JERIN: Jurnal Riset Elektro dan Informatika
Vol. 1, No. 1, Mei 2025, pp. 26-32

Deteksi Pesan Phishing pada Media Sosial Menggunakan Metode Naïve Bayes

Nurvania Syakir[1], Syarah Yanti[2*], Nur Azizah[3]
[123]Program Studi Teknik Informatika, Universitas Muslim Indonesia, Kota Makassar

***Email Korespondensi:** 13020230038@student.umi.ac.id

Riwayat Artikel: Diterima: 11/02/2025; Direvisi: 19/04/2025; Disetujui: 02/05/2025

ABSTRAK

Penelitian ini membahas deteksi pesan *phishing* pada media sosial menggunakan metode Multinomial Naïve Bayes berbasis klasifikasi teks. *Dataset* yang digunakan adalah *SMS Spam Collection* yang terdiri dari 5.572 pesan dengan dua kelas, yaitu *ham* dan *spam/phishing*. Tahapan penelitian meliputi *preprocessing* teks (*lowercase*, penghapusan tanda baca, *stopword removal*, dan *stemming*), vektorisasi menggunakan *Bag-of-Words,* serta pembagian data latih dan uji dengan perbandingan 80:20. Evaluasi Kinerja model dilakukan menggunakan metrik *accuracy*, *precision*, *recall*, dan *F1-score*. Hasil penelitian menunjukkan bahwa model Naïve Bayes mampu mencapai *accuracy* sebesar 98,65%, dengan *recall* kelas *spam* sebesar 0,93 yang menandakan kemampuan tinggi dalam mendeteksi pesan *phishing* meskipun *dataset* tidak seimbang. Temuan ini membuktikan bahwa metode Naïve Bayes efektif, efisien, dan dapat diadaptasi untuk sistem deteksi *phishing* otomatis pada media sosial berbasis teks pendek.

Kata Kunci: klasifikasi teks; machine learning; media sosial; naïve bayes; phishing

1. PENDAHULUAN

Phishing merupakan salah satu bentuk kejahatan siber yang bertujuan mengumpulkan informasi pribadi pengguna seperti nama pengguna, kata sandi, dan data keuangan melalui manipulasi psikologis dan pesan palsu yang menyerupai sumber resmi [1], [2]. Aktivitas ini umumnya dilakukan lewat *email*, pesan teks, atau media sosial yang berisi tautan berbahaya dan memancing korban untuk memberikan data sensitif [2], [3]. *Phishing* termasuk tindak penipuan daring (*online fraud*) yang memanfaatkan kepercayaan pengguna terhadap platform digital [4]. Media sosial seperti Facebook, Instagram, dan Twitter menjadi sasaran utama karena tingkat interaksi pengguna yang tinggi serta kemampuan penyebaran pesan yang cepat [1], [4]. Pelaku sering mengirim pesan menyerupai komunikasi resmi sehingga sulit dibedakan dari pesan asli [1]. Kondisi ini menimbulkan kebutuhan akan sistem pendeteksi otomatis yang mampu mengenali pesan *phishing* secara cepat, akurat, dan efisien guna melindungi pengguna dari ancaman siber.

Pendekatan berbasis *machine learning* banyak digunakan untuk mendeteksi pesan *phishing* melalui klasifikasi teks [2], [3], [5]. Salah satu algoritma yang sering diterapkan adalah Naïve Bayes, karena bekerja berdasarkan prinsip probabilistik dengan menghitung distribusi kemunculan kata dalam pesan [1], [5]. Algoritma ini menentukan kemungkinan suatu pesan termasuk kategori *phishing* atau bukan berdasarkan hubungan antar fitur dalam teks [6]. Keunggulan Naïve Bayes terletak pada kesederhanaan matematis, waktu pelatihan cepat, serta performa stabil meskipun digunakan pada *dataset* besar [5], [7]. Berbagai penelitian membuktikan bahwa algoritma ini mampu mendeteksi *phishing* dengan akurasi tinggi. Fadlilah dan Fahmi [1] menunjukkan bahwa penerapan Naïve Bayes pada postingan Facebook menghasilkan akurasi hingga 99,01%. Pasha dan Azis [4] membuktikan efektivitasnya dalam mengidentifikasi penipuan daring di media sosial, sedangkan Tham et al. [2] dan Patel et al. [3] menegaskan kemampuan metode ini dalam mengenali pola bahasa dan tautan berbahaya pada pesan *phishing*. Selain itu, Haq et al. [8] menunjukkan bahwa algoritma Naïve Bayes efektif dalam mengklasifikasikan konten tidak valid di media sosial meskipun menggunakan bahasa yang informal. Penelitian oleh Lusi [9] membuktikan bahwa algoritma Naïve Bayes efektif dalam mengklasifikasi komentar *spam* berbahasa Indonesia di media sosial. Hasil ini menunjukkan relevansi metode tersebut untuk konteks lokal.

