Sistemasi: Jurnal Sistem Informasi ISSN:2302-8149
Volume 15, Nomor 5, 2026: 1721-1730 e-ISSN:2540-9719

Analisis Perbandingan Model *TinyBERT*, *SVM*, dan *Char-CNN* pada Deteksi *URL Phishing*

Comparative Analysis of TinyBERT, SVM, and Char-CNN Models for Phishing URL Detection

[1]Haeranisa Bella Krisanti*, [2]Chaerul Umam
[1,2]Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Dian Nuswantoro
[1,2]Jl. Imam Bonjol No. 207, Pendrikan Kidul, Kec. Semarang Tengah, Kota Semarang, Jawa Tengah 50131, Indonesia
*e-mail: 111202214786@mhs.dinus.ac.id

(***received*:** 25 April 2026, ***revised*:** 7 May 2026, ***accepted*:** 8 May 2026)

Abstrak

Phishing merupakan salah satu ancaman keamanan siber yang banyak memanfaatkan URL berbahaya untuk menipu pengguna dan mencuri informasi sensitif. Penelitian ini mengusulkan metode deteksi phishing berbasis URL menggunakan model Transformer ringan TinyBERT serta membandingkannya dengan tiga baseline, yaitu SVM berbasis character n-gram, Random Forest berbasis fitur leksikal URL, dan Char-CNN. Dataset yang digunakan terdiri dari 49.750 URL dengan label multi-kelas (benign, defacement, malware, dan phishing) yang kemudian dibinerisasi menjadi phishing (label 1) dan non-phishing (label 0). Data dibagi menggunakan stratified split menjadi train-validation-test (70%-15%-15%). Untuk menangani ketidakseimbangan kelas, model TinyBERT dilatih menggunakan weighted loss berdasarkan class weight. Evaluasi dilakukan menggunakan confusion matrix, accuracy, precision, recall, F1-score, serta kurva ROC dan Precision–Recall. Hasil eksperimen menunjukkan bahwa TinyBERT mencapai performa terbaik dengan akurasi 0,9925, recall phishing 0,9512, dan F1-score 0,9387, serta menghasilkan false negative paling rendah (22) dibanding baseline. Temuan ini menunjukkan bahwa TinyBERT lebih efektif dalam meminimalkan phishing yang lolos sebagai benign, sehingga lebih sesuai untuk penerapan deteksi phishing berbasis URL pada sistem keamanan siber.

Kata kunci: *Char-CNN, phishing, SVM, TinyBert, URL phishing detection, random forest*

Abstract

Phishing is one of the most prevalent cybersecurity threats that exploits malicious URLs to deceive users and steal sensitive information. This study proposes a URL-based phishing detection method using the lightweight Transformer model TinyBERT and compares its performance with three baseline models: SVM based on character n-grams, Random Forest based on lexical URL features, and Char-CNN. The dataset used in this study consists of 49,750 URLs with multi-class labels (benign, defacement, malware, and phishing), which were subsequently binarized into phishing (label 1) and non-phishing (label 0). The data were divided using a stratified split into training, validation, and testing sets with a ratio of 70%–15%–15%. To address class imbalance, the TinyBERT model was trained using a weighted loss approach based on class weights. The evaluation was conducted using a confusion matrix, accuracy, precision, recall, F1-score, as well as ROC and Precision–Recall curves. Experimental results demonstrate that TinyBERT achieved the best performance, with an accuracy of 0.9925, phishing recall of 0.9512, and an F1-score of 0.9387. In addition, the model produced the lowest number of false negatives (22) compared with the baseline models. These findings indicate that TinyBERT is more effective in minimizing phishing URLs that are incorrectly classified as benign, making it more suitable for implementing URL-based phishing detection in cybersecurity systems.

Keywords: *Char-CNN, phishing, SVM, TinyBert, URL phishing detection, random forest*

 ISSN:2302-8149
e-ISSN:2540-9719

1 Pendahuluan

Phishing merupakan salah satu ancaman keamanan siber di dunia maya yang menggunakan teknik manipulasi sosial untuk mengelabui pengguna agar membagikan informasi sensitif. Menurut penelitian terbaru pada bidang phishing detection, jenis serangan ini masih terus berubah dan menunjukkan peningkatan dalam kerumitan teknik penyamaran [1], [2]. Pola URL phishing sering didesain menyerupai URL yang valid melalui pemanfaatan subdomain yang panjang, token acak, atau karakter atau symbol tertentu sehingga sulit dibedakan secara kasat mata. Salah satu metode yang sering digunakan adalah deteksi phishing berbasis URL tanpa harus menganalisis konten halaman web. Metode ini dianggap lebih efektif karena dapat diterapkan pada sistem real-time dan tidak memerlukan proses crawling atau rendering halaman [1]. Model tradisional biasanya memanfaatkan fitur leksikal URL, misalnya panjang URL, jumlah simbol khusus, ataupun karakteristik domain, kemudian diklasifikasikan menggunakan algoritma machine learning seperti Random Forest. Namun, sebagaimana dijelaskan dalam literatur, pendekatan berbasis fitur statis dapat mengalami keterbatasan dalam menghadapi teknik phishing yang adaptif dan terus berubah [2]. Sebagai alternatif, beberapa studi mengusulkan model deep learning berbasis karakter. Misalnya, pendekatan Char-CNN mempelajari URL sebagai urutan karakter mentah sehingga mampu menangkap pola string yang tidak selalu terwakili oleh fitur leksikal [3]. Selain itu, penggunaan representasi n-gram juga tetap relevan karena mampu merepresentasikan potongan substring yang khas pada URL berbahaya, sebagaimana ditunjukkan pada metode GramBeddings yang memanfaatkan embedding n-gram untuk identifikasi URL phishing [10]. Perkembangan Transformer juga mendorong pemanfaatan model berbasis attention untuk mendeteksi URL berbahaya. Penelitian TransURL menunjukkan bahwa Transformer dapat memberikan kinerja yang kompetitif dalam deteksi malicious URL karena kemampuannya memodelkan dependensi token secara lebih kaya [5]. Pada sisi lain, model seperti TCURL mengombinasikan Transformer dan CNN untuk meningkatkan representasi pada tugas deteksi phishing URL [6]. Akan tetapi, penggunaan Transformer dalam sistem nyata sering terkendala oleh biaya komputasi dan ukuran model.

