Vol.5 No.2 Tahun 2024
Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi
https://ejournal.unuja.ac.id/index.php/core
E-ISSN : 2774-7875 and P-ISSN : 2775-0124

# Penerapan Metode *K-Nearest Neighbors* dalam Mendeteksi *Website Phishing*

Arfian Jauhar Himawan[1*], Anindya Meyla Kartika Sari[2], Nabila Agatha Parsa[3], Keisha Sabilah Putri Hermansyah[4] , Elsya Sabrina Dea Rizki[5]
[1,2,3,4,5] Universitas Negeri Surabaya, Indonesia

**Info Artikel**

Riwayat Artikel
Diterima: 30-11-2024
Disetujui: 29-12-2024

Kata Kunci
*K-Nearest Neighbors*;
*Website Phishing*;
Klasifikasi;

*arfian.23026@mhs.unesa.ac.id

**ABSTRAK**

Penerapan metode *K-Nearest Neighbors* (KNN) dalam mendeteksi *website phishing* merupakan langkah penting untuk melindungi pengguna dari ancaman siber. *Phishing* adalah teknik penipuan yang bertujuan mencuri informasi sensitif dengan menyamar sebagai entitas terpercaya. Penelitian ini mengembangkan sistem deteksi menggunakan algoritma KNN untuk mengklasifikasikan *website* sebagai *phishing* atau *non-phishing*. Proses dimulai dengan pengumpulan *dataset* yang berisi contoh *website phishing* dan *non-phishing*, diikuti dengan ekstraksi fitur relevan seperti struktur URL dan konten. Model KNN dilatih dan diuji, menunjukkan tingkat akurasi yang tinggi dalam mendeteksi *phishing*. Temuan ini menegaskan bahwa KNN adalah alat efektif dalam meningkatkan keamanan siber, dengan potensi untuk mengatasi variasi serangan *phishing* yang lebih kompleks.

## 1. PENDAHULUAN

Pada era teknologi yang terus berkembang, internet memiliki peran yang penting dalam kehidupan baik bagi individu maupun organisasi. Berbagai aktivitas seperti perdagangan online, penjualan barang maupun jasa sebagian besar dilakukan melalui internet. Dengan kemajuan teknologi yang memberikan kemudahan, internet juga memberikan sejumlah ancaman seperti kejahatan dunia maya. Salah satu ancaman dunia maya yang signifikan adalah phishing, yaitu bentuk kejahatan meniru situs web yang legal untuk mencuri informasi sensitif seperti nama pengguna dan kata sandi ataupun data keuangan. *Website phishing* dirancang menyerupai *website* sah sehingga sulit dibedakan oleh pengguna.

*Phishing* tergolong dalam ancaman serius yang dapat menyebabkan kerugian finansial, pencurian identitas, dan lain sebagainya. Seiring dengan berkembangnya teknologi, metode yang dilakukan oleh pelaku *phishing* akan semakin canggih. Menurut dari statista.com, pada tahun 2024 telah terdeteksi lebih dari 932.000 *Website phishing* di seluruh dunia[1]. Industri online di seluruh dunia telah menjadi sasaran serangan *phishing* pada kuartal ketiga tahun 2024 yang mencapai paling besar yaitu social media sebesar 30.5%, webmail sebesar 21.2%, dan untuk ecommerce sebesar 8% [2].

Di kawasan Asia Tenggara, Negara Indonesia tercatat sebagai negara dengan kasus phising tertinggi pada urutan ketiga, dengan 97.000 serangan diantara total sekitar 500.000 serangan yang terjadi di Asia Tenggara pada tahun 2023 (Bisnis.com, 18 Maret 2024). Peningkatan yang cukup signifikan telah menunjukkan pentingnya pengembangan system deteksi *phishing* [3].

Berbagai penelitian terdahulu untuk mendeteksi *Website phishing* dengan metode penelitian yang berbeda-beda seperti random forest yang diikuti oleh decision tree, metode

Support Vector Machine (SVM), dan lain sebagainya. Salah satu algoritma yang sering digunakan dalam mendeteksi phishing yaitu menggunakan metode k-Nearest Neighbor (kNN). Penelitian sebelumnya telah menunjukkan bahwa algoritma kNN memiliki akurasi yang cukup tinggi dalam mendeteksi *Website phishing*. Namun, tetap terdapat tantangan dalam mendeteksi *Website phishing* yang semakin kompleks.