Berbagai studi menyatakan bahwa metode Naïve Bayes tetap relevan dalam keamanan informasi modern karena efisien, mudah diimplementasikan, dan mampu memberikan hasil klasifikasi cepat. Johnson et al. [5] menunjukkan kinerja tinggi dengan waktu komputasi rendah pada sistem deteksi *phishing* berbasis *email*, sedangkan Safi et al. [6] menekankan efektivitas algoritma ini karena strukturnya yang ringan dan adaptif terhadap berbagai format data. Ige [7] menyebutkan bahwa meskipun terdapat algoritma modern seperti

Support Vector Machine dan *Deep Learning*, Naïve Bayes tetap unggul karena kesederhanaannya dan kemampuan beroperasi secara *real-time*. Namun, penelitian sebelumnya masih berfokus pada teks umum dan belum memperhatikan karakteristik linguistik khas media sosial yang informal, banyak singkatan, serta mengandung emotikon dan tautan tersamar. Selain itu, sebagian besar *dataset* yang digunakan berbahasa Inggris, sehingga efektivitas Naïve Bayes untuk mendeteksi *phishing* berbahasa Indonesia belum diuji secara memadai. Evaluasi kinerja juga sering hanya menekankan akurasi tanpa mempertimbangkan aspek deteksi waktu nyata. Oleh karena itu, penelitian lebih lanjut diperlukan untuk menerapkan dan mengadaptasi metode Naïve Bayes pada media sosial berbahasa Indonesia secara komprehensif, guna mendukung deteksi cepat dan meningkatkan keamanan data di era digital. Temuan ini juga konsisten dengan analisis sistematis yang dilakukan oleh Safi dan Singh [10], yang mengungkapkan bahwa metode berbasis pembelajaran mesin, seperti Naïve Bayes, tetap menjadi teknik utama dan paling efisien dalam mengidentifikasi serangan *phishing* dengan tingkat akurasi yang melebihi 99%.

2. METODOLOGI PENELITIAN

Bagian Metodologi Penelitian menjelaskan secara tahapan, rancangan dan pendekatan yang akan digunakan untuk mencapai tujuan penelitian.

2.1. Dataset

Dataset yang digunakan dalam penelitian ini adalah SMS *Spam Collection Dataset* yang diperoleh dari platform Kaggle. *Dataset* ini merupakan kumpulan data publik berupa pesan SMS yang dikumpulkan untuk tugas klasifikasi *spam/phishing*. Penggunaan *dataset* ini sangat relevan untuk mendeteksi ancaman pada media sosial karena pola teks yang serupa, seperti penggunaan tautan berbahaya, penawaran palsu, dan teknik manipulasi psikologis [1], [2].

Dataset berformat CSV (spam.csv) berisi 5.572 instance pesan teks dalam bahasa Inggris dengan distribusi kelas yang tidak seimbang (*imbalanced*), yaitu 4.825 pesan *ham* (normal) dan 747 pesan *spam* [3]. Proses akuisisi dilakukan di lingkungan Google Colab menggunakan library Pandas untuk memuat data dan Matplotlib untuk analisis distribusi kelas guna memastikan model dapat menangani ketidakseimbangan data secara efektif [4], [10]. Berikut ini adalah link akses Dataset https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset

2.2. Preprocessing Data

Tahap *preprocessing data* dilakukan untuk membersihkan dan mempersipakan teks mentah agar dapat digunakan sebagai input model klasifikasi, sehingga meningkatkan akurasi dan mengurangi noise [1], [5]. Langkah-langkah utama meliputi:

a. Pembersihan Karakter (*Data Cleaning*): Menghapus karakter non-alfabet dan angka menggunakan *regular expression* re.sub. Hal ini bertujuan agar model fokus hanya pada pola kata-kata.
b. *Case Folding*: Mengonversi seluruh teks ke huruf kecil (*lowercase*) menggunakan fungsi Lower() untuk memastikan konsistensi, sehingga kata seperti "FREE" dan "free" dianggap sebagai entitas yang sama.
c. Tokenisasi: Memecah kalimat menjadi kumpulan kata individu (*words*) menggunakan fungsi. Split() untuk mempermudah proses penyaringan kata.
d. Penghapusan *Stopwords*: Menghapus kata-kata umum yang sering muncul namun tidak memiliki makna signifikan dalam klasifikasi (seperti *"is", "the", "to*") menggunakan daftar kata dari library nltk.corpus.stopwords.
e. *Stemming*: Mengubah kata ke bentuk dasarnya menggunakan algoritma PorterStemmer dari library NLTK (contoh: kata *"running*" menjadi "*run*"). Hal ini dilakukan untuk mengurangi dimensi fitur (jumlah kata unik) dalam dataset.
f. *Label Encoding*: Melakukan transformasi label kategori menjadi numerik, di mana label *'ham*' dipetakan menjadi **0** dan *'spam'* menjadi **1** agar dapat diproses oleh algoritma matematika. Setelah proses pembersihan, teks yang telah diproses (*clean_message*) diubah menjadi representasi numerik menggunakan *CountVectorizer* (metode *Bag of Words*). Proses ini membangun matriks frekuensi kata di mana setiap kolom mewakili kata unik dari seluruh dataset.

2.3. Algoritma Naive Bayes

Algoritma yang digunakan adalah Multinomial Naïve Bayes, varian Naïve Bayes yang optimal untuk data teks dengan fitur diskrit seperti frekuensi kata atau TF-IDF, karena sesuai dengan distribusi multinomial pada data kata [7]. Algoritma ini bekerja berdasarkan teorema Bayes dengan asumsi independensi antar fitur (kata dalam pesan)[1]. Tahapan algoritma sebagai berikut:

a. Hitung *prior probalility* untuk setiap kelas C_k (*Spam/phishing* atau *ham*):

$$P(C_k) = \frac{N_k}{N} \quad (1)$$

Di mana Nk adalah jumlah sampel kelas *K*, dan *N* adalah sampel.

b. Hitung *likelihood* untuk setiap kata *i* dalam kelas dengan *Laplace smoothing* ($\alpha = 1.0$) untuk menghindari nilai nol

$$P(x_i|C_k) = \frac{count(x_i,C_k)+\alpha}{\Sigma countt(x,C_k)+\alpha|v|} \quad (2)$$

c. Posterior: Menentukan kelas akhir berdasarkan *probalitas* tertinggi

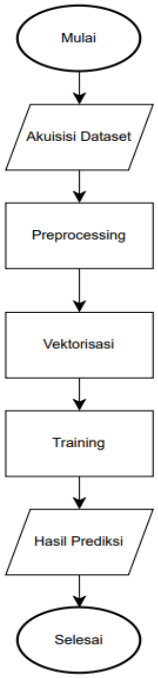
$$P(C_k \mid x) = \frac{P(C_k)\prod_i P(x_i \mid C_k)}{P(x)} \quad (3)$$

d. Parameter yang digunakan: aplha=1.0 (*smoothing*), fit_prior=*True*, dan pembagian data menggunakan *test_size*=0.2 (80% latih,20% uji) dengan parameter *stratify*=y [4].

e. Implementasi dilakukan menggunakan *library scikit-learn* dengan kelas MultinomialNB. Parameter yang dilakukan: alpha=1.0 (untuk *Laplace smoothing), fit_prior=True* (menghitung *prior* dari data), dan *class_prior*=None (tidak ada *prior* khusus). Model dilatih dengan model.fit(X_*train,* y_*train*), di mana X_*train* adalah matriks vektorisasi dari data latih.