TinyBERT sebagai model hasil distilasi menjadi solusi yang menarik karena mampu mempertahankan performa yang baik dengan ukuran yang lebih ringan. TinyBERT diperkenalkan sebagai strategi distilasi BERT untuk efisiensi komputasi pada tugas pemahaman bahasa alami [1], dan pendekatannya juga relevan untuk klasifikasi teks pendek seperti URL. Studi lain juga menunjukkan bahwa pendekatan berbasis BERT dapat diterapkan untuk klasifikasi URL berbahaya secara efektif [7]. Oleh karena itu, penelitian ini mengimplementasikan TinyBERT untuk mendeteksi phishing berbasis URL dan membandingkannya dengan baseline SVM character n-gram, Random Forest fitur leksikal, serta Char-CNN. Selain itu, peningkatan penggunaan internet dan layanan digital menyebabkan potensi serangan phishing semakin luas, terutama pada sektor perbankan, e-commerce, dan layanan berbasis akun pengguna. Serangan phishing modern tidak hanya memanfaatkan kemiripan tampilan halaman, tetapi juga mengandalkan teknik manipulasi URL yang semakin kompleks dan sulit dideteksi oleh metode tradisional [1], [2]. Oleh karena itu, diperlukan pendekatan yang lebih adaptif dan mampu menangkap pola tersembunyi dalam URL.

Dalam konteks ini, pemanfaatan deep learning dan Transformer menjadi semakin relevan karena kemampuannya dalam memodelkan representasi data secara otomatis tanpa bergantung pada fitur manual. Penelitian sebelumnya menunjukkan bahwa model berbasis Transformer mampu memberikan performa yang lebih baik dalam tugas klasifikasi teks dibandingkan motode tradisional [5], [7]. Penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem deteksi phishing berbasis URL yang lebih akurat, efisien, dan siap digunakan dalam skenario nyata.

2 Tinjauan Literatur

Penelitian terkait penelitian deteksi phishing berbasis URL telah banyak dilakukan dengan berbagai pendekatan, mulai dari metode machine learning tradisional hingga deep learning. Salah satu pendekatan yang umum digunakan adalah pemanfaatan fitur leksikal URL yang kemudian diklasifikasikan menggunakan algoritma seperti Random Forest. Pendekatan ini cukup efektif dalam mengenali pola dasar URL phishing, namun memiliki keterbatasan dalam menangani variasi pola yang semakin kompleks dan adaptif [5].

 ISSN:2302-8149
e-ISSN:2540-9719

Pendekatan lain yang banyak digunakan adalah metode berbasis karakter, seperti Character-level Convolutional *Neural Network* (Char-CNN). Model ini memproses URL sebagai urutan karakter sehingga mampu menangkap pola tersembunyi tanpa perlu ekstraksi manual. Penelitian menunjukkan bahwa Char-CNN mampu memberikan performa yang baik dalam mendeteksi URL phishing dengan karakteristik yang beragam [3]. Selain itu, penggunaan teknik representasi n-gram juga menjadi pendekatan yang relevan dalam mendeteksi URL berbahaya. Metode ini bekerja dengan memecah URL menjadi potongan karakter tertentu sehingga dapat mengidentifikasi pola substring yang khas pada URL phishing. Pendekatan ini sering dikombinasikan dengan algoritma klasifikasi seperti Support Vector Machine (SVM) untuk meningkatkan akurasi deteksi [10].

