Contents lists available at JESA

Journal of Engineering and Science Application

journal homepage: https://jesa.aks.or.id/index.php/jesa/index

Klasifikasi Gangguan Jaringan Load Balancing Menggunakan Metode Naive Bayes

Sriyono Pambudi[a], Atang Susila[b]

[a]*Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang, Tangerang Selatan, Indonesia*

Article Info

Keywords:
Naive Bayes,
Load Balancing,
Machine Learning,
Klasifikasi,
Gangguan Jaringan,
TF-IDF.

ABSTRACT

Perkembangan teknologi informasi yang semakin pesat menyebabkan kebutuhan terhadap jaringan komputer yang stabil dan andal menjadi sangat penting dalam menunjang operasional perusahaan. Pada lingkungan jaringan yang menggunakan metode load balancing, gangguan jaringan seperti *Server Down*, *Packet Loss*, *Latency*, *Overload*, *Routing Loop*, dan *Delay* dapat mempengaruhi kestabilan distribusi trafik jaringan. PT. Indomarco Prismatama sebagai perusahaan yang menggunakan beberapa ISP dalam sistem *load balancing* masih melakukan pencatatan gangguan jaringan secara manual, sehingga proses identifikasi dan analisis gangguan menjadi kurang efektif. Penelitian ini bertujuan untuk menerapkan metode Naive Bayes dalam mengklasifikasikan gangguan jaringan load balancing berdasarkan data historis gangguan jaringan. Metode penelitian yang digunakan adalah metode eksperimen dengan pendekatan machine learning menggunakan algoritma Naive Bayes. Tahapan penelitian meliputi pengumpulan data, preprocessing data teks, transformasi data menggunakan TF-IDF, pembagian data training dan testing, proses klasifikasi, serta evaluasi model menggunakan Accuracy, Precision, Recall, dan F1-Score. Dataset yang digunakan terdiri dari data gangguan jaringan yang diklasifikasikan ke dalam beberapa kategori gangguan jaringan. Hasil penelitian menunjukkan bahwa metode Naive Bayes mampu mengklasifikasikan gangguan jaringan dengan baik. Berdasarkan hasil pengujian terhadap total data 950 dengan 750 data training dan 200 data testing, model berhasil memperoleh nilai *Accuracy* sebesar 90%, *Precision* sebesar 90%, Recall sebesar 90%, dan *F1-Score* sebesar 90%. Hasil tersebut menunjukkan bahwa metode Naive Bayes efektif digunakan dalam proses klasifikasi gangguan jaringan load balancing berbasis data teks. Sistem yang dibangun diharapkan dapat membantu teknisi jaringan dalam melakukan identifikasi gangguan secara lebih cepat, objektif, dan efisien dibandingkan metode manual.

This is an open access article under the CC BY-SA license.

Corresponding Author:
Sriyono Pambudi
Program Studi Teknik Informatika, Fakultas Ilmu Komputer
Universitas Pamulang
Tangerang Selatan, Indonesia
Email: pambudisriyono@gmail.com

1. INTRODUCTION

Perkembangan teknologi informasi membuat jaringan komputer menjadi bagian penting dalam mendukung kegiatan operasional perusahaan. Ketersediaan jaringan yang stabil diperlukan agar pertukaran data, akses aplikasi, dan layanan internal dapat berjalan secara berkelanjutan. Pada lingkungan yang menggunakan beberapa jalur Internet Service Provider (ISP), load balancing digunakan untuk membagi trafik sehingga beban

jaringan dapat didistribusikan melalui jalur yang tersedia. Namun, penerapan load balancing tidak menghilangkan kemungkinan terjadinya gangguan pada jaringan. Pada PT. Indomarco Prismatama, gangguan jaringan pada lingkungan load balancing dicatat berdasarkan hasil monitoring teknisi. Data historis tersebut memuat deskripsi kondisi jaringan dan informasi provider, tetapi proses identifikasi jenis gangguan masih banyak bergantung pada pencatatan dan analisis manual. Kondisi tersebut membuat proses penentuan kategori gangguan menjadi kurang efektif ketika jumlah data bertambah. Gangguan yang menjadi fokus penelitian meliputi Server Down, Packet Loss, Latency, Overload, Routing Loop, dan Delay. Permasalahan klasifikasi tersebut dapat dipandang sebagai masalah supervised learning karena data historis telah memiliki label gangguan. Machine learning dapat digunakan untuk mempelajari pola dari data tersebut sehingga deskripsi gangguan baru dapat diklasifikasikan ke kelas yang sesuai. Dalam penelitian ini, data berbentuk teks perlu melalui preprocessing sebelum digunakan sebagai masukan model. Tahapan tersebut meliputi pembersihan teks, tokenizing, stopword removal, dan stemming. Setelah itu, teks direpresentasikan menggunakan TF-IDF agar informasi penting pada setiap dokumen dapat digunakan sebagai fitur klasifikasi. Metode Naive Bayes dipilih karena merupakan metode klasifikasi probabilistik yang sederhana dan dapat diterapkan pada data teks. Penelitian dalam skripsi menunjukkan bahwa metode ini digunakan untuk menghitung probabilitas prior, likelihood, dan posterior, kemudian memilih kelas dengan probabilitas terbesar. Implementasi menggunakan Python memanfaatkan TfidfVectorizer dengan n-gram (1,2) dan MultinomialNB. Pendekatan ini memungkinkan model mengenali kata tunggal maupun kombinasi dua kata, seperti 'server' dan 'Server Down'. Penelitian terdahulu yang digunakan sebagai landasan dalam skripsi membahas penerapan machine learning, text mining, preprocessing, supervised learning, serta evaluasi model menggunakan Accuracy, Precision, Recall, F1-Score, dan Confusion Matrix [1]–[19]. Literatur tersebut menjadi dasar pemilihan tahapan pengolahan data dan evaluasi model dalam penelitian ini.