Penelitian ini bertujuan untuk memberikan solusi yang lebih efektif dan efisien dalam mendeteksi *Website phishing*, terutama dengan menggunakan algoritma KNN. Sebagai algoritma pembelajaran mesin berbasis *instance-based learning*, KNN dapat mengklasifikasikan data dengan akurat berdasarkan fitur utama dari web *phishing*. Dengan adanya penelitian ini, diharapkan dapat meningkatkan akurasi deteksi dan meminimalisir kesalahan dalam mendeteksi serta mengidentifikasi faktor-faktor yang membedakan *Website phishing* dari web yang sah.

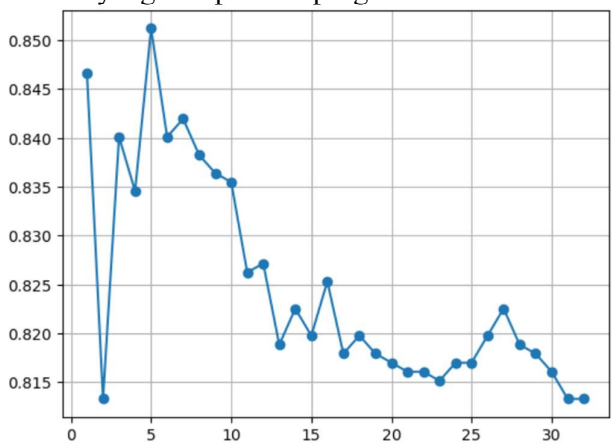
## 2. METODE

Metode yang digunakan dalam laporan ini adalah metode klasifikasi menggunakan *K-Nearest Neighbors* (KNN). *K-Nearest Neighbors* (KNN) adalah salah satu metode klasifikasi yang bekerja berdasarkan kemiripan atau kedekatan data baru dengan data yang telah diketahui kelasnya dalam ruang fitur. Metode ini termasuk algoritma non-parametrik, sehingga tidak memerlukan asumsi distribusi data tertentu. Proses klasifikasi dengan KNN dimulai dengan menghitung jarak antara data baru dan seluruh data dalam dataset, biasanya menggunakan metrik seperti Euclidean, Manhattan, atau Minkowski. Setelah itu, KNN memilih sejumlah tetangga terdekat (k) berdasarkan jarak yang telah dihitung, dan menentukan kelas data baru berdasarkan mayoritas kelas dari tetangga tersebut.

Kelebihan metode KNN adalah kesederhanaan implementasi dan kemampuannya untuk menangani data non-linear dengan baik. Namun, metode ini memiliki beberapa kekurangan, seperti sensitivitas terhadap data *outlier* dan kebutuhan komputasi tinggi untuk *dataset* besar, karena menghitung jarak antara data baru dan setiap data dalam *dataset*. Selain itu, pemilihan nilai k yang tepat sangat penting untuk mendapatkan hasil yang optimal; k yang terlalu kecil dapat menyebabkan *overfitting*, sementara k yang terlalu besar dapat menyebabkan *underfitting*. Meski demikian, KNN tetap menjadi metode yang populer dalam banyak aplikasi, seperti pengenalan pola, analisis teks, dan pengelompokan data.

## 3. HASIL DAN PEMBAHASAN

Berikut adalah hasil-hasil yang didapat dari program.