Seiring dengan perkembangan teknologi, model berbasis Transformer mulai banyak digunakan dalam deteksi phishing. Model seperti TransURL menunjukkan bahwa mekanisme attention mampu menangkap hubungan antar token secara lebih kompleks dibandingkan metode sebelumnya [5]. TinyBERT sebagai model hasil distilasi BERT menawarkan solusi yang lebih ringan dengan performa tetap kompetitif dalam klasifikasi URL phishing [1]. Meskipun berbagai metode telah dikembangkan untuk mendeteksi phishing berbasis URL, masih terdapat beberapa keterbatasan yang perlu diperhatikan. Pendekatan berbasis fitur leksikal cenderung bergantung pada fitur yang telah ditentukan sebelumnya sehingga kurang adaptif terhadap pola baru [1]. Sementara itu, model deep learning seperti CNN membutuhkan data yang cukup besar dan komputasi yang relatif tinggi. Selain itu, sebagian penelitian belum secara spesifik menangani permasalahan ketidakseimbangan data (class imbalance) yang dapat menyebabkan model bias terhadap kelas mayoritas [11]. Oleh karena itu, penelitian ini mengintegrasikan pendekatan weighted loss untuk meningkatkan performa deteksi pada kelas minoritas.

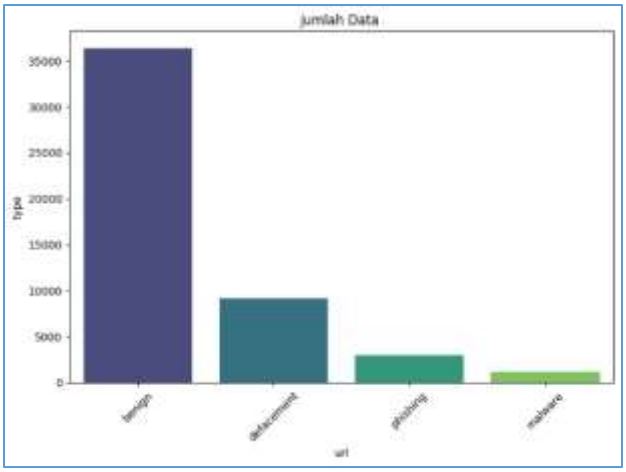
3 Metode Penelitian

Penelitian ini bertujuan untuk membangun model klasifikasi URL phishing menggunakan teknik deep learning berbasis Transformer, yaitu TinyBERT, untuk membedakan URL phishing dan URL benign. Tahapan penelitian meliputi pengumpulan dataset, pra-pemrosesan data, pembagian data train/validation/test, perancangan dan pelatihan model, serta pengujian dan evaluasi. Evaluasi dilakukan menggunakan metrik Accuracy, Precision, Recall, F1-score, ROC-AUC, dan confusion matrix serta analisis kesalahan false positive dan false negative. TinyBERT merupakan hasil distilasi dari BERT yang dirancang untuk mempertahankan performa tinggi dengan ukuran model yang lebih kecil [1]. Model ini menggunakan mekanisme self-attention untuk memahami hubungan antar token dalam suatu urutan teks, termasuk URL. Pada penelitian ini, URL diproses sebagai sequence token yang kemudian dikonversi menjadi embedding sebelum masuk ke layer Transformer. Proses pelatihan dilakukan dengan fine-tuning terhadap dataset phishing sehingga model dapat menyesuaikan representasi terhadap karakteristik URL berbahaya [7].

3.1 Pengambilan Data

Penelitian ini menggunakan dataset berisi 49.750 URL dengan label multi-kelas: benign, defacement, malware, dan phishing. Sebagaimana ditunjukkan pada Gambar 1. Mengacu pada praktik umum dalam penelitian phishing detection [15], fokus penelitian diarahkan pada deteksi phishing sehingga label dataset dibinerisasi menjadi phishing (label 1) dan non-phishing (label 0). Distribusi kelas yang dominan pada label benign menyebabkan dataset bersifat tidak seimbang, kondisi yang sering ditemui pada permasalahan keamanan siber nyata [14], dan tantangan utama pada dataset ini adalah ketidakseimbangan kelas (class imbalance).

 ISSN:2302-8149
e-ISSN:2540-9719



Gambar 1 Jumlah data

Dataset yang digunakan memiliki distribusi kelas yang tidak seimbang, dimana jumlah URL benign jauh lebih banyak dibandingkan phishing. Kondisi ini umum terjadi pada permasalahan keamanan siber [14]. Untuk mengatasi hal ini, digunakan pendekatan weight loss, yaitu memberikan bobot lebih besar pada kelas minoritas (phishing). Pendekatan ini terbukti efektif dalam meningkatkan performa model pada kelas minoritas tanpa mengorbankan performa keseluruhan secara signifikan[11].

3.2 Pra-pemrosesan Data (*Preprocessing*)

Pra-pemrosesan data dilakukan untuk memastikan dataset bersih, konsisten, serta dapat digunakan secara optimal pada proses pelatihan model baseline maupun model Transformer. Tahapan preprocessing yang dilakukan pada penelitian ini adalah sebagai berikut:

1. Menghapus kolom
   Dataset awal memiliki kolom Unnamed: 0 yang merupakan indeks hasil penyimpanan dan tidak mengandung informasi terkait karakteristik URL. Oleh karena itu, kolom ini dihapus agar tidak memengaruhi proses pemodelan.
2. Missing values
   Pada tahap berikutnya dilakukan pembersihan data dengan menghapus baris yang memiliki nilai kosong pada kolom url atau type karena model klasifikasi membutuhkan input URL yang valid serta label yang lengkap.
3. Normalisasi label dan transformasi menjadi klasifikasi biner
   Dataset asli berisi label multi-kelas (benign, defacement, phishing, malware). Karena fokus penelitian adalah deteksi phishing, label diubah menjadi klasifikasi biner:
   - label = 1 untuk URL dengan type = phishing
   - label = 0 untuk URL dengan type = non-phishing (gabungan benign + defacement + malware)