Berdasarkan permasalahan tersebut, penelitian ini bertujuan menerapkan Naive Bayes untuk mengklasifikasikan gangguan jaringan load balancing berdasarkan data historis PT. Indomarco Prismatama serta mengukur kinerjanya menggunakan Accuracy, Precision, Recall, dan F1-Score. Nilai tambah penelitian terletak pada penerapan rangkaian preprocessing teks dan TF-IDF terhadap data gangguan jaringan nyata yang berasal dari aktivitas monitoring beberapa ISP, kemudian menggunakannya sebagai dasar klasifikasi otomatis

2. RESEARCH METHOD

2.1 Research Design

Penelitian menggunakan metode eksperimen dengan pendekatan machine learning. Proses penelitian dilakukan secara berurutan mulai dari pengumpulan data gangguan jaringan, pemeriksaan dan preprocessing data teks, transformasi teks menjadi fitur numerik menggunakan TF-IDF, pembagian data training dan testing, penerapan Naive Bayes, pengujian, hingga evaluasi hasil klasifikasi. Alur tersebut disusun untuk memastikan model diuji pada data yang tidak digunakan pada saat pelatihan.

2.2 Data Acquisition

Data penelitian berupa data historis gangguan jaringan load balancing pada PT. Indomarco Prismatama yang diperoleh dari catatan teknisi berdasarkan monitoring beberapa ISP. Provider yang digunakan dalam dataset adalah Astinet, Fiberstar, dan Icon. Atribut utama terdiri atas deskripsi gangguan, provider, dan label. Label yang digunakan adalah Server Down, Packet Loss, Latency, Overload, Routing Loop, dan Delay.

Kelas Gangguan	Karakteristik Deskripsi dalam Data
Server Down	Server mati, tidak dapat diakses, atau gagal dihubungi
Packet Loss	Paket/data tidak sampai tujuan atau packet loss tinggi
Latency	Ping tinggi atau waktu respons jaringan tinggi
Overload	Bandwidth penuh, trafik tinggi, atau beban jaringan tinggi
Routing Loop	Perulangan jalur/routing pada jaringan
Delay	Koneksi lambat, delay, atau waktu respons lama

Dataset akhir yang digunakan dalam pengujian terdiri atas 950 data. Skripsi membagi data tersebut menjadi 750 data training dan 200 data testing. Data training digunakan untuk membangun model, sedangkan data testing digunakan untuk mengukur kemampuan model dalam mengklasifikasikan data yang belum digunakan saat pelatihan.

2.3 Text Preprocessing

Preprocessing dilakukan untuk menyiapkan deskripsi gangguan sebelum transformasi fitur. Proses yang dijelaskan dalam skripsi meliputi lowercasing, pembersihan karakter, tokenizing, stopword removal, dan stemming. Implementasi program menggunakan fungsi preprocessing berbasis Python untuk mengubah teks menjadi huruf kecil dan menghapus karakter selain huruf dan spasi. Tahapan konseptual preprocessing digunakan untuk mengurangi variasi penulisan dan menghasilkan teks yang lebih terstruktur [3], [5], [11], [17].

Tahap	Tujuan	Contoh
Lowercasing	Menyeragamkan huruf	Server Down → server down
Cleaning	Menghapus karakter yang tidak diperlukan	packet-loss! → packetloss
Tokenizing	Memecah teks menjadi token	server down → server | down
Stopword removal	Mengurangi kata yang kurang informatif	kata umum dihilangkan
Stemming	Mengubah kata ke bentuk dasar	mengakses → akses

2.4 TF-IDF Feature Representation

Setelah preprocessing, teks diubah menjadi representasi numerik menggunakan Term Frequency-Inverse Document Frequency (TF-IDF). TF menunjukkan frekuensi kemunculan suatu kata dalam dokumen, sedangkan IDF memberikan bobot lebih besar pada kata yang relatif jarang muncul di seluruh dokumen. Implementasi penelitian menggunakan TfidfVectorizer dengan parameter ngram_range=(1,2). Dengan demikian, model dapat mengenali unigram seperti 'server' dan bigram seperti 'Server Down'. Penggunaan representasi fitur ini mengikuti kebutuhan penelitian untuk mengubah deskripsi gangguan menjadi bentuk numerik yang dapat diproses oleh algoritma klasifikasi.

2.5 Naive Bayes Classification

Naive Bayes menentukan kelas berdasarkan probabilitas suatu kelas terhadap fitur yang terdapat pada data. Dalam penelitian ini, proses perhitungan dijelaskan melalui prior, likelihood, dan posterior. Probabilitas prior menggambarkan peluang awal setiap kelas. Likelihood menunjukkan peluang kemunculan fitur pada kelas tertentu, sedangkan posterior digunakan untuk menentukan kelas akhir. Secara umum, keputusan klasifikasi diberikan kepada kelas dengan probabilitas posterior terbesar.