2.4. Alur Proses Klasifikasi

Alur proses klasifikasi secara keseluruhan adalah sebagai berikut;



Gambar 1. *Flowchart* Alur proses Klasifikasi Naïve Bayes

Dataset diakuisisi dari Kaggle, kemudian data di-*impor* menggunakan fungsi pd.read_csv('spam.csv', encoding='latin-1'). Selanjutnya, dilakukan tahap *preprocessing* teks menggunakan fungsi *custom* seperti def preprocess_text(text): yang mencakup proses *lowercasing*, penghapusan tanda baca (*remove punctuation*), penghapusan *stopwords*, dan *stemming*. Setelah itu, teks diubah menjadi representasi numerik melalui proses vektorisasi menggunakan CountVectorizer() atau TfidfVectorizer(), misalnya dengan perintah vectorizer.fit_transform(texts).

Data kemudian dibagi menjadi data latih dan data uji menggunakan train_test_split(X, y, test_size=0.2, stratify=y, random_state=42) untuk menjaga proporsi kelas. Model selanjutnya dilatih menggunakan algoritma Multinomial Naïve Bayes dengan perintah nb_model = MultinomialNB(); nb_model.fit(X_train, y_train). Model yang telah dilatih digunakan untuk melakukan prediksi pada data uji atau pesan baru melalui nb_model.predict(teks_vec), serta menghitung probabilitas kepercayaan prediksi menggunakan nb_model.predict_proba(teks_vec). Terakhir, performa model dievaluasi menggunakan metrik seperti akurasi dan confusion matrix untuk menilai kinerja klasifikasi.

2.5. Metode Evaluasi Kinerja Algoritma

Evaluasi Kinerja dilakukan menggunakan metrik klasifikasi biner standar yang dihitung dari *confusion matrix* (*True Positive/TP, True Negative/TN, False Positive/FP, False Negative/FN*) menggunakan *library scikit-learn*. Berikut rumus-rumus utama beserta contoh implementasi kode pada lingkungan Python (seperti dinotebook kaggle atau Google Colab):

a. Accuracy digunakan untuk mengukur proporsi prediksi benar secara keseluruhan.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

b. Precision digunakan untuk mengukur akurasi prediksi kelas *spam/phishing* (proporsi prediksi *spam* yang benar-benar *spam.*

$$\text{Precision} = \frac{TP}{TP+FP} \tag{5}$$

c. Recall digunakan untuk mengukur kemampuan mendeteksi semua pesan spam/phishing (proporsi spam aktual yang terdeteksi).

$$\text{Recall} = \frac{TP}{TP+FN} \tag{6}$$

d. F1-Score adalah rata-rata harmonik precision dan recall yang sangat relevan untuk dataset *imbalanced*.

$$\text{F1} = 2 \text{ x } \frac{Precision x Recall}{Precision+Recall} \tag{7}$$

Di mana:
TP (True Positive): jumlah data yang diprediksi positif oleh sistem dan memang positif sebenarnya.
FP (False Positive): jumlah data yang diprediksi positif, tetapi sebenarnya negatif.
TN (True Negative): jumlah data yang diprediksi negatif oleh sistem dan memang negatif sebenarnya.
FN (False Negative): jumlah data yang diprediksi negativ, tetapi sebenarnya positif.

3. HASIL DAN PEMBAHASAN

3.1. Hasil

Hasil penelitian ini menunjukkan bahwa model Naive Bayes berhasil dikembangkan untuk mendeteksi pesan phising menggunakan dataset SMS *Spam Collection UCI Machine Learning Repository* yang terdiri dari 5.572 pesan, dengan 4.825 pesan berkategori ham (*non-phishing*) dan 747 pesan berkategori spam/*phising*. Setelah *preprocessing* teks (l*owercase*, penghapusan tanda baca dan angka, *stopwords removal*, serta *stemming* menggunakan *PorterStemmer*), data dibagi menjadi 80% data latih dan 20% data uji dengan stratifikasi.