   Transformasi dilakukan dengan cara menurunkan seluruh teks label menjadi huruf kecil (lowercase) agar konsisten terhadap perbedaan format penulisan label.
4. Menyiapkan fitur input (URL) dan target (label)
   Setelah data bersih dan label biner terbentuk, kolom URL digunakan sebagai input utama. Semua URL dikonversi menjadi tipe string untuk menjaga konsistensi data.
5. Train–Validation–Test
   Dataset dibagi menjadi data training, validation, dan test menggunakan metode stratified split. Pembagian data dilakukan dengan rasio 70% training, 15% validation, dan 15% test.

3.3 Konfigurasi Model

Penelitian ini menggunakan empat model untuk klasifikasi URL phishing, terdiri dari tiga baseline (SVM character n-gram, Random Forest fitur leksikal, dan Char-CNN) serta satu model

 ISSN:2302-8149
e-ISSN:2540-9719

utama berbasis Transformer (TinyBERT). Seluruh model dirancang untuk melakukan klasifikasi biner, yaitu phishing (label 1) dan non-phishing (label 0).

Tabel 1 Hyperparameter model

Model	Hyperparameter
SVM (Char 3-5gram)	TF-IDF (min_df=2), LinearSVC, CalibratedCV (sigmoid, cv=3)
Random Forest (Lexical)	n_estimators=300, class_weight=balanced, random_state=42
Char-CNN	maxlen=200, Embedding=64, Conv1D(128,k=5), Dropout=0.3, Epoch=5, Batch=128
TinyBERT	max_len=128, Epoch=2, lr=2e-5, batch=32/64, wd=0.01, weighted loss

Tabel 1 menunjukkan konfigurasi hyperparameter yang digunakan pada setiap model untuk mendukung proses deteksi phishing berbasis URL. Model SVM berbasis character n-gram menggunakan respresentasi TF-IDF dengan *min_df=2* untuk mengurangi noise dari token yang jarang muncul. Algoritma utama yang digunakan adalah LinearSVC karena efektif menangani data teks berdimensi tinggi, sedangkan *CalibratedClassifierCV* dengan metode sigmoid dan *cv=3* diterapkan agar model mampu menghasilkan probibalitas prediksi yang lebih stabil untuk evaluasi Precision-Recall curve. Pada model Random Forest berbasis fitur leksikal, digunakan *n_estimators=300* untuk meningkatkan stabilitas generalisasi model, sementara class_weight=balanced diterapkan untuk mengatasi ketidakseimbangan kelas antara URL phishing dan non-phishing. Parameter *random_state=42* digunakan untuk menjaga konsistensi hasil eksperimen.

Sementara itu, model Char-CNN menggunakan panjang maksimum URL sebesar 200 karakter dengan layer embedding berukuran 64 untuk mempresesntasikan karakter ke dalam bentuk vektor numerik. Lapisan *Conv1D* dengan 128 filter dan kernel berukuran 5 digunakan untuk menangkap pola lokal antara karakter URL, sedangkan *dropout=0,3* diterapkan untuk mengurangi overfitting. Model filatih selama 5 epoch dengan batch size 128 agar pelatihan tetap efisien. Pada model TinyBERT, digunakan *max_len=128* untuk menyesuaikan panjang URL, dengan proses fine-tuning selama 2 epoch meggunakan *learning rate* 2e-5 agar penyesuaian parameter berlangsung stabil. Selain itu, diterapkan *wight decay=0,01* sebagai regularisasi dan *weighted loss* untuk meningkatkan sensivitas model terhadap kelas phishing pada dataset yang bersifat imbalanced.

3.4 Skenario Evaluasi

Skenario evaluasi pada penelitian ini dirancang untuk menilai kinerja model dalam mendeteksi URL phishing pada klasifikasi biner (phishing vs non-phishing). Seluruh model dievaluasi menggunakan data uji (test set) yang tidak digunakan pada proses pelatihan. Mengingat dataset bersifat tidak seimbang, evaluasi difokuskan pada performa kelas phishing (label 1), terutama untuk meminimalkan false negative.

1. Evaluasi dilakukan pada test set (data tidak digunakan saat training)
2. Metrik utama: Accuracy, Precision1, Recall1, F1-score1 (kelas phishing = 1).
3. Analisis kesalahan menggunakan Confusion Matrix untuk memperoleh FP dan FN
4. Evaluasi berbasis threshold menggunakan ROC Curve (AUC) dan Precision–Recall Curve (AP).
5. Perbandingan model dilakukan dengan overlay ROC/PR untuk baseline vs TinyBERT.