Pada tahap implementasi, model yang digunakan adalah MultinomialNB dari scikit-learn. Model dilatih menggunakan fitur TF-IDF dan label gangguan. Data baru diproses dengan vectorizer yang sama, kemudian model menghasilkan prediksi kelas dan nilai confidence. Sistem juga menyediakan pemetaan solusi berdasarkan kelas gangguan, misalnya Server Down diarahkan pada pemeriksaan resource dan restart server, sedangkan Packet Loss diarahkan pada pemeriksaan kabel dan kualitas jaringan.

2.6 Evaluation

Evaluasi dilakukan menggunakan Accuracy, Precision, Recall, F1-Score, dan Confusion Matrix. Accuracy mengukur proporsi prediksi yang benar terhadap seluruh data uji. Precision menunjukkan ketepatan prediksi suatu kelas, Recall menunjukkan kemampuan model menemukan data yang benar pada kelas tersebut, sedangkan F1-Score menggabungkan Precision dan Recall. Confusion Matrix digunakan untuk melihat distribusi prediksi benar dan kesalahan pada setiap kelas [2], [4], [8], [9], [12].

Tahap	Input	Output
1. Pengumpulan data	Catatan gangguan teknisi	Dataset berlabel
2. Preprocessing	Deskripsi gangguan	Teks lebih terstruktur
3. TF-IDF	Teks hasil preprocessing	Vektor fitur
4. Split data	Dataset	Training dan testing
5. Training	Fitur training + label	Model Naive Bayes
6. Testing	Fitur data uji	Prediksi kelas
7. Evaluasi	Prediksi + label aktual	Metrik performa

Langkah terakhir adalah evaluasi dan interpretasi. Accuracy, Precision, Recall, dan F1-Score dihitung, sedangkan Confusion Matrix digunakan untuk mengetahui kelas yang paling mudah maupun paling sulit dibedakan. Hasil tersebut kemudian dianalisis untuk mengetahui efektivitas Naive Bayes pada karakteristik data gangguan jaringan yang digunakan. Data kemudian dibagi menjadi training dan testing. Model MultinomialNB dilatih menggunakan data training. Setelah proses training selesai, data testing diproses dengan pipeline yang sama dan diprediksi menggunakan model yang telah terbentuk. Hasil prediksi dibandingkan dengan label aktual untuk memperoleh jumlah prediksi benar dan salah. Prosedur penelitian dimulai dengan identifikasi permasalahan pencatatan dan pengelompokan gangguan jaringan. Selanjutnya data historis dikumpulkan dari catatan teknisi dan diperiksa agar atribut deskripsi, provider, dan label dapat digunakan sebagai dasar supervised learning. Setelah data siap, deskripsi gangguan diproses melalui preprocessing dan ditransformasikan menjadi fitur TF-IDF.

2.8 Research Procedure

Fokus Kajian	Kaitan dengan Penelitian
Machine learning	Dasar penggunaan model klasifikasi pada data berlabel [1], [7]
Text mining	Dasar pengolahan deskripsi gangguan berbasis teks [6], [14]

Fokus Kajian	Kaitan dengan Penelitian
Preprocessing	Membersihkan dan menyeragamkan data sebelum klasifikasi [3], [5], [11], [17]
Supervised learning	Pembelajaran dari data yang telah memiliki label [13], [16], [18]
Evaluasi model	Accuracy, Precision, Recall, F1-Score, dan Confusion Matrix [2], [4], [8], [9]

Evaluasi juga menjadi bagian penting karena keberhasilan model tidak cukup dinilai hanya dari satu ukuran. Accuracy memberikan gambaran umum ketepatan, sedangkan Precision, Recall, dan F1-Score memberikan informasi tambahan mengenai kualitas prediksi dan keseimbangan performa model [2], [4], [8], [9], [15]. Berdasarkan kajian tersebut, penelitian ini menggunakan keempat metrik tersebut serta Confusion Matrix untuk melihat kesalahan pada masing-masing kelas. Pada pengolahan data teks, penelitian terdahulu yang dibahas dalam skripsi menekankan pentingnya preprocessing sebelum data digunakan pada proses klasifikasi. Cleaning, tokenizing, stopword removal, dan stemming bertujuan mengurangi variasi data yang tidak diperlukan sehingga fitur yang digunakan model menjadi lebih terstruktur [3], [5], [11], [14], [17]. Landasan penelitian dalam skripsi memuat sejumlah kajian yang berkaitan dengan machine learning, text mining, preprocessing, supervised learning, serta evaluasi model. Kajian tersebut digunakan untuk memperkuat pemilihan tahapan pengolahan data dan metrik evaluasi pada penelitian ini. Literatur mengenai machine learning digunakan untuk menjelaskan bahwa data berlabel dapat dipelajari oleh model untuk menghasilkan klasifikasi [1], [7], [13], [18].

3. RESULTS AND DISCUSSION

3.1 Dataset and Data Distribution

Hasil pengumpulan data menunjukkan bahwa penelitian menggunakan data gangguan jaringan yang berasal dari lingkungan load balancing PT. Indomarco Prismatama. Dataset akhir berjumlah 950 data. Sebanyak 750 data digunakan sebagai training dan 200 data sebagai testing. Data testing menjadi dasar pengukuran performa model pada tahap akhir.