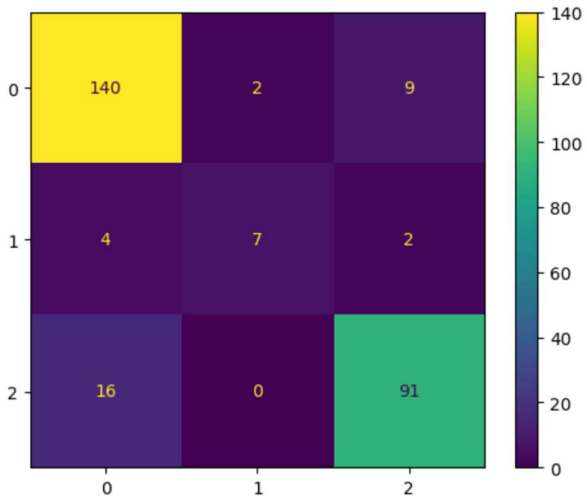

**Gambar 1.** Grafik yang menunjukkan nilai k optimal

Berdasarkan metode *cross-validation* diperoleh grafik tersebut yang menunjukkan bahwa nilai k optimal untuk proses klasifikasi yaitu k=5, yang digunakan sebagai acuan dalam

mempertimbangkan lima tetangga terdekat. Dengan memilih k optimal yaitu k=5, memberikan pengaruh yang signifikan terhadap penentuan akurasi dan efektivitas model klasifikasi. Nilai optimal tersebut dapat mengurangi *overfitting* yaitu model terlalu sesuai dengan data pelatihan dan kurang mampu men-generalisasi pada data baru atau dapat menyebabkan model terlalu sensitif terhadap *noise*, serta *underfitting* yaitu kondisi hilangnya detail penting dari data pelatihan [4].

Dengan menggunakan nilai k=5, model mampu mencapai keseimbangan secara fleksibel dalam mengenali pola data dan kemampuan dalam melakukan generalisasi terhadap data baru. Jadi, dapat disimpulkan bahwa model tidak hanya bekerja baik dalam data pelatihan tetapi juga memberikan performa yang terbaik pada data uji.

Langkah selanjutnya, mengevaluasi model berdasarkan matriks kebingungan (*confusion matriks*) yang akan menunjukkan distribusi prediksi model untuk setiap kelasnya dengan detail sebagai berikut :




**Gambar 2.** Gambar dari *confusion matrix*

*Confusion matrix* bertujuan menggambarkan kinerja model klasifikasi pada serangkaian data uji yang nilai sebenarnya diketahui [5]. Pada gambar yang menunjukkan hasil dari *confusion matrix* diperoleh beberapa keterangan, yaitu :

1. y-label = *True label* = Label aktual,
2. x-label = *Predicted label* = Label prediksi.

Selain itu, label juga terbagi menjadi 3 kelas, yaitu:

1. Kelas 0 yang mendefinisikan *website phishing* (label -1),
2. Kelas 1 yang mendefinisikan *website* mencurigakan (label 0),
3. Kelas 2 yang mendefinisikan *website* sah (label 1).

Berdasarkan hasil dari tabel *confusion matrix* tersebut diperoleh beberapa informasi, yaitu:

1. Kelas 0 yang mendefinisikan *website phishing* (label -1)

- *True Positive* : Prediksi "*website phishing*"- Aktual "*website phishing*" = 140 (Benar)
- *False Positive* : Prediksi "*website phishing*" - Aktual "bukan *website phishing*" = 20
- *False Negative* : Prediksi "bukan *website phishing*"- Aktual "*website phishing*" = 11

2. Kelas 1 yang mendefinisikan *website* mencurigakan (label 0)

- *True Positive* : Prediksi "*website* mencurigakan"- Aktual "*website* mencurigakan" = 7 (Benar)

- *False Positive* : Prediksi "*website* mencurigakan" - Aktual "bukan *website* mencurigakan" = 2
- *False Negative*: Prediksi "bukan *website* mencurigakan"- Aktual "*website* mencurigakan" = 0

3. Kelas 2 yang mendefinisikan *website* sah (label 1)

- *True Positive* : Prediksi "*website* sah"- Aktual "*website* sah" = 91 (Benar)
- *False Positive* : Prediksi "*website* sah" - Aktual "bukan *website* sah" = 11
- *False Negative* : Prediksi "bukan *website* yang sah"- Aktual "*website* sah" = 16

Untuk mengetahui berapa persen *website* yang benar diprediksi *phishing*, masih dicurigai maupun yang diprediksi sah dari keseluruhan *website*, dapat menggunakan pencarian nilai *accuracy*. Nilai akurasi diperoleh berdasarkan jumlah *website* diprediksi benar (*website* sah, *website* yang dicurigai, serta *website phishing*) dibagi dengan jumlah *website* keseluruhan. Diperoleh nilai akurasi dari model klasifikasi tersebut sebesar 0.88 atau 88%, yang menunjukkan bahwa nilai akurasi yang cukup tinggi sehingga model secara keseluruhan bekerja dengan baik.