4 Hasil dan Pembahasan

4.1 Hasil Evaluasi Model

Pengujian dilakukan pada data uji (test set) untuk mengevaluasi kinerja model dalam mendeteksi URL phishing. Evaluasi difokuskan pada kelas phishing (label 1) dengan mempertimbangkan accuracy, precision, recall, F1-score, serta analisis FP dan FN melalui confusion matrix. Ringkasan hasil ditunjukkan pada **Tabel 2.**

Tabel 2 Perbandingan performa model

Model	Accuracy	Precision1	Recall1	F1_1	FP	FN
Lexical	0,9912	0,9661	0,8847	0,9236	14	52

 ISSN:2302-8149
e-ISSN:2540-9719

Features + Random Forest						
Char-CNN	0,9910	0,9638	0,8847	0,9225	15	52
SVM (Char 3–5gram TF-IDF)	0,9897	0,9452	0,8803	0,9116	23	54
TinyBERT (Transformer)	**0,9925**	0,9266	**0,9512**	**0,9387**	**34**	**22**

Berdasarkan Tabel 2, seluruh model menunjukkan akurasi tinggi (>0,989). Namun, perbedaan utama terlihat pada metrik kelas phishing (label 1), khususnya recall dan jumlah false negative (FN). Jika dibandingkan secara keseluruhan, setiap model memiliki karakteristik yang berbeda dalam mendeteksi phishing. Model berbasis fitur leksikal seperti Random Forest cenderung lebih stabil dan memiliki false positive rendah, namun kurang adaptif terhadap variasi pola URL baru [1]. Sementara itu, model Char-CNN mampu menangkap pola berbasis karakter secara lebih fleksibel, tetapi masih memiliki keterbatasan dalam memahami konteks yang lebih kompleks [3]. Di sisi lain, TinyBERT menunjukkan keunggulan dalam memahami hubungan antar token melalui mekanisme attention, sehingga mampu mengenali pola phishing yang lebih kompleks dan tersembunyi [5], [14]. Hal ini menunjukkan bahwa penggunaan Transformer memberikan peningkatan signifikan dalam tugas klasifikasi URL dibandingkan metode tradisional.

4.2 Perbandingan Kinerja Baseline

Model baseline menunjukkan performa kompetitif. Random Forest berbasis fitur leksikal menghasilkan precision phishing tertinggi (0,9661) dan false positive terendah [13], yang menunjukkan bahwa model ini efektif meminimalkan false alarm. Char-CNN memiliki performa yang hampir sama dengan Random Forest, dengan precision1 0,9638 dan recall1 0,8847. Hasil ini menunjukkan bahwa pendekatan berbasis fitur sederhana maupun deep learning karakter tetap mampu menangkap pola URL phishing secara efektif. SVM berbasis character n-gram memberikan hasil paling rendah dibanding baseline lain pada metrik phishing, khususnya F1_1 sebesar 0,9116 dan FN sebesar 54. Hal ini mengindikasikan bahwa SVM cenderung kurang optimal pada kasus phishing yang bentuknya lebih bervariasi dan kompleks.

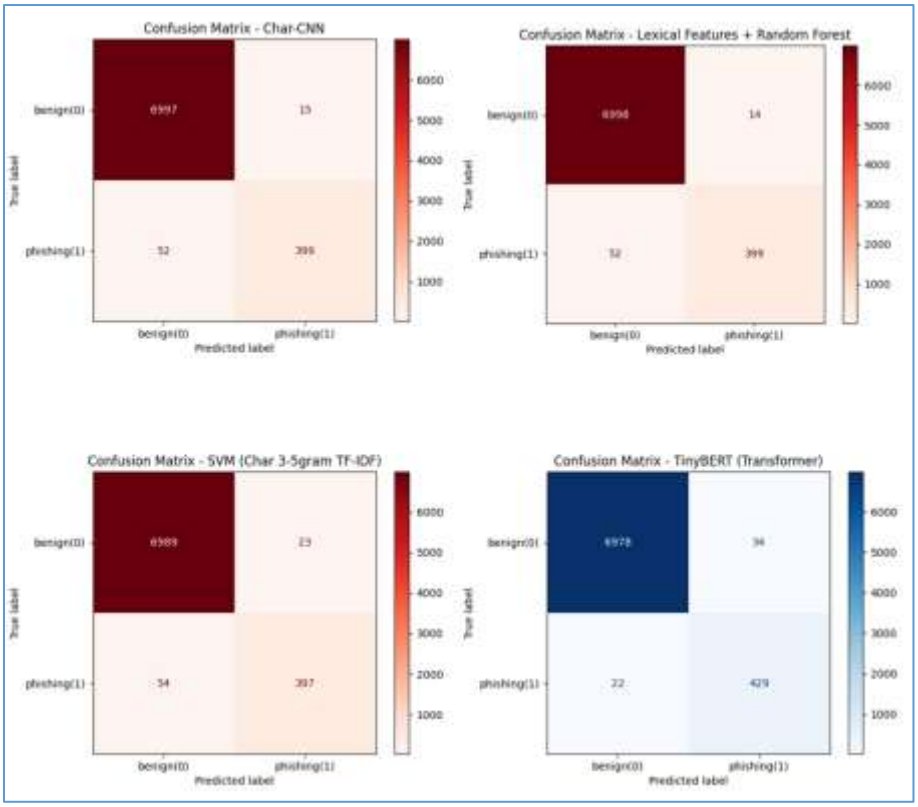
4.3 Kinerja TinyBERT

TinyBERT menghasilkan performa terbaik dengan accuracy tertinggi (0,9925) dan recall phishing tertinggi (0,9512). Nilai recall yang tinggi menunjukkan bahwa model TinyBERT lebih sensitif dalam mendeteksi phishing dibanding baseline. Keunggulan utama TinyBERT terlihat pada jumlah false negative (FN) yang paling rendah, yaitu 22. Dalam konteks keamanan siber, FN merupakan kesalahan yang paling kritis karena URL phishing dapat lolos dari sistem dan berpotensi menyebabkan pengguna mengakses tautan berbahaya. Oleh karena itu, kemampuan TinyBERT dalam menurunkan FN menjadi keunggulan penting dibandingkan baseline, meskipun precision1 lebih rendah dibanding Random Forest dan Char-CNN.

