses data yang baik untuk meningkatkan akurasi model [4].

Selain *Naive Bayes*, algoritma lain seperti *Random Forest* juga banyak digunakan untuk prediksi kemacetan jaringan. *Random Forest* merupakan algoritma *ensemble* yang mampu menangani data dengan fitur yang kompleks dan menghasilkan akurasi yang tinggi. Studi komparatif menunjukkan bahwa *Random Forest* sering kali memberikan hasil yang lebih baik dibandingkan *Naive Bayes* dalam mendeteksi kemacetan jaringan [5], meskipun dengan kebutuhan sumber daya komputasi yang lebih besar. Penelitian terbaru juga menunjukkan bahwa kombinasi teknik *feature selection* dan *random forest* dapat meningkatkan akurasi prediksi kemacetan pada data lalu lintas jaringan nyata.

Salah satu pendekatan untuk meningkatkan performa pengendalian kemacetan dalam jaringan komputer nirkabel adalah melalui algoritma *machine learning*. Menurut studi oleh Al-Karaki et al. (2005), pendekatan berbasis pohon keputusan seperti *random forest* menunjukkan potensi yang signifikan dalam memperbaiki kontrol kemacetan dibandingkan metode konvensional [6]. Hal ini menunjukkan bahwa pemanfaatan *machine learning* dalam konteks jaringan dapat memberikan solusi yang lebih adaptif dan efisien terhadap dinamika lalu lintas jaringan. Pemilihan algoritma *random forest* dalam penelitian ini didasarkan pada performa konsistennya dalam berbagai studi terkini terkait prediksi *traffic* dan kemacetan jaringan. Pada penelitian yang dilakukan oleh Chen et al. (2024) menunjukkan bahwa *random forest* efektif dalam mengelola prediksi *traffic* seluler dengan akurasi tinggi untuk alokasi sumber daya jaringan [7]. Dalam konteks prediksi rute lalu lintas, menunjukkan bahwa *random forest* mengungguli algoritma lain dengan akurasi mencapai 92.1% [8]. Selain itu, penelitian oleh Labonne et al. (2020) juga menunjukkan bahwa *random forest* memiliki keunggulan dalam prediksi *bandwidth* jangka pendek dengan kesalahan prediksi yang lebih rendah dibandingkan model lainnya [9]. Penelitian lain yang dilakukan untuk menentukan apakah terjadinya *congestion* atau tidak berdasarkan identifikasi *packet loss* juga memberikan hasil yang konsisten dibandingkan dengan algoritma lain, hal ini memperkuat keputusan penggunaan *random forest* dalam penelitian ini sebagai algoritma utama untuk klasifikasi kemacetan *traffic* jaringan.

Selain itu, pentingnya pemilihan fitur yang relevan dan teknik *preprocessing* data dalam meningkatkan performa model untuk klasifikasi kemacetan juga ditekankan dalam beberapa penelitian. Penggunaan teknik seleksi fitur berbasis *mutual information* dan normalisasi data terbukti mampu meningkatkan akurasi model *machine learning* secara signifikan. Evaluasi model menggunakan beberapa

metrik seperti *accuracy*, *precision*, *recall*, dan *F1-score* juga penting untuk memastikan keandalan sistem klasifikasi dalam berbagai skenario jaringan [10, 11].

Berdasarkan uraian di atas, penelitian ini bertujuan untuk membangun dan membandingkan model klasifikasi kemacetan *traffic* jaringan menggunakan algoritma *Random Forest* menjadi dua skema yang berbeda. Skema pembagian data 80/20 digunakan dalam penelitian ini untuk memastikan bahwa model mendapatkan proporsi data pelatihan yang cukup, sekaligus mempertahankan porsi evaluasi yang representatif. Skema ini juga banyak digunakan dalam penelitian terdahulu seperti pada sistem prediksi rute kemacetan berbasis *random forest*, yang menunjukkan akurasi prediksi melebihi 90% dengan pembagian data 80/20 disertai *cross validation* [8]. Model akan dilatih menggunakan *dataset* lalu lintas jaringan yang dikumpulkan dari perangkat jaringan atau *dataset* publik, dengan mempertimbangkan beberapa parameter. *Random Forest* dipilih pada penelitian ini karena *Random Forest* lebih mampu untuk menangkap hubungan non-linear dan interaksi antar fitur melalui *decision tree*, selain itu *Random Forest* mampu meminimalisir *outlier* dengan menggunakan *bagging* dan *voting* dari banyaknya jumlah pohon yang ada. Pemilihan *random forest* didasarkan juga karena pada *random forest* memiliki kemampuan untuk melakukan evaluasi fitur parameter apa yang paling berpengaruh melalui *feature importance*, sedangkan algoritma lain seperti *Naive Bayes* tidak memiliki sistem bawaan untuk melakukan evaluasi kontribusi pada fitur. Menurut penelitian oleh Ermatita dan Hafyz [12], *random forest* memiliki beberapa keunggulan yakni mampu menangani *dataset* besar dan kompleks dengan baik, *robust* terhadap *overfitting* karena proses randomisasi data dan fitur dan cocok untuk klasifikasi maupun regresi serta dapat digunakan untuk mengukur pentingnya fitur dalam prediksi. Hasil implementasi pada penelitian yang mengklasifikasikan atau memprediksi kelulusan mahasiswa menunjukkan bahwa *random forest* mencapai akurasi sebesar 98.41%, dengan *precision* dan *recall* yang tinggi, menunjukkan efektivitasnya dalam memprediksi status kelulusan [12]. Sehingga hal ini memperkuat pemilihan *random forest* sebagai algoritma yang digunakan karena *random forest* adalah salah satu algoritma klasifikasi yang sangat kuat di mana *random forest* mampu menggabungkan kelebihan dari banyak pohon keputusan sekaligus sehingga mengatasi kekurangan masing-masing pohon tunggal, dengan teknik *bootstrap* dan pemilihan fitur acak, membuat *random forest*menghasilkan prediksi yang stabil. Dengan menggunakan *random forest* diharapkan penelitian ini dapat memberikan kontribusi dalam pengembangan sistem prediksi kemacetan jaringan yang lebih efektif dan efisien, serta menjadi