Jenis Data	Jumlah	Persentase
Training	750	79%
Testing	200	21%
Total	950	100%

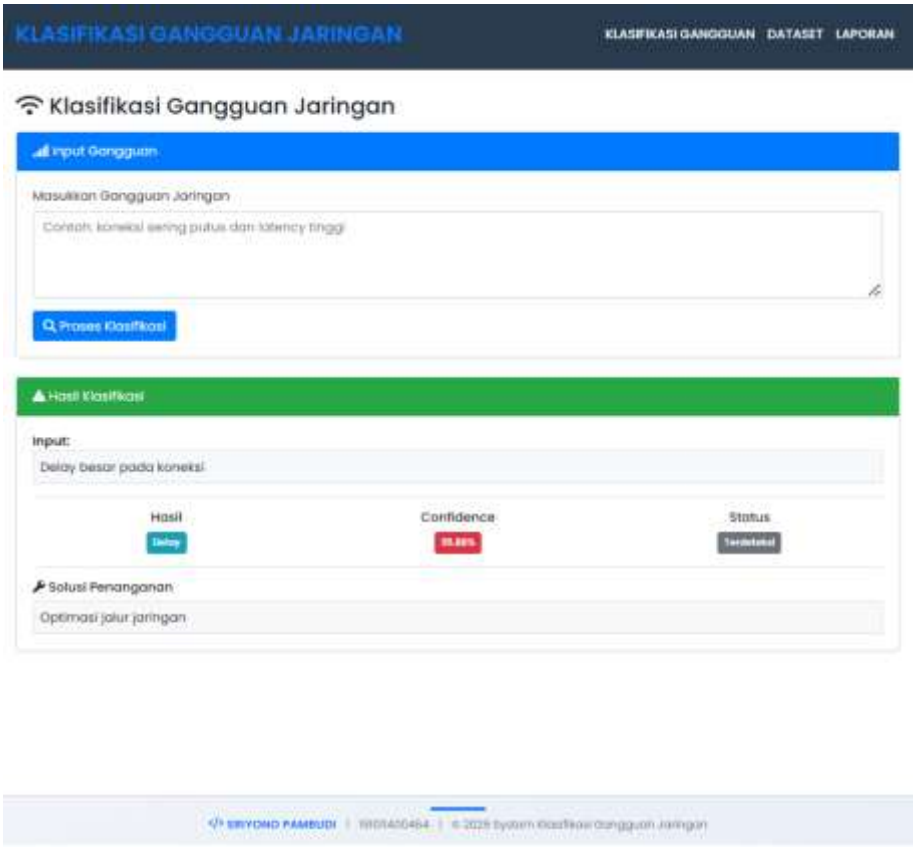
Pada data testing, model menghadapi enam kategori gangguan. Sampel yang ditampilkan dalam skripsi antara lain 'server gagal dihubungi' untuk Server Down, 'Packet Loss tinggi saat akses' untuk Packet Loss, 'ping tinggi pada jaringan' untuk Latency, 'bandwidth penuh saat sibuk' untuk Overload, 'Routing Loop pada jaringan' untuk Routing Loop, dan 'koneksi lambat sekali' untuk Delay. Contoh tersebut menunjukkan bahwa pola kata menjadi bagian penting dalam membedakan kelas.

3.2 Preprocessing and TF-IDF

Preprocessing menghasilkan teks yang lebih seragam sehingga variasi penulisan tidak terlalu mempengaruhi model. Deskripsi gangguan kemudian ditransformasikan menggunakan TF-IDF. Penggunaan n-gram (1,2) memungkinkan model mempertimbangkan kata tunggal dan pasangan kata. Hal ini relevan karena beberapa label memiliki istilah yang khas, misalnya 'Server Down' dan 'Routing Loop'. Sebaliknya, kelas Latency dan Delay memiliki beberapa kata yang berdekatan secara makna sehingga lebih berpotensi menghasilkan kesalahan klasifikasi.