4.4 Analisis *False Positive dan False Negative*

Pada sistem deteksi phishing, kesalahan prediksi tidak hanya dinilai dari akurasi, tetapi juga dari jenis kesalahan yang dihasilkan model. Oleh karena itu, penelitian ini melakukan analisis menggunakan confusion matrix untuk mengidentifikasi False Positive (FP) dan False Negative (FN).

 ISSN:2302-8149
e-ISSN:2540-9719



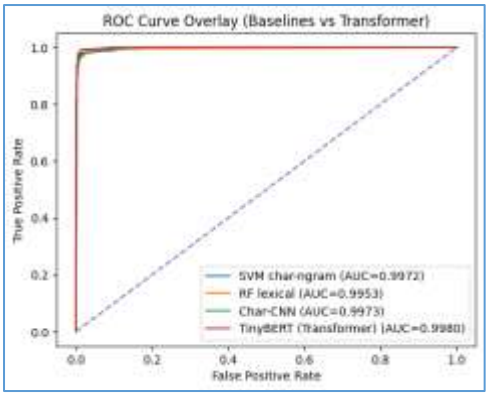
Gambar 2 Confusion matrix

Berdasarkan hasil evaluasi pada Gambar 2, baseline Random Forest dan Char-CNN menghasilkan false positive yang lebih rendah (masing-masing FP=14 dan FP=15), sehingga lebih konservatif dalam mendeteksi phishing. Namun, kedua model tersebut menghasilkan false negative yang cukup tinggi (FN=52), yang menunjukkan masih banyak URL phishing yang luput terdeteksi. Sebaliknya, TinyBERT menghasilkan false negative paling rendah (FN=22), yang berarti model ini lebih efektif dalam menangkap URL phishing dan mengurangi risiko serangan yang lolos sebagai benign. Walaupun TinyBERT memiliki FP lebih tinggi (FP=34), peningkatan false alarm ini masih dapat ditoleransi dalam konteks keamanan siber karena dapat ditangani melalui verifikasi tambahan atau mekanisme pengecekan ulang. Dengan demikian, TinyBERT lebih sesuai untuk skenario keamanan yang memprioritaskan minimalisasi false negative.

4.5 Hasil Evaluasi ROC dan Precision-Recall Curve

ROC curve menunjukkan hubungan antara True Positive Rate (TPR) dan False Positive Rate (FPR). Pada penelitian ini, ROC curve digunakan untuk menilai kemampuan model dalam membedakan kelas phishing dan non-phishing secara global. Nilai AUC (Area Under Curve) digunakan sebagai indikator ringkas performa model, di mana nilai yang mendekati 1 menunjukkan kemampuan klasifikasi yang semakin baik. Seperti di Gambar 3, SVM char-gram (AUC=0,9972), RF lexical (AUC=0.9953), Char-CNN (AUC=0,9973), TinyBERT (Tansformer) (AUC=0,9980).

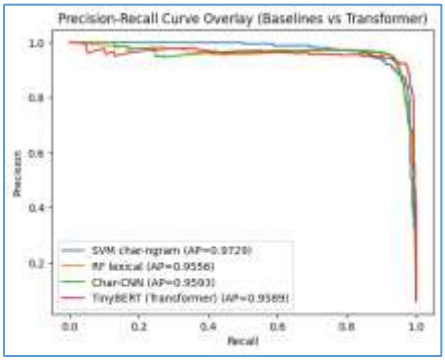
 ISSN:2302-8149
e-ISSN:2540-9719



Gambar 3 ROC curve overlay

Berdasarkan ROC overlay, TinyBERT menunjukkan performa yang lebih stabil dibanding baseline, lihat Gambar 3. Terutama dalam mempertahankan TPR yang tinggi dengan peningkatan FPR yang relatif terkendali. Hasil ini menunjukkan TinyBERT lebih konsisten dalam mendeteksi phishing pada berbagai threshold dibandingkan pendekatan tradisional.

Precision–Recall (PR) curve digunakan untuk menganalisis hubungan antara precision dan recall pada berbagai threshold. PR curve penting digunakan karena dataset pada penelitian ini bersifat tidak seimbang (imbalanced), sehingga metrik PR lebih representatif untuk menilai performa pada kelas minoritas, yaitu phishing. Hasil PR curve pada Gambar 4, overlay menunjukkan bahwa TinyBERT memiliki kecenderungan recall yang lebih tinggi dibanding baseline, yang berarti model lebih sensitif dalam mendeteksi URL phishing. Selain itu, nilai Average Precision (AP) digunakan sebagai ringkasan performa PR curve, di mana nilai yang lebih tinggi menunjukkan performa yang lebih baik terhadap kelas positif.