Berikut kami konversikan hasil tersebut menjadi tabel agar lebih mudah dipahami.

**Tabel 1.** Tabel Nilai Akurasi dan Evaluasi

| ***Label*** | ***Precision*** | ***Recall*** | ***F1-score*** | ***Recall*** |
|---|---|---|---|---|
| -1 | 0.88 | 0.93 | 0.90 | 151 |
| 0. | 0.78 | 0.54 | 0.64 | 13 |
| 1. | 0.89 | 0.85 | 0.87 | 107 |
| | | | | |
| *accuracy* | | | 0.88 | 271 |
| *Macro Avg* | 0.85 | 0.77 | 0.80 | 271 |
| *Weighted Avg* | 0.88 | 0.88 | 0.88 | 271 |

Berdasarkan tabel diatas, dapat diperoleh informasi sebagai berikut.

### 3.1 Analisis Performa Model

Pada subbab ini dapat diketahui peforma kinerja model yang dihasilkan pada label yang telah diketahui sebelumnya, yaitu.

#### 3.1.1 Kinerja pada Label (-1)

Pada label (-1) diperoleh kinerja sebagai berikut.

a. *Precision* = 0.88
b. *Recall* = 0.93
c. *F1-score* = 0.90

Hal ini menunjukkan bahwa model memperoleh presentase akurasi sekitar 88% dari prediksi *website phishing* yang benar, model juga berhasil menangkap sekitar 93% dari semua *website phishing* yang ada, selain itu model menunjukkan keseimbangan yang sangat baik antara *precision* dan *recall* yaitu sebesar 90%.

#### 3.1.2 Kinerja pada Label (0)

Pada label (0) diperoleh kinerja sebagai berikut.

a. *Precision* = 0.78
b. *Recall* = 0.54
c. *F1-score* = 0.64

Berdasarkan nilai diatas, menunjukkan bahwa performa cenderung rendah, model dapat memprediksi 78% *website* mencurigakan dengan benar, namun masih ada sekitar 22% dari prediksi yang salah, model hanya berhasil menangkap 54% dari semua *website* mencurigakan yang ada. Meskipun *precision*-nya cukup baik, *recall*-nya jauh lebih rendah sehingga mempengaruhi nilai *F1-score*. Hal ini mungkin disebabkan jumlah data (*support* = 13) untuk label ini yang jauh lebih sedikit dibanding label lainnya, sehingga model kurang bisa mempelajari pola dengan maksimal.

### 3.1.3 Kinerja pada Label (1)

Pada label (1) diperoleh kinerja sebagai berikut.

a. *Precision* = 0.89
b. *Recall* = 0.85
c. *F1-score* = 0.87

Label ini menunjukkan performa yang baik dengan 89% dari semua prediksinya benar, serta model berhasil menangkap 85% dari semua *website* sah yang ada, selain itu dengan *F1-score* 87% menunjukkan bahwa model memiliki keseimbangan yang sangat baik antara *precision* dan *recall*.