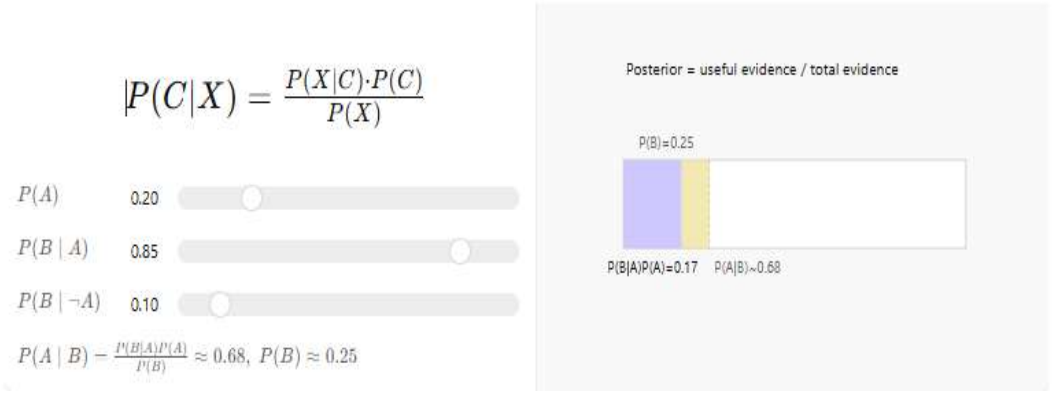
3.3 Penerapan Metode Naive Bayes

Setelah data melalui tahap preprocessing, langkah selanjutnya adalah menerapkan metode Naive Bayes untuk melakukan klasifikasi gangguan jaringan *load balancing*. Metode ini digunakan untuk menentukan kategori gangguan berdasarkan pola kata pada data historis yang telah dipelajari sebelumnya. Pada penelitian ini, model dilatih menggunakan data training yang telah diubah ke bentuk numerik menggunakan TF-IDF. Selanjutnya, model digunakan untuk memprediksi label gangguan pada data testing. Proses klasifikasi dilakukan dengan menghitung probabilitas setiap kelas gangguan, kemudian me milih kelas dengan nilai probabilitas tertinggi sebagai hasil prediksi. Kategori gangguan yang diprediksi meliputi *Server Down*, *Packet Loss*, *Latency*, *Overload*, *Routing Loop*, dan *Delay*.



3.1 Penerapan Metode Naive Bayes

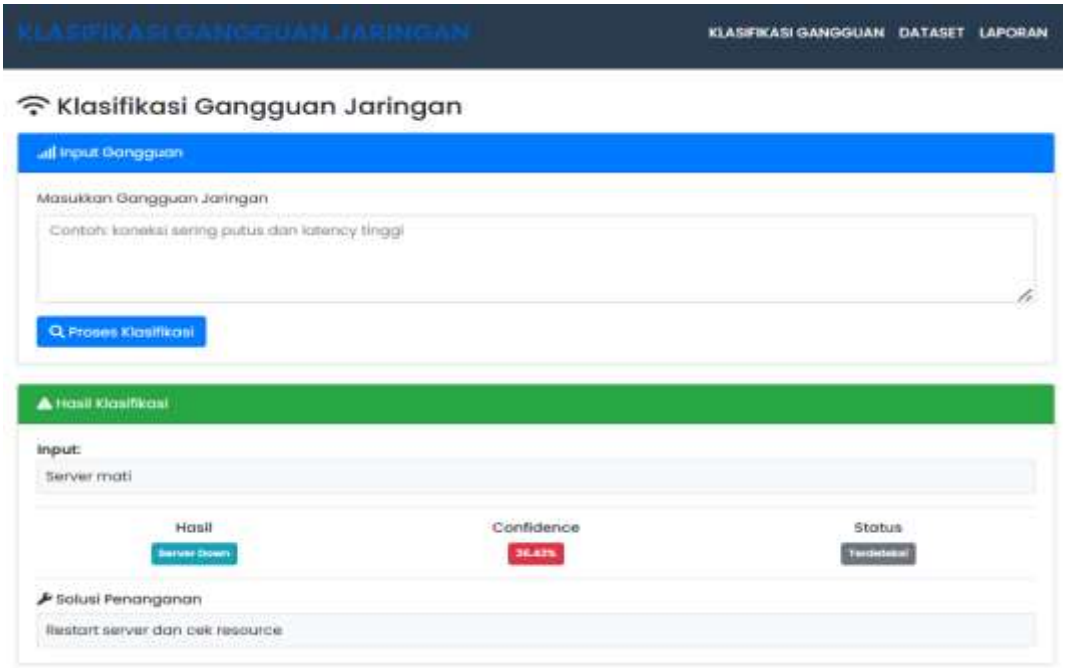
Metode Naive Bayes merupakan algoritma klasifikasi probabilistik yang dikembangkan berdasarkan teorema Bayes. Teorema ini digunakan untuk menghitung probabilitas bersyarat, yaitu peluang suatu sampel data *X* termasuk ke dalam kelas tertentu *C*. Pada penelitian ini, rumus Naive Bayes digunakan untuk menentukan jenis gangguan jaringan dengan probabilitas tertinggi



3.2 Rumus Metode Naive Bayes



3.3 User Mangisi Form Input



3.4 Menghitung Probabilitas dan Hasil Prediksi

3.4 Naive Bayes Training and Classification

Model MultinomialNB dilatih menggunakan hasil transformasi TF-IDF dari data training. Pada proses klasifikasi, data uji melewati preprocessing dan transformasi vectorizer yang sama, kemudian dimasukkan ke model. Model menentukan kelas berdasarkan probabilitas terbesar. Dalam contoh perhitungan manual pada skripsi, input 'Server Down' menghasilkan probabilitas posterior lebih tinggi pada kelas Server Down dibandingkan kelas lain. Contoh lain berupa 'koneksi lambat sekali' setelah preprocessing menjadi 'koneksi lambat' dan diklasifikasikan sebagai Latency dengan nilai probabilitas terbesar.