Gambar 4 Precision-recall curve overlay

Precision–Recall (PR) curve digunakan untuk menganalisis hubungan antara precision dan recall pada berbagai threshold. PR curve penting digunakan karena dataset pada penelitian ini bersifat tidak seimbang (imbalanced), sehingga metrik PR lebih representatif untuk menilai performa pada kelas minoritas, yaitu phishing. Hasil PR curve pada Gambar 4, overlay menunjukkan bahwa TinyBERT memiliki kecenderungan recall yang lebih tinggi dibanding baseline, yang berarti model lebih sensitif dalam mendeteksi URL phishing. Selain itu, nilai Average Precision (AP) digunakan sebagai ringkasan performa PR curve, di mana nilai yang lebih tinggi menunjukkan performa yang lebih baik terhadap kelas positif.

Berdasarkan hasil pengujian, seluruh model menunjukkan performa yang tinggi dengan akurasi di atas 0,98. Namun, perbedaan performa lebih terlihat pada metrik kelas phishing (label 1), terutama recall dan jumlah false negative (FN), yang menjadi aspek penting dalam konteks keamanan siber. Hal ini sejalan dengan kajian terbaru yang menekankan bahwa pada kasus phishing dengan data tidak seimbang, evaluasi tidak cukup hanya mengandalkan akurasi, melainkan perlu mempertimbangkan kemampuan model dalam mendeteksi kelas minoritas secara tepat [13], [14].

Model baseline seperti Random Forest berbasis fitur leksikal dan Char-CNN menghasilkan precision lebih tinggi serta false positive (FP) yang lebih rendah sehingga lebih konservatif dan minim false alarm, sesuai dengan temuan bahwa fitur URL sederhana dan pembelajaran karakter efektif sebagai pendekatan deteksi phishing [12], [13]. Namun, baseline menghasilkan FN yang lebih besar

 ISSN:2302-8149
e-ISSN:2540-9719

sehingga lebih banyak phishing yang lolos. Sebaliknya, TinyBERT memiliki recall phishing tertinggi dan FN terendah, menunjukkan kemampuan yang lebih baik dalam menangkap variasi URL phishing yang kompleks, sejalan dengan penelitian TinyBERT sebagai Transformer ringan yang tetap kompetitif dan efektif dalam klasifikasi [1], serta studi Transformer untuk malicious URL detection yang menunjukkan keunggulan attention dalam mengenali pola yang lebih kaya [2], [9]. Evaluasi ROC dan Precision–Recall juga mendukung stabilitas TinyBERT pada berbagai threshold, khususnya pada kondisi data imbalanced yang lebih representatif dianalisis melalui PR curve [11], [14]. Hasil penelitian ini memiliki implikasi penting dalam pengembangan sistem keamanan siber, khususnya pada deteksi phishing secara real-time. Model TinyBERT dapat diintegrasikan ke dalam sistem keamanan seperti browser extension, email filtering, atau sistem deteksi berbasis server. Selain itu , pendekatan ini juga dapat dikembangkan lebih lanjut untuk mendeteksi jenis serangan lain seperti malware distribution URL atau spam URL, sebagaimana ditunjukkan dalam penelitian terkait deteksi malicious URL berbasis deep learning [7], [8].

5 Kesimpulan

Berdasarkan hasil penelitian, dapat disimpulkan bahwa seluruh model mampu melakukan klasifikasi URL phishing dengan performa tinggi, namun TinyBERT memberikan hasil terbaik dalam mendeteksi URL phishing. TinyBERT mencapai akurasi 0,9925 serta menghasilkan recall phishing tertinggi (0,9512) dan false negative terendah (22), sehingga lebih efektif dalam menekan risiko URL phishing yang lolos sebagai non-phishing. Selain itu, penelitian ini menunjukkan bahwa pendekatan berbasis Transformer tidak hanya meningkatkan akurasi, tetapi juga memberikan keseimbangan yang lebih baik antara precision dan recall pada kondisi dataset tidak seimbang [11], [14]. Temuan ini sejalan dengan penelitian yang menyatakan bahwa pendekatan Transformer mampu menangkap pola yang lebih kompleks dibanding metode tradisional, dan TinyBERT tetap efisien karena merupakan model hasil distilasi yang ringan [1], [2], [9]. Sementara itu, Random Forest dan Char-CNN menghasilkan false positive lebih rendah, sehingga lebih baik dalam meminimalkan false alarm, namun berisiko lebih tinggi meloloskan phishing karena false negative yang lebih besar [12], [13]. Kondisi tersebut menunjukkan adanya trade-off antara kemampuan mendeteksi phishing dan kemampuan menekan kesalahan klasifikasi pada URL normal. Dengan demikian, TinyBERT dinilai sebagai model yang potensial untuk diterapkan pada sistem keamanan nyata yang memprioritaskan deteksi phishing secara maksimal. Meskipun demikian, penelitian ini masih memiliki beberapa keterbatasan, antara lain penggunaan dataset dari satu sumber, deteksi yang hanya berbasis URL tanpa mempertimbangkan konten halaman web, serta pengujian yang masih dilakukan secara offline. Sebagai saran, penelitian selanjutnya dapat mengembangkan strategi untuk menyeimbangkan trade-off antara false positive dan false negative. Salah satu pendekatan yang dapat dilakukan adalah threshold tuning agar sistem dapat disesuaikan dengan kebutuhan implementasi, misalnya meningkatkan recall untuk skenario keamanan tinggi atau menekan false alarm untuk kebutuhan operasional [14]. Selain itu, penelitian lanjutan juga dapat menerapkan model hybrid atau ensemble, misalnya menggabungkan fitur leksikal dengan Transformer, karena beberapa studi menunjukkan kombinasi pendekatan dapat meningkatkan stabilitas performa dan generalisasi model pada variasi URL phishing yang lebih luas [9], [10]. Terakhir, pengujian pada dataset yang lebih beragam dan skenario real-time disarankan untuk mengevaluasi konsistensi model pada kondisi implementasi nyata [13], [14].