### 3.1.4 Keseluruhan Performa (*Accuracy* dan Rata-rata)

Pada keseluruhan performa dapat diperoleh informasi sebagai berikut.

a. *Accuracy* sebesar 88% menunjukkan bahwa model ini berhasil memprediksi dengan benar 88% dari seluruh data.
b. *Macro average* (rata-rata dari metrik (*precision, recall,* dan *F1-score*)) untuk setiap kelas secara terpisah, tanpa mempertimbangkan distribusi jumlah data dalam setiap kelas), yaitu:
    - *Precision* = 0.85
    - *Recall* = 0.77
    - *F1-score* = 0.80

Dengan *precision* sebesar 85%, model cukup baik dalam menghindari kesalahan positif (*false positive*) untuk setiap kelas, rata-rata *recall* yang cukup rendah kemungkinan disebabkan *recall* label (0) yang juga rendah. Selain itu dengan *F1-score* 80%, model menunjukkan keseimbangan yang cukup baik antara *precision* dan *recall*, meskipun *recall* masih sedikit lebih rendah daripada *precision*.

c. *Weighted average* (rata-rata dari metrik dengan memperhitungkan distribusi jumlah data dalam setiap kelas), yaitu:
    - *Precision* = 0.88
    - *Recall* = 0.88
    - *F1-score* = 0.88

*Weighted average* yang tinggi (88% untuk *Precision*, *Recall*, dan *F1-score*) mengindikasikan bahwa model bekerja lebih baik pada label dengan jumlah data yang lebih besar (label -1 dan 1).

## 4. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian dan analisis data, dapat disimpulkan bahwa Metode *K-Nearest Neighbors* (KNN) efektif dalam mendeteksi *website phishing* dengan akurasi sebesar 88%. Metode ini mampu mengklasifikasikan *website* ke dalam tiga kategori, yaitu *phishing* (label -1), mencurigakan (label 0), dan sah (label 1), dengan kinerja yang baik pada label mayoritas (-1 dan 1). Nilai optimal untuk parameter k adalah 5, yang memberikan keseimbangan antara *underfitting* dan *overfitting*, sehingga model dapat mengenali pola pada data dengan efektif tanpa terpengaruh *noise*.

Evaluasi model menunjukkan bahwa *precision* tertinggi dicapai pada *website phishing* 88% dan *website* sah 89%, sedangkan *recall* untuk *website phishing* mencapai 93%,

menunjukkan kemampuan model mendeteksi sebagian besar kasus *phishing*. Namun, performa untuk *website* mencurigakan masih rendah (*recall* 54% dan *F1-score* 64%) akibat jumlah data yang terbatas.

Sehingga penelitian ini berhasil menentukan metrik evaluasi yang relevan, seperti *accuracy*, *precision*, *recall*, dan *Fl-score*, yang memberikan gambaran menyeluruh tentang kinerja model. Dengan demikian, hasil penelitian ini tidak hanya menjawab rumusan masalah secara jelas, tetapi juga mencapai tujuan yang telah ditetapkan, yakni mengevaluasi efektivitas KNN dalam mendeteksi *website phishing* serta menentukan parameter dan metrik evaluasi yang optimal.

Berdasarkan hasil penelitian, disarankan agar penelitian serupa di masa mendatang memperhatikan distribusi data yang lebih seimbang, terutama untuk label dengan jumlah data minoritas seperti *website* mencurigakan, guna meningkatkan performa model dalam mendeteksi semua kategori dengan lebih baik. Hal ini dapat dilakukan dengan menambah jumlah sampel melalui teknik *oversampling*, seperti SMOTE, atau pengumpulan data tambahan dari sumber yang relevan. Selain itu, perusahaan yang bergerak di bidang teknologi dan keamanan siber disarankan untuk mengadopsi algoritma KNN sebagai bagian dari sistem deteksi *phishing*, dengan penyesuaian parameter nilai k untuk mengoptimalkan akurasi sesuai dengan kebutuhan.

## 5. DAFTAR PUSTAKA

[1]. Petrosyan. Ani, “Number of global phishing sites Q3 2013- Q3 2024”, Statista, 9 Desember 2024. [Online], Tersedia : https://www.statista.com/statistics/266155/number-of-phishing-domain-names-worldwide/ [Diakses: 30 Desember 2024]

[2]. Statista, "Websites Most Affected by Phishing," 2024. [Online]. Available: https://www.statista.com/statistics/266161/websites-most-affected-by-phishing/. [Diakses: 30 Desember 2024].