referensi bagi pengembangan manajemen jaringan berbasis *machine learning* di masa depan.

1.2 Rumusan Masalah

1. Bagaimana membangun model *machine learning* berbasis *Random Forest* untuk mengklasifikasikan *traffic* kemacetan jaringan berdasarkan parameter performa jaringan?
2. Berapa performa model *Random Forest* berdasarkan metrik evaluasi dalam mengklasifikasikan kemacetan *traffic* jaringan pada dua skema pembagian data 60/20/20 dan 80/20 setelah dilakukan kalibrasi dan *threshold tuning*?

1.3 Batasan Permasalahan

Dalam membantu penyusunan skripsi hingga terbentuknya sebuah skripsi, batasan permasalahan menjadi salah satu hal penting yang membantu dalam proses tersebut. Berikut adalah beberapa batasan masalah yang diharapkan dapat membantu penyusunan dan pengerjaan:

1. Variabel model hanya menggunakan 6 fitur utama (*latency*, *jitter*, *throughput*, *packet loss*, *bandwidth usage*) dan 4 fitur hasil rekayasa menggunakan *feature engineering* dan tidak mengeksplorasi parameter jaringan lain seperti protokol layer aplikasi atau geolokasi perangkat.
2. Pelatihan dan pengujian akan menggunakan *dataset* yang didapatkan dari *Kaggle*, dengan format dan parameter yang telah ditentukan (*latency*, *jitter*, *throughput*, *packet loss*).
3. Skema pembagian data yang digunakan dibatasi pada dua metode, yaitu, 60% data latih, 20% data validasi, 20% data uji dan 80% data latih, 20% data uji (tanpa data validasi terpisah).

1.4 Tujuan Penelitian

Adapun tujuan dari penelitian ini dilakukan dapat dirincikan sebagai berikut:

1. Membangun model klasifikasi berbasis algoritma *random forest* untuk mendeteksi kemacetan jaringan komputer, menggunakan parameter performa jaringan.

2. Mengukur performa model *random forest* berdasarkan metrik evaluasi dalam mengklasifikasikan kemacetan *traffic* pada dua skema pembagian data, yaitu 60/20/20 dan 80/20, setelah menerapkan teknik kalibrasi dan *threshold tuning*.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini dapat dibagi menjadi beberapa, yakni:

1. Menambah wawasan tentang bagaimana penerapan *machine learning* untuk analisis jaringan.
2. Dapat meningkatkan efisiensi pengelolaan atau manajemen jaringan dengan prediksi berbasis data.
3. Menjadi referensi bagi pengembangan sistem otomatis dalam melakukan *monitoring* performa jaringan kedepannya.
4. Dapat menjadi dasar untuk penelitian lebih lanjut menggunakan algoritma *machine learning* tingkat lanjut seperti *deep learning*.

1.6 Sistematika Penulisan

Sistematika penulisan laporan adalah sebagai berikut:

1. Bab 1 PENDAHULUAN

   Bab ini berisikan latar belakang masalah, rumusan masalah, batasan permasalahan, tujuan penelitian, manfaat penelitian, dan sistematika penelitian.

2. Bab 2 LANDASAN TEORI

   Bab ini berisikan landasan teori yang menjadi acuan selama penelitian ini berlangsung.

3. Bab 3 METODOLOGI PENELITIAN

   Bab ini berisikan tahapan-tahapan dari metodologi penelitian dan proses pembangunan model *machine learning*.

4. Bab 4 HASIL DAN DISKUSI

   Bab ini berisikan hasil implementasi dari model yang telah dibangun dan dilakukan *testing* atau pengujian.

5. Bab 5 SIMPULAN DAN SARAN

   Bab ini berisikan kesimpulan serta saran dari hasil penelitian yang telah dilakukan, yang diharapkan bisa dipakai dan dikembangkan di masa mendatang.