Referensi

[1] X. Jiao *et al.*, "*TinyBERT : Distilling BERT for Natural Language Understanding*," *Find. ofthe Assoc. Comput. Linguist. EMNLP 2020*, pp. 4163–4174, 2020, DOI: 10.18653/v1/2020.findings-emnlp.372.

[2] R. Liu, Y. Wang, Z. Guo, H. Xu, and Z. Qin, "*Transurl: Improving Malicious URL Detection with Multi-Layer Transformer Encoding and Multi-Scale Pyramid Features*", DOI: 10.1016/j.comnet.2024.110707.

[3] N. Q. Do, A. N. Selamat, H. Fujita, and O. Krejcar, "*An Integrated Model based on Deep Learning Classifiers and Pre-Trained Transformer for Phishing URL Detection*," Vol. 161, No. December, 2024, DOI: doi.Org/10.1016/j.future.2024.06.031.

 ISSN:2302-8149
e-ISSN:2540-9719

[4] A. Selman, F. Coskun, and M. Aydos, "*Computers & Security GramBeddings : A New Neural Network for URL based Identification of Phishing Web Pages Through N-Gram Embe ddings*," *Comput. Secur.*, Vol. 124, p. 102964, 2023, DOI: 10.1016/j.cose.2022.102964.

[5] Q. Emad, M. H. Faheem, and I. Ahmad, "*Detecting Phishing URLs based on a Deep Learning Approach to Prevent Cyber-Attacks Fourth Quarter Quarter 2023 Stats Fourth*," 2024, DOI: 10.3390/app142210086.

[6] A. Ozcan, C. Catal, E. Donmez, and B. Senturk, "*A Hybrid DNN – LSTM Model for Detecting Phishing URLs*," *Neural Comput. Appl.*, Vol. 35, No. 7, pp. 4957–4973, 2023, DOI: 10.1007/s00521-021-06401-z.

[7] S. Srinivasan, R. Vinayakumar, A. Arunachalam, M. Alazab, and S. Kp, "*Malicious URL Detection using Deep Learning*," *Prepr. Submitt. to Elsevier*, pp. 1–19, 2021.

[8] X. Xiao *et al.*, "*Phishing Websites Detection via CNN and Multi-Head Self-Attention on Imbalanced Datasets R*," *Comput. Secur.*, Vol. 108, p. 102372, 2021, DOI: 10.1016/j.cose.2021.102372.

[9] A. Safi and S. Singh, "*A Systematic Literature Review on Phishing Website Detection Techniques*," *J. King Saud Univ. - Comput. Inf. SCI.*, Vol. 35, No. 2, pp. 590–611, 2023, DOI: 10.1016/j.jksuci.2023.01.004.

[10] M. Alshehri, A. Abugabah, A. Algarni, and S. Almotairi, "*Character-Level Word Encoding Deep Learning Model for Combating Cyber Threats in Phishing URL Detection*," *Comput. Electr. Eng.*, Vol. 100, No. March, p. 107868, 2022, DOI: 10.1016/j.compeleceng.2022.107868.

[11] C. Wang and Y. Chen, "*Knowledge-based Systems TCURL : Exploring Hybrid Transformer and Convolutional Neural Network on Phishing URL Detection*," *Knowledge-based Syst.*, Vol. 258, p. 109955, 2022, DOI: 10.1016/j.knosys.2022.109955.

[12] Z. Chen, Y. Liu, C. Chen, M. Lu, and X. Zhang, "*Malicious URL Detection based on Improved Multilayer Recurrent Convolutional Neural Network Model*," *Secur. Commun. Networks*, Vol. 2021, 2021, DOI: 10.1155/2021/9994127.

[13] A. Safi and S. Singh, "*A systematic literature review on phishing website detection Techniques*," *J. King Saud Univ. - Comput. Inf. SCI.*, Vol. 35, No. 2, pp. 590–611, 2023, DOI: 10.1016/j.jksuci.2023.01.004.

[14] S. K. D. Sumathi, "*Staying Ahead of Phishers : A Review of Recent Advances and Emerging Methodologies in Phishing Detection*," *Artif. Intