Analisis Komparasi Algoritma *Supervised Machine Learning* (*Random Forest* vs *Naive Bayes*) untuk Klasifikasi Kekuatan *Password* Berbasis Pola Karakter

Quinta Bilqis Kharisma
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
quintabilqis@student.telkomuniversity.ac.id

Farah Afianti
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
farahafi@telkomuniversity.ac.id

Abdullah Hanifan
Fakultas Informatika
Universitas Telkom
Bandung, Indonesia
abdullahhanifanah@telkomuniversity.ac.id

***Abstrak* — Di era digital saat ini, keamanan *password* masih menjadi masalah krusial, di mana berbagai insiden keamanan sering kali disebabkan oleh *password* yang lemah. Penelitian ini melakukan perbandingan antara algoritma *Random Forest* dan *Naive Bayes* dalam mengklasifikasikan kekuatan *password* berdasarkan pola karakter, dengan Shannon Entropy sebagai acuan dasar. Dataset yang digunakan terdiri dari 50.000 *password* yang diklasifikasikan ke dalam tiga kategori (*weak, medium, strong*), dengan ekstraksi 24 fitur yang mencakup aspek struktural, statistik, dan leksikal. Evaluasi dilakukan pada berbagai skenario pembagian data (80:20, 70:30, 50:50) menggunakan metrik akurasi, presisi, *recall, F1-score*, waktu inferensi, serta jejak memori. Hasilnya menunjukkan bahwa *Random Forest* mencapai akurasi sebesar 99,98% dengan waktu inferensi 0,0629 ms per prediksi, yang lebih unggul dibandingkan *Naive Bayes* (86,09%, 0,0011 ms) dan *Shannon Entropy* (82,73%). Uji *McNemar* mengonfirmasi bahwa perbedaan performa ini signifikan secara statistik ($p < 0,0001$). Analisis kesalahan mengungkapkan bahwa *Naive Bayes* mengalami penurunan performa hingga 21,24% pada *password* dengan pola kompleks karena keterbatasan asumsi independensi fitur. Fitur utama yang membedakan meliputi panjang *password* (32,06%), skor kompleksitas (22,37%), dan entropi (10,67%). Kedua model ini cocok untuk implementasi *real-time* dengan waktu inferensi di bawah 10 ms. *Random Forest* direkomendasikan untuk aplikasi keamanan kritis, sedangkan *Naive Bayes* lebih sesuai untuk sistem berskala besar dengan batasan latensi yang ketat.**

Kata kunci*— klasifikasi *password, machine learning, random forest, naive bayes*, keamanan siber, validasi *real-time

I. PENDAHULUAN

Transformasi digital telah mengubah hampir semua aspek kehidupan manusia menjadi bergantung pada akun digital, mulai dari layanan perbankan, pendidikan, kesehatan, hingga berbagai transaksi daring. Meskipun berbagai metode autentikasi terus dikembangkan, *password* masih menjadi mekanisme utama karena sifatnya yang sederhana, efisien dari segi biaya, dan mudah diimplementasikan. Namun, *password* sering kali menjadi titik lemah dalam keamanan sistem, bukan karena kekurangan teknologi, melainkan karena kecenderungan pengguna memilih *password* yang mudah diingat tetapi juga mudah ditebak.

Laporan Verizon Data Breach Investigations Report (2023) menunjukkan bahwa sebagian besar pelanggaran data terkait dengan kompromi kredensial pengguna. Penelitian Wang et al. (2021) menemukan bahwa pengguna biasanya membuat *password* berdasarkan pola yang dapat diprediksi, seperti *leet speak*, urutan keyboard, atau kata-kata dari kamus, yang memudahkan serangan *dictionary attack* dan *pattern-based attack* yang semakin maju.