3.5 Prediction Results

No	Deskripsi Gangguan	Aktual	Prediksi	Status
1	server gagal dihubungi	Server Down	Server Down	Benar
2	Packet Loss tinggi saat akses	Packet Loss	Packet Loss	Benar
3	ping tinggi pada jaringan	Latency	Latency	Benar
4	bandwidth penuh saat sibuk	Overload	Overload	Benar
5	Routing Loop pada jaringan	Routing Loop	Routing Loop	Benar
6	koneksi lambat sekali	Delay	Latency	Salah
7	data tidak sampai tujuan	Packet Loss	Packet Loss	Benar
8	server mati mendadak	Server Down	Server Down	Benar
9	Delay saat browsing	Delay	Delay	Benar
10	traffic jaringan tinggi	Overload	Overload	Benar

Tabel tersebut memperlihatkan bahwa sebagian besar sampel dapat dikenali sesuai label aktual. Kesalahan terlihat pada deskripsi 'koneksi lambat sekali' yang sebenarnya berlabel Delay tetapi diprediksi sebagai Latency. Temuan ini sejalan dengan pembahasan skripsi bahwa Latency dan Delay memiliki kemiripan istilah seperti lambat, delay, respons lama, dan waktu respons sehingga menjadi pasangan kelas yang lebih sulit dibedakan.

3.6 Confusion Matrix

Aktual \ Prediksi	Server Down	Packet Loss	Latency	Overload	Routing Loop	Delay
Server Down	32	0	0	0	0	1
Packet Loss	0	30	2	0	0	1
Latency	0	1	29	0	0	3
Overload	0	0	1	31	0	1
Routing Loop	0	0	0	0	33	0
Delay	0	0	4	0	0	31

Confusion Matrix menunjukkan bahwa kelas Routing Loop memiliki pemisahan yang paling jelas karena seluruh data pada kelas tersebut diprediksi benar pada matriks yang disajikan. Server Down juga memiliki performa tinggi dengan 32 prediksi benar. Kesalahan terbesar terlihat pada hubungan Latency dan Delay. Sebanyak empat data Delay diprediksi sebagai Latency, sedangkan tiga data Latency diprediksi sebagai Delay. Pola ini memperkuat hasil analisis bahwa kedua kelas mempunyai karakteristik teks yang mirip.

3.7 Model Evaluation

Metrik	Hasil
Accuracy	90%
Precision	90%
Recall	90%
F1-Score	90%

Dari 200 data testing, sebanyak 180 data diprediksi dengan benar dan 20 data salah. Dengan demikian, Accuracy yang diperoleh adalah 180/200 atau 90%. Precision dan Recall masing-masing dilaporkan sebesar 90%, sedangkan F1-Score juga mencapai 90%. Nilai yang seimbang menunjukkan bahwa model tidak hanya memiliki ketepatan prediksi yang baik, tetapi juga mampu menemukan sebagian besar data yang sesuai dengan kelas aktual.

3.8 Comparison with Other Algorithms

Algoritma	Accuracy	Precision	Recall	F1-Score
Naive Bayes	90%	90%	90%	90%
Decision Tree	87%	86%	87%	86%
K-Nearest Neighbor	85%	84%	85%	84%
Random Forest	89%	88%	89%	88%
Support Vector Machine	88%	87%	88%	87%

Perbandingan pada skripsi menunjukkan Naive Bayes memperoleh nilai tertinggi pada empat metrik yang digunakan. Decision Tree, K-Nearest Neighbor, Random Forest, dan Support Vector Machine berada di bawah

hasil Naive Bayes pada pengujian yang dilaporkan. Berdasarkan hasil tersebut, Naive Bayes dinilai sesuai untuk karakteristik data teks gangguan jaringan pada penelitian ini.

3.9 Discussion

Secara keseluruhan, hasil penelitian menunjukkan bahwa Naive Bayes dapat diterapkan untuk klasifikasi gangguan jaringan load balancing berbasis deskripsi teks. Tahapan preprocessing dan TF-IDF membantu mengubah data mentah menjadi fitur yang lebih terstruktur sebelum proses klasifikasi. Model kemudian dapat memanfaatkan pola kata untuk membedakan kategori gangguan. Kelas Server Down dan Routing Loop relatif mudah dikenali karena mempunyai kata yang lebih spesifik. Pada Server Down, istilah seperti server mati, tidak bisa diakses, dan gagal dihubungi menjadi indikator yang kuat. Pada Routing Loop, istilah loop, routing berulang, dan jalur berputar juga lebih spesifik sehingga pemisahan kelas lebih jelas. Sebaliknya, Latency dan Delay masih memiliki tumpang tindih karakteristik. Kata lambat, delay, respons lama, dan waktu respons dapat muncul pada kedua kategori. Kondisi tersebut menjelaskan kesalahan yang terlihat pada confusion matrix. Dengan dataset yang lebih besar dan variasi deskripsi yang lebih beragam, model berpotensi memperoleh pola pembeda yang lebih kuat.

Naive Bayes juga memiliki kelebihan pada kesederhanaan perhitungan dan kemudahan implementasi. Model dapat dilatih menggunakan MultinomialNB dan dipadukan dengan TF-IDF tanpa membutuhkan proses komputasi yang kompleks. Namun, asumsi independensi antarfitur dapat menjadi keterbatasan ketika kata-kata dalam deskripsi gangguan memiliki hubungan yang kuat. Kualitas dataset dan preprocessing juga tetap berpengaruh terhadap hasil klasifikasi. Secara praktis, hasil penelitian dapat membantu teknisi dalam melakukan identifikasi awal gangguan secara lebih cepat dan objektif. Sistem dapat menerima deskripsi gangguan, melakukan preprocessing dan klasifikasi, kemudian menampilkan kelas, confidence, serta solusi yang dipetakan untuk kelas tersebut. Dengan demikian, pendekatan ini dapat menjadi media pendukung troubleshooting, bukan pengganti proses monitoring dan keputusan teknisi.