[3]. Suhartanto. Crysania, “RI Peringkat Ketiga, Negara dengan Serangan Phising Terbanyak di Asean 2023”, Teknologi Bisnis, 18 Maret 2024. [Online], Tersedia : https://teknologi.bisnis.com/read/20240318/84/1750246/ri-peringkat-ketiga-negara-dengan-serangan-phising-terbanyak-di-asean-2023 [Diakses: 30 Desember 2024]

[4]. Dicoding. "Apa itu Phishing dan Bagaimana Cara Mencegahnya," Dicoding Indonesia, https://www.dicoding.com/academies/184/tutorials/38728 [Diakses: 30 Desember 2024].

[5]. K. S. Nugroho, "Confusion Matrix untuk Evaluasi Model pada Unsupervised Machine Learning," Medium, Nov. 17, 2020. [Online]. Available: https://ksnugroho.medium.com/confusion-matrix-untuk-evaluasi-model-pada-unsupervised-machine-learning-bc4b1ae3f. [Diakses: 30 Desember 2024].

[6]. Rahayu, S., Mz, Y., Bororing, J. E., & Hadiyat, R. (2022). Implementasi Metode K-Nearest Neighbor (K-NN) untuk Analisis Sentimen Kepuasan Pengguna Aplikasi Teknologi Finansial FLIP. Edumatic J. Pendidik. Inform, 6(1), 98-106.

[7]. Pawening, R. E., Shudiq, W. J. F., & Wahyuni, W. (2020). Klasifikasi Kualitas Jeruk Lokal Berdasarkan Tekstur dan Bentuk Menggunakan Metode k-Nearest Neighbor (k-NN). COREAI: Jurnal Kecerdasan Buatan, Komputasi dan Teknologi Informasi, 1(1), 10-17.

[8]. Homaidi, A., & Fatah, Z. (2024). Implementasi Metode K-Nearest Neighbors (KNN) untuk Klasifikasi Penyakit Jantung. G-Tech: Jurnal Teknologi Terapan, 8(3), 1720-1728.

[9]. Arifin, Z., Shudiq, W. J., & Maghfiroh, S. (2019). Penerapan Metode Knn (K-Nearest Neighbor) Dalam Sistem Pendukung Keputusan Penerimaan Kip (Kartu Indonesia Pintar) Di Desa Pandean Berbasis Web Dan Mysql. NJCA (Nusantara Journal of Computers and Its Applications), 4(1), 27-34.

[10]. Aisyah, A., & Anraeni, S. (2022). Analisis penerapan metode K-Nearest Neighbor (K-NN) pada dataset citra penyakit malaria. Indonesian Journal of Data and Science, 3(1), 17-29.

[11]. Shudiq, W. J. F., As, A. H., & Rahman, M. F. (2020). Penentuan Metode Terbaik Dalam Menentukan Jenis Pohon Pisang Menurut Tekstur Daun (Metode K-NN dan SVM). Jurnal Teknologi dan Manajemen Informatika, 6(2), 128-136.

[12]. Rismala, R., Ali, I., & Rinaldi, A. R. (2023). Penerapan Metode K-Nearest Neighbor Untuk Prediksi Penjualan Sepeda Motor Terlaris. JATI (Jurnal Mahasiswa Teknik Informatika), 7(1), 585-590.

[13]. Widad, N. R., Shudiq, W. J. F., & Nadhiroh, A. Y. (2024). Designing a Website-Based Tracking of Sales Information System to Improve Business Performance at Estoh Jember Company. Jurnal Indonesia Sosial Teknologi, 5(10).

[14]. Hidayat, M. T., & Laluma, R. H. (2022). Penerapan Metode K-Nearest Neighbor Untuk Klasifikasi Gizi Balita. Infotronik: Jurnal Teknologi Informasi dan Elektronika, 7(2), 64-69.

[15]. Permatasari, U. O. R., Shudiq, W. J. F., & Jasri, M. (2024). Prediksi Kelayakan Mahasiswa sebagai Penerima Beasiswa Bank Indonesia pada Tahap Seleksi Administrasi di Universitas Nurul Jadid Menggunakan Algoritma K Nearest Neighbor. Journal of Electrical Engineering and Computer (JEECOM), 6(1), 252-260.