Klasifikasi kekuatan *password* secara *real-time* menjadi hal yang sangat diperlukan dalam berbagai situasi, seperti validasi pendaftaran pengguna baru, *reset password*, autentikasi adaptif di lingkungan perusahaan, dan program edukasi keamanan. Klasifikasi dengan tiga tingkat (*weak, medium, strong*) menawarkan fleksibilitas kebijakan keamanan yang lebih baik dibandingkan klasifikasi biner sederhana, sehingga memungkinkan penerapan kontrol keamanan yang disesuaikan dengan tingkat risiko.

Penelitian ini bertujuan untuk menganalisis performa algoritma *supervised machine learning,* terutama *Random Forest* dan *Naive Bayes*, dalam mengklasifikasikan kekuatan password berdasarkan pola karakter. *Shannon Entropy* digunakan sebagai *baseline konvensional* untuk menunjukkan keterbatasan pendekatan *single-metric* saat menghadapi kompleksitas pola *password* masa kini. Kontribusi penelitian ini mencakup: (1) perbandingan menyeluruh antara algoritma dengan validasi statistik menggunakan uji *McNemar*, (2) analisis mendalam terhadap pentingnya fitur dan pola kesalahan pada berbagai pola leksikal, serta (3) evaluasi kelayakan implementasi *real-time* berdasarkan *trade-off* antara akurasi dan efisiensi komputasi.

II. KAJIAN TEORI

A. *Password Security* dan *Shannon Entropy*

Password adalah kredensial autentikasi berbasis pengetahuan yang keamanannya tergantung pada kerahasiaannya dan kekuatannya. *Shannon Entropy* mengukur ketidakpastian *password* melalui rumus

$$H(X) = -\Sigma p(i) \log_2 p(i) \quad (1)$$

di mana p(i) menunjukkan probabilitas kemunculan karakter ke-i. Meskipun efisien dari segi komputasi, *Shannon Entropy* memiliki beberapa keterbatasan, seperti tidak mampu mendeteksi pola struktural seperti *keyboard pattern* atau *leet speak*, bersifat *context-free*, serta menggunakan *threshold* yang arbitrer tanpa standar universal yang baku.

B. *Random Forest* dan *Naive Bayes*

Random Forest merupakan algoritma *ensemble learning* yang mengombinasikan beberapa *decision trees* melalui *bootstrap sampling* dan pemilihan fitur secara acak. Keunggulannya terletak pada kemampuan menangkap hubungan *non-linear*, ketahanan terhadap outliers, serta tidak memerlukan penskalaan fitur. Sementara itu, *Naive Bayes* adalah algoritma probabilistik yang didasarkan pada teorema *Bayes* dengan asumsi independensi kondisional antar fitur. Walaupun asumsi ini sering kali dilanggar dalam praktik, *Naive Bayes* tetap menunjukkan performa yang baik dengan efisiensi komputasi yang tinggi dan ukuran model yang minimal.

C. *Feature Engineering* untuk *Password*

Feature engineering bertujuan mengubah data mentah menjadi representasi numerik yang bermakna. Dalam penelitian ini, diekstraksi 24 fitur yang meliputi: (1) *Basic composition features* (panjang, jumlah huruf besar/kecil/angka/karakter khusus), (2) *Ratio features* (normalisasi terhadap panjang), (3) *Diversity & entropy features* (char_diversity, Shannon entropy), (4) *Pattern detection features* (keyboard pattern, kata umum, leet speak, digit berurutan), dan (5) *Composite features* (complexity_score, consonant_vowel_ratio).

III. METODE

A. Dataset dan *Preprocessing*

Dataset yang digunakan dalam penelitian ini mencakup 50.000 *password*, yang dikelompokkan ke dalam tiga kategori kekuatan *weak* (dengan 6.690 sampel atau sekitar 13,38%), medium (36.949 sampel atau 73,90%), dan *strong* (6.361 sampel atau 12,72%). Distribusi yang tidak merata ini sebenarnya merefleksikan situasi sehari-hari di mana orang-orang memilih *password*, yang sering kali tidak seimbang. Tahap *preprocessing* melibatkan pengambilan 24 fitur dari data tersebut, penerapan *stratified sampling* agar proporsi kelas tetap terjaga, serta normalisasi fitur khusus untuk model *Naive Bayes* dengan menggunakan *StandardScaler*.