Tabel 1. Distribusi *Label* pada *Dataset*

Label	Jumlah Pesan	Persentase (%)
Ham	4.825	86,59
Spam	747	13,41
Total	5.572	100,00

3.1.1. Evaluasi Kinerja Model

Hasil evaluasi menunjukkan bahwa model Naive Bayes memiliki kinerja yang sangat baik dalam mengklasifikasikan pesan Ham dan Spam. Pada kelas Ham, model memperoleh nilai *precision, recall*, dan F1-*score* masing-masing sebesar 0,99 dengan jumlah data sebanyak 966 sampel, yang menunjukkan bahwa hampir

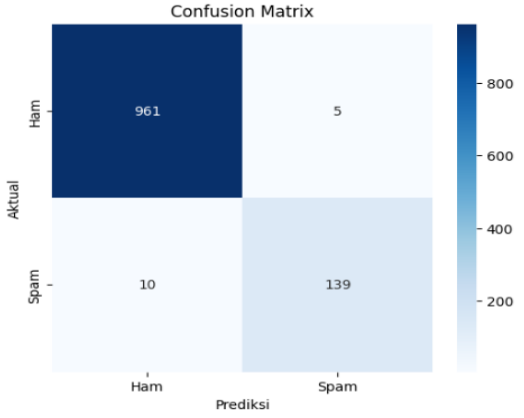
seluruh pesan Ham dapat diklasifikasikan dengan benar. Sementara itu, pada kelas Spam, model menghasilkan *precision* sebesar 0,97, *recall* sebesar 0,93, dan F1-*score* sebesar 0,95 dari total 149 sampel. Nilai *recall* yang lebih rendah pada kelas Spam mengindikasikan masih terdapat sebagian kecil pesan spam yang salah diklasifikasikan sebagai Ham. Secara keseluruhan, model mencapai akurasi sebesar 98,65% dari total 1.115 sampel. Nilai *macro average* dan *weighted average* yang tinggi menunjukkan bahwa model memiliki performa yang stabil dan konsisten meskipun terdapat perbedaan jumlah data pada masing-masing kelas.

Tabel 2. Hasil Evaluasi Model Naïve Bayes

Metrik	***Precision***	***Recall***	**F1-*Score***	***Support***
Ham	0,99	0,99	0,99	966
Spam	0,97	0,93	0,95	149
Accuracy	-	-	-	98,65%
Macro avg	0,98	0,96	0,97	1.115
Weighted avg	0,99	0,99	0,99	1.115

3.1.2. Analisis Confusion Matrix

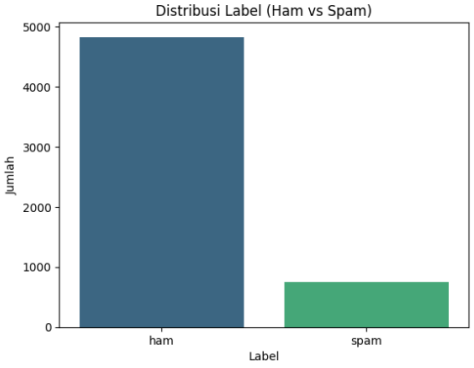
Pada Bagian *confusion matrix* menunjukkan perbandingan antara data aktual dan hasil prediksi model pada klasifikasi pesan Ham dan Spam. Pada kelas Ham, sebanyak 961 data berhasil diprediksi dengan benar sebagai Ham, sedangkan 5 data Ham salah diprediksi sebagai Spam. Sementara itu, pada kelas Spam, sebanyak 139 data berhasil diprediksi dengan benar sebagai Spam, dan 10 data Spam salah diprediksi sebagai Ham. Hasil ini menunjukkan bahwa sebagian besar data dapat diklasifikasikan dengan benar oleh model, dengan jumlah kesalahan prediksi yang relatif kecil, sehingga model memiliki kinerja yang sangat baik dalam membedakan pesan Ham dan Spam.



Gambar 2. *Confusion Matrix* Model Naïve Bayes

3.1.3. Distribusi Kelas dan Ketidakseimbangan Data

Selanjutnya menunjukkan distribusi jumlah data berdasarkan label Ham dan Spam. Terlihat bahwa jumlah pesan dengan label Ham jauh lebih banyak dibandingkan dengan pesan berlabel Spam.