3.10 Research Contribution

Kontribusi penelitian mencakup penerapan Naive Bayes pada klasifikasi gangguan jaringan load balancing, integrasi preprocessing teks dan TF-IDF, serta penyediaan hasil klasifikasi yang dapat digunakan sebagai dasar identifikasi gangguan. Hasil Accuracy 90% menunjukkan bahwa pendekatan yang digunakan memberikan performa yang baik pada dataset historis yang tersedia. Penelitian juga memberikan gambaran bahwa karakteristik kata pada deskripsi gangguan berpengaruh terhadap kemampuan model membedakan kelas.

3.11 Error Analysis and Practical Interpretation

Analisis kesalahan menunjukkan bahwa tidak semua kesalahan model disebabkan oleh kelemahan algoritma. Sebagian kesalahan berasal dari karakteristik data teks yang mempunyai istilah berdekatan. Pada penelitian ini, hubungan paling jelas terdapat antara Latency dan Delay. Deskripsi seperti lambat, respons lama, dan delay dapat digunakan untuk menggambarkan kedua kondisi tersebut sehingga model memperoleh fitur yang mirip. Pada kelas Server Down, pola kata cenderung lebih spesifik. Istilah server mati, server gagal dihubungi, dan tidak bisa diakses mempunyai hubungan yang lebih langsung dengan label Server Down. Hal yang sama terlihat pada Routing Loop karena kata loop dan routing berulang memiliki karakteristik yang lebih khusus. Pola kata yang spesifik membantu Naive Bayes memberikan probabilitas yang lebih tinggi pada kelas yang sesuai. Packet Loss dan Overload juga menunjukkan karakteristik yang cukup jelas. Packet Loss berkaitan dengan data atau paket yang tidak sampai tujuan, sedangkan Overload berkaitan dengan bandwidth penuh dan trafik yang tinggi. Meskipun demikian, deskripsi yang terlalu singkat atau tidak konsisten tetap dapat menurunkan kemampuan model dalam membedakan kelas. Secara praktis, hasil klasifikasi sebaiknya digunakan sebagai pendukung keputusan awal. Sistem dapat memberikan indikasi jenis gangguan dan solusi yang telah dipetakan, tetapi teknisi tetap melakukan pemeriksaan jaringan untuk memastikan kondisi sebenarnya. Pendekatan ini sesuai dengan ruang lingkup skripsi yang menempatkan sistem sebagai media input dan penyajian hasil, bukan sebagai sistem monitoring jaringan real-time.

3.12 Mapping Classification to Troubleshooting

Hasil klasifikasi pada penelitian tidak berhenti pada label gangguan. Implementasi Python pada skripsi juga memiliki pemetaan solusi berdasarkan kelas yang diprediksi. Pemetaan tersebut memungkinkan keluaran model memberikan informasi tambahan yang dapat digunakan sebagai petunjuk awal troubleshooting.

Kelas	Solusi yang Dipetakan dalam Sistem
Overload	Tambah bandwidth atau optimasi load balancing
Packet Loss	Periksa kabel dan kualitas jaringan
Latency	Optimasi routing dan kurangi trafik
Routing Loop	Perbaiki konfigurasi routing

Kelas	Solusi yang Dipetakan dalam Sistem
Server Down	Restart server dan cek resource
Delay	Optimasi jalur jaringan

3.13 Robustness and Limitations

Nilai Accuracy 90% menunjukkan performa yang baik pada data testing yang digunakan, tetapi hasil tersebut tetap bergantung pada kualitas dan variasi dataset. Dataset yang lebih besar dan lebih beragam dapat memberikan pola bahasa yang lebih representatif. Hal ini penting karena model berbasis teks sangat dipengaruhi oleh cara teknisi menuliskan gejala gangguan. Penelitian juga memiliki keterbatasan sesuai batasan dalam skripsi. Metode yang digunakan hanya Naive Bayes dan tidak dilakukan modifikasi terhadap algoritma. Faktor eksternal seperti kondisi geografis, cuaca, kebijakan ISP, serta faktor fisik perangkat secara rinci tidak menjadi bagian dari klasifikasi. Sistem juga belum diarahkan untuk monitoring real-time. Keterbatasan lain adalah adanya kemiripan istilah antar kelas. Kesalahan pada Latency dan Delay menunjukkan bahwa model membutuhkan fitur yang lebih diskriminatif apabila penelitian dikembangkan lebih lanjut. Penambahan data, standardisasi penulisan deskripsi gangguan, dan pengujian metode pembanding dapat menjadi arah pengembangan yang relevan.

4. CONCLUSION

Penerapan metode Naive Bayes berhasil digunakan untuk mengklasifikasikan gangguan jaringan load balancing berdasarkan data historis PT. Indomarco Prismatama yang berasal dari beberapa provider, yaitu Astinet, Fiberstar, dan Icon. Proses klasifikasi dilakukan melalui preprocessing teks, transformasi TF-IDF, pelatihan MultinomialNB, dan pengujian terhadap data testing. Pada pengujian akhir dengan 950 data yang terdiri atas 750 data training dan 200 data testing, model memperoleh Accuracy sebesar 90%, Precision 90%, Recall 90%, dan F1-Score 90%. Sebanyak 180 dari 200 data testing berhasil diklasifikasikan dengan benar. Kesalahan klasifikasi terutama ditemukan pada kelas Latency dan Delay yang memiliki kemiripan karakteristik kata. Meskipun demikian, hasil keseluruhan menunjukkan bahwa Naive Bayes efektif digunakan sebagai pendekatan klasifikasi gangguan jaringan berbasis teks pada penelitian ini. Pengembangan selanjutnya dapat menggunakan dataset yang lebih besar dan membandingkan metode lain, serta mengembangkan sistem menuju integrasi monitoring real-time.

REFERENCES

[1] Andika, R., Fernando, L., and Y. P. Bagas, “Penerapan Algoritma Machine Learning untuk Klasifikasi Data pada Sistem Informasi,” Jurnal Ilmu Komputer dan Artificial Intelligence, vol. 12, no. 1, pp. 23–30, 2024.

[2] Hidayat, A., Riko, P., and Danu, S., “Implementasi Confusion Matrix dalam Evaluasi Model Klasifikasi,” Jurnal Teknologi Informasi, vol. 10, no. 1, pp. 60–68, 2022.

[3] Hidayat, Didik, B., and Satria, M., “Pengaruh Preprocessing pada Text Mining terhadap Akurasi Klasifikasi Data,” Jurnal Sistem Cerdas dan Teknologi Informasi, vol. 12, no. 1, pp. 45–53, 2024.

[4] Nugraha, A., Kevin, T., and Fajar, R., “Analisis F1-Score pada Model Machine Learning,” Jurnal Informatika dan Artificial Intelligence, vol. 11, no. 2, pp. 78–86, 2023.

[5] Prasetyo, A., Galih, N., and Rizky, A., “Implementasi Stemming dalam Pengolahan Data Teks untuk Meningkatkan Akurasi Klasifikasi,” Jurnal Sistem Cerdas dan AI, vol. 12, no. 1, pp. 40–48, 2024.

[6] Pratama, R., Aldo, K., and Dimas, H., “Penerapan Text Mining untuk Klasifikasi Data Teks Menggunakan Machine Learning,” Jurnal Informatika dan Komputer, vol. 10, no. 2, pp. 120–128, 2022.

[7] Putra, A., Vino, C., and Reza, F., “Penerapan Machine Learning untuk Klasifikasi Data pada Sistem Informasi,” Jurnal Teknologi Informasi, vol. 9, no. 2, pp. 101–108, 2022.

[8] Putri, D., Melani, S., and Rendra, W., “Analisis Accuracy pada Model Klasifikasi Data,” Jurnal Informatika, vol. 10, no. 2, pp. 70–78, 2022.

[9] Rahman, A., Bayu, T., and Ilham, K., “Evaluasi Model Machine Learning Menggunakan Berbagai Metrik Klasifikasi,” Jurnal Teknologi Informasi dan Sistem Cerdas, vol. 11, no. 1, pp. 50–58, 2023.

[10] Putra, D., Ardi, L., and Yoga, P., “Penerapan Machine Learning dalam Analisis Data dan Prediksi Sistem,” Jurnal Teknologi Informasi dan Komputer, vol. 10, no. 2, pp. 85–92, 2022.

[11] Rahmawati, D., Tegar, S., and Wira, H., “Analisis Preprocessing pada Text Mining untuk Klasifikasi Data Teks,” Jurnal Informatika dan Data Mining, vol. 10, no. 2, pp. 90–98, 2022.

[12] Ramadhani, F., Fikri, A., and Bagus, M., “Evaluasi Model Klasifikasi Menggunakan Recall dan F1-Score,” Jurnal Artificial Intelligence dan Machine Learning, vol. 12, no. 1, pp. 25–33, 2024.

[13] Saputra, F., Andra, K., and Hendra, P., “Analisis Kinerja Model Klasifikasi Menggunakan Metode Supervised Learning,” Jurnal Informatika Modern, vol. 11, no. 1, pp. 55–63, 2023.

[14] Saputri, D., Nabila, R., and Farhan, T., “Analisis Data Teks Menggunakan Metode Text Mining pada Sistem Informasi,” Jurnal Teknologi Informasi, vol. 11, no. 1, pp. 67–75, 2023.

[15] Sari, M., Joko, H., and Rian, P., “Pengukuran Precision dan Recall dalam Klasifikasi Data,” Jurnal Sistem Cerdas, vol. 11, no. 1, pp. 40–48, 2023.

[16] Sari, Dimas, A., and Lestari, D., “Implementasi Metode Machine Learning pada Klasifikasi Data Berbasis Supervised Learning,” Jurnal Informatika dan Sistem Cerdas, vol. 11, no. 1, pp. 45–53, 2023.

[17] Utami, S., Fadli, M., and Arman, K., “Pengaruh Stopword Removal terhadap Kinerja Klasifikasi Teks,” Jurnal Teknologi Informasi, vol. 11, no. 1, pp. 56–64, 2023.

[18] Wahyuni, S., Putu, R., and Mahendra, D., “Implementasi Supervised Learning dalam Klasifikasi Data Berbasis Machine Learning,” Jurnal Sistem Cerdas dan AI, vol. 12, no. 1, pp. 34–42, 2024.

[19] Sriyono Pambudi, “Klasifikasi Gangguan Jaringan Load Balancing Menggunakan Metode Naive Bayes,” Skripsi, Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Pamulang, Tangerang Selatan, 2026.