B. Konfigurasi Model

Untuk model *Shannon Entropy*, menerapkan *threshold* sebagai berikut: jika entropi kurang dari 2.5 bits, maka diklasifikasikan sebagai *weak*; entropi antara 2.5 hingga kurang dari 3.5 bits masuk kategori medium; sedangkan entropi 3.5 bits atau lebih dikategorikan sebagai *strong*. Model *Naive Bayes* diasumsikan menggunakan distribusi Gaussian untuk menangani fitur-fitur kontinyu. Adapun Random Forest, diatur dengan parameter n_estimators=100, max_depth=20, dan min_samples_split=10, tujuannya untuk menghindari overfitting sekaligus memungkinkan model menangkap pola-pola kompleks dalam data.

C. Skenario Evaluasi

Evaluasi model dilakukan melalui tiga skenario pembagian data, yaitu 80:20 (sebagai skenario utama), 70:30, dan 50:50, dengan tujuan menguji ketahanan serta kemampuan generalisasi model di berbagai kondisi. Metrik yang digunakan mencakup *accuracy, precision, recall,* dan *F1-score* untuk mengukur performa klasifikasi, *inference time, throughput,* serta *memory footprint* guna menilai kesesuaian untuk aplikasi *real-time*. Selain itu, uji *McNemar* diterapkan untuk memverifikasi signifikansi statistik dari perbedaan performa antara model-model yang dibandingkan.

IV. HASIL DAN PEMBAHASAN

A. Performa Klasifikasi

Dari Tabel 1, terlihat bahwa pada skenario pembagian data 80:20, model *Random Forest* berhasil mencapai akurasi sebesar 99,98%, dengan gap generalisasi yang sangat kecil, yaitu hanya 0,02%. Menunjukkan bahwa model tersebut tidak mengalami *overfitting*, meskipun akurasi pada data training mencapai 100%. Sementara itu, *Naive Bayes* mencatat akurasi 86,09%, dengan gap yang sedikit negatif (-0,08%), yang mengindikasikan adanya *underfitting* ringan. Sebagai *baseline*, *Shannon Entropy* hanya mencapai 82,73%, yang berarti lebih rendah 17,25% dibandingkan *Random Forest.*

TABEL 1
Performa model pada skenario 80:20

Metrik	Random forest	Naive bayes	Shannon entropy
Test accuracy	99,98%	86,09%	82,73%
Precision	99,98%	91,35%	81,43%
Recall	99,98%	86,09%	82,73%
F1-score	99,98%	87,26%	81,92%
Inference time	0,0629 ms	0,0011 ms	-

Uji *McNemar* juga mengonfirmasi bahwa perbedaan performa antar model ini bersifat signifikan secara statistik, dengan nilai χ^2 sebesar 1.385,01 dan *p-value* kurang dari 0,0001. Rasio kesalahannya mencapai 1390:1, di mana *Random Forest* lebih sering benar dibandingkan *Naive Bayes* yang salah, dan sebaliknya. Dari segi *trade-off*, peningkatan akurasi sebesar 13,89% ternyata hanya memerlukan tambahan waktu inferensi sekitar 0,0618 ms, yang masih sangat dapat diterima untuk aplikasi di bidang keamanan.

B. Analisis Error pada Pola Leksikal

Tabel 2 memperlihatkan tingkat error pada berbagai pola leksikal. Ternyata, *Naive Bayes* mengalami penurunan performa paling besar, yaitu 21,24%, khususnya pada *password* yang "Tanpa Pola" seperti "xK9#mQ2". *Password* jenis ini memiliki struktur laten yang tidak dapat ditangkap oleh fitur eksplisit, meskipun tampak acak dengan pola alternasi huruf-angka-simbol yang sebenarnya deterministik. *Random Forest*, di sisi lain, mampu mendeteksi pola tersembunyi melalui aturan keputusan yang bersarang, sesuatu yang sulit dilakukan oleh *Naive Bayes* karena asumsi independensi fiturnya.

TABEL 2
Error rate pada pola leksikal

Pola	**Sampel**	**RF Error**	**NB Error**	**Selisih**
Leet speak	9.271	0,02%	13,34%	+13,32%
Keyboard pattern	134	0,00%	3,73%	+3,73%
Common words	29	0,00%	13,79%	+13,79%
Sequential digits	221	0,00%	3,17%	+3,17%
Tanpa pola	725	0,00%	21,24%	+21,24%

Pada *leet speak*, error meningkat 13,32%, di mana *Naive Bayes* kesulitan menangkap interaksi kontekstual, seperti substitusi '@' untuk 'a' dan '0' untuk 'o' yang sering muncul bersama dalam kata-kata tertentu. *Random Forest* berhasil mendeteksinya dengan aturan seperti: IF leet_speak_count > 1 AND has_common_word = True THEN *classify as weak*.

C. *Feature Importance* dan Redudansi

Berdasarkan analisis *Mean Decrease in Impurity,* tiga fitur utama yang dominan adalah *length* (32,06%), complexity_score (22,37%), dan entropy (10,67%). Panjang *password* menjadi indikator dasar karena secara eksponensial menentukan ukuran *keyspace*. Complexity_score, di lain pihak, menangkap interaksi kompleks antar fitur yang tidak bisa dijelaskan oleh fitur tunggal. Dalam eksperimen pengurangan fitur dari 24 menjadi 21, akurasi *Random Forest* hanya turun sedikit (0,02%), sedangkan untuk *Naive Bayes* penurunannya lebih besar (0,22%). Hal ini menunjukkan bahwa redundansi positif justru memberikan stabilitas dalam prediksi.

D. Kelayakan Implementasi *Real-Time*

Seperti yang ditampilkan di Tabel 3, kedua model memenuhi ambang batas latensi yang dapat diterima, yaitu di bawah 10 ms. *Random Forest*, dengan waktu inferensi 0,0629 ms dan throughput 15.902 pred/s, tepat untuk aplikasi keamanan kritis seperti perbankan yang memprioritaskan nol *false negatives*. Adapun *Naive Bayes*, dengan waktu inferensi 0,0011 ms dan throughput 920.810 pred/s, lebih sesuai untuk sistem berskala besar yang memiliki batasan latensi ekstrem atau *deployment* pada *embedded systems* dengan *overhead* memori yang minimal.

TABEL 3
Kelayakan Implementasi *Real-Time*

Metrik	Random forest	Naive bayes	Threshold
Inference time	0,0629 ms	0,0011 ms	< 10ms
P99Latency	0,0700 ms	0,0021 ms	< 100ms
Throughput	15.902 pred/s	920.810 pred/s	-
Model Size	0,79 MB	0,002 MB	-
Weak Detection (Recall)	100%	97,28%	>95%

V. KESIMPULAN

Dalam penelitian ini, menganalisis performa algoritma *Random Forest* dan *Naive Bayes* untuk mengklasifikasikan kekuatan *password* berdasarkan pola karakter. Hasilnya, *Random Forest* menunjukkan keunggulan yang jelas dengan akurasi 99,98%, jauh melampaui *Naive Bayes* (86,09%) dan *Shannon Entropy* (82,73%). Uji *McNemar* mengonfirmasi bahwa perbedaan ini signifikan secara statistik ($p < 0,0001$), dengan rasio kesalahan 1390:1. Analisis error mengungkapkan bahwa *Naive Bayes* mengalami penurunan performa hingga 21,24% pada *password* tanpa pola eksplisit, disebabkan oleh keterbatasan asumsi independensi fitur dalam menangkap struktur laten dan interaksi kompleks antar fitur.

Analisis *feature importance* menunjukkan bahwa *length* (32,06%), complexity_score (22,37%), dan entropy (10,67%) adalah fitur paling dominan. Eksperimen pengurangan fitur membuktikan bahwa redundansi positif memberikan stabilitas prediksi, dengan penurunan akurasi minimal 0,02% untuk *Random Forest* dan 0,22% untuk *Naive Bayes.* Kedua model juga memenuhi syarat *real-time* dengan waktu inferensi di bawah 10 ms. Kami merekomendasikan *Random Forest* untuk aplikasi keamanan kritis yang menuntut nol false negatives, sementara *Naive Bayes* lebih cocok untuk sistem berskala besar atau perangkat *embedded* dengan batasan memori ekstrem.

Untuk penelitian selanjutnya, dapat dipertimbangkan eksplorasi transformer embeddings guna menangkap kesamaan semantik dengan *password* yang telah bocor, implementasi fitur temporal untuk mendeteksi penggunaan ulang *password*, serta integrasi biometrik perilaku seperti dinamika pengetikan. Evaluasi *deployment* di lingkungan produksi dengan inferensi terdistribusi dan kompresi model juga diperlukan untuk memvalidasi kelayakan praktis pada skala *enterprise.*

REFERENSI

[1] P. A. Grassi, M. E. Garcia, and J. L. Fenton, "Digital Identity Guidelines: Authentication and Lifecycle Management," NIST Special Publication 800-63B, National Institute of Standards and Technology, June 2020.

[2] J. Bonneau, C. Herley, P. C. van Oorschot, and F. Stajano, "The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes," in 2012 IEEE Symposium on Security and Privacy, San Francisco, CA, USA, 2012, pp. 553-567.

[3] Verizon, "2023 Data Breach Investigations Report," Verizon Enterprise Solutions, 2023. [Online]. Available: https://www.verizon.com/business/resources/reports/dbir/

[4] D. Wang, Z. Zhang, P. Wang, J. Yan, and X. Huang, "Targeted Online Password Guessing: An Underestimated Threat," in Proceedings of the 2016 ACM SIGSAC

Conference on Computer and Communications Security, Vienna, Austria, 2016, pp. 1242-1254.
[5] M. Dell'Amico, P. Michiardi, and Y. Roudier, "Password Strength: An Empirical Analysis," in IEEE INFOCOM 2010, San Diego, CA, USA, 2010, pp. 1-9.
[6] T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd ed., New York: Springer, 2020.
[7] P7. Jha, H. Hamid, O. Olukola, and N. Rahimi, "Adversarial Machine Learning for Robust Password Strength Estimation," IEEE Access, vol. 13, pp. 12450-12465, Jan. 2025.
[8] Y. Zhang, Z. Hou, Y. Zou, Z. Li, and D. Wang, "EditPSM: A New Password Strength Meter Based on Password Reuse via Deep Learning," IEEE Transactions on Dependable and Secure Computing, vol. 22, no. 2, pp. 456-470, Feb. 2025.
[9] K. Patel, M. Shah, R. Gupta, and S. Mehta, "Password Strength Classification Using Machine Learning Methods," in Proc. 2024 Int. Conf. Intell. Comput. Commun. (ICICC), Mumbai, India, Dec. 2024, pp. 112-117.
[10] W. Chen, Y. Liu, and H. Wang, "Machine Learning Approaches for Password Strength Evaluation: A Comparative Study," ACM Computing Surveys, vol. 56, no. 3, Art. 45, pp. 1-35, Mar. 2023.
[11] C. M. Bishop, Pattern Recognition and Machine Learning, New York: Springer, 2020.