Gambar 3. Distribusi Panjang Pesan berdasarkan Label

Perbedaan ini mengindikasikan adanya ketidakseimbangan kelas (*class imbalance*) pada dataset, di mana kelas Ham mendominasi keseluruhan data. Kondisi ini perlu diperhatikan dalam proses pelatihan dan evaluasi

model, karena ketidakseimbangan data dapat memengaruhi kinerja model klasifikasi, khususnya dalam mendeteksi kelas minoritas yaitu Spam.

3.1.4. Uji Coba Prediksi Data Baru

Pada pesan *"FREE entry to win $1000! Click here now…",* model memprediksi pesan sebagai Spam/*Phishing* dengan tingkat kepercayaan sebesar 99,94%, sedangkan kepercayaan sebagai Ham hanya sebesar 0,06%. Sebaliknya, pesan "Hey, apa kabar?" diprediksi sebagai pesan aman (Ham) dengan tingkat kepercayaan 99,98%. Selain itu, pesan "Besok ketemu ya." juga diklasifikasikan sebagai Ham, sementara pesan "*Your account has been locked. Verify now…"* terdeteksi sebagai Spam/Phishing dengan tingkat kepercayaan 92,07%. Hasil ini menunjukkan bahwa model mampu memberikan prediksi yang akurat disertai nilai probabilitas kepercayaan yang tinggi.

Tabel 3. Contoh Prediksi Pesan Baru

Pesan Asli	Prediksi	Kepercayaan Spam	Kepercayaan Ham
"FREE entry to win $1000! Click here now…"	Spam / Phishing Detected	99,94%	0,06%
"Hey, apa kabar? Besok ketemu ya."	Pesan Aman (Ham)	0,02%	99,98%
"Your account has been locked. Verify now…"	Spam / Phishing Detected	92,07%	7,93%

3.2. Pembahasan

Model Multinomial Naive Bayes yang dikembangkan mencapai akurasi 98,65% dengan *recall* kelas spam 0,93, menunjukkan kemampuan yang sangat baik dalam mendeteksi pesan *phising* meskipun dataset tidak seimbang (hanya 13,41%spam). Dari *confusion matrix*, terlihat hanya 10 pesan spam yang salah diklasifikasikan sebagai ham (*false negative)* dan 5 pesan yang salah sebagai spam (*false positive*), sehingga model relatif aman dalam mengurangi risiko phishing yang lolos deteksi.

Keberhasilan model ini dipengaruhi oleh beberapa faktor. Pertama, *preprocesing* teks yang komprehensif (*stemming, stopwords removal,* dan penghapusan *noise*) berhasil mengekstrak fitur kata kunci phishing seperti "*free", "win", "click*", dan "*verify*". Kedua, representasi *Bag-of-Words* melalui *CountVectorizer* sesuai dengan asumsi independensi kondisional Naive Bayes, yang membuat model efisien dan robust pada data teks pendek seperti SMS atau pesan media sosial.

Dibandingkan dengan penelitian Fahmi dan Fadillah (2024) yang menerapkan Naive Bayes untuk klasifikasi postingan pengguna Facebook dan mencapai akurasi 99,01%, hasil penelitian ini menunjukkan performa yang sangat kompetitif dengan selisih hanya 0,36%. Perbedaan kecil ini wajar karena karakteristik dataset yang berbeda: postingan Facebook umumnya lebih panjang dan kaya konteks, sementara dataset SMS lebih singkat dan langsung. Namun, hasil ini membuktikan bahwa pendekatan Naive Bayes tetap efektif dan dapat diiadaptasi dengan baik ke konteks pesan phishing pada media sosial modern, termasuk *Direct Message* atau komentar singkat.

Kelemahan utama model adalah ppotensi kegagalan pada pesan *phishing* baru yang menggunakan teknik obfuscation (misalnya penggantian huruf dengan angka/simbol atau bahasa campuran) yang tidak terdapat pada data latih. Pengembangan selanjutnya dapat menambahkan fitur tambahan seperti deteksi URL mencurigakan, analisisis kapitalisasi berlebih, atau penggunaan TF-IDF untuk meningkatkan sensitivitas terhadap variasi teks.

Temuan ini memberikan konstribusi praktis bagi pengembangan sistem terdeteksi phishing ototmatis berbiaya rendah pada plastform media sosial, dengan akurasi tinggi dan kecepatan inferensi cepat, sehingga dapat meningkatkan kesadaran serta keamanan pengguna terhadap ancaman phishing berbasis teks.

4. KESIMPULAN

Penelitian ini berhasil mengembangkan model deteksi phishing pada media sosial menggunakan metode Multinomial Naive Bayes berbasis dataset SMS *Spam Collection UCI Machine Learning Repository*. Model yang dibangun mencapai akurasi 98,65%, precision 0,97 untuk kelas spam, serta recall 0,93 untuk kelas spam, menunjukkan kemmapuan yang sangat baik dalam mengidentifikasi pesan *phising* meskipun dataset tidak seimbang. *Preprocessing* teks yang meliputi *stemming*, penghapusan *stopwords*, dan representasi *Bag-of-Words* terbukti efektif mendukung performa tabel. Temuan ini mengonfirmasi bahwa Naive Bayes merupakan pendekatan yang robust, efisien, dan akurat untuk tugas klasifikasi teks *phishing* pada pesan singkat, dengan

hasil yang kompetitif dibandingkan penelitian serupa pada konteks media sosial. Model ini dapat diadaptasi secara lansgung untuk aplikasi deteksi *phishing* pada platform seperti *Direct Message* Twitter/X, Instagram, atau komentar Facebook. Untuk pengembangan selanjutnya, disarankan penambahan fitur tambahan seperti deteksi URL mencurigakan, analisis kapitalisasi berlebih, atau penggunaan dataset yang lebih besar dan beragam dari media sosial aktual guna meningkatkan ketahanan model terhadap teknik *obfuscation phishing* terbaru.

DAFTAR PUSTAKA

[1] M. Fahmi, F. Arsyi, and N. Fadlilah, "Klasifikasi Postingan Pengguna Facebook Untuk Deteksi Phising Menggunakan Naive Bayes," 2023.

[2] L. A. D. Pasha and Z. Azis, "Predicting The Risk of Online Sales Fraud with The Naïve Bayes Approach on Facebook Social Media," *Hanif Journal of Information Systems*, vol. 2, no. 2, pp. 46–53, Feb. 2025, doi: 10.56211/hanif.v2i2.41.

[3] K. T. Tham, K. W. Ng, and S. C. Haw, "Phishing Message Detection Based on Keyword Matching," *Journal of Telecommunications and the Digital Economy*, vol. 11, no. 3, pp. 105–119, Sep. 2023, doi: 10.18080/jtde.v11n3.776.

[4] C. Feresa Mohd Foozy *et al.*, "Phishing URLs Detection Using Naives Baiyes, Random Forest and LightGBM Algorithms," 2024.

[5] A. Thiruvoth and P. Ogale, "A Phishing Detection System for Enhanced Cybersecurity Using Machine Learning," INSTICC, Jun. 2025, pp. 355–360. doi: 10.5220/0013570800003964.

[6] R. Jayaprakash *et al.*, "Heuristic machine learning approaches for identifying phishing threats across web and email platforms," *Front Artif Intell*, vol. 7, 2024, doi: 10.3389/frai.2024.1414122.

[7] T. Ige, C. Kiekintveld, A. Piplai, A. Waggler, O. Kolade, and B. H. Matti, "An investigation into the performances of the Current state-of-the-art Naive Bayes, Non-Bayesian and Deep Learning Based Classifier for Phishing Detection: A Survey," Nov. 2024, [Online]. Available: http://arxiv.org/abs/2411.16751

[8] M. Z. Haq, C. S. Octiva, A. Ayuliana, U. W. Nuryanto, and D. Suryadi, "Algoritma Naïve Bayes untuk Mengidentifikasi Hoaks di Media Sosial," *Jurnal Minfo Polgan*, vol. 13, no. 1, pp. 1079–1084, Jul. 2024, doi: 10.33395/jmp.v13i1.13937.

[9] Z. Lusi, "Identifikasi Komentar Spam Pada Sosial Media."

[10] A. Safi and S. Singh, "A systematic literature review on phishing website detection techniques," *Journal of King Saud University - Computer and Information Sciences*, vol. 35, no. 2, pp. 590–611, Feb. 2023, doi: 10.1016/j.jksuci.2023.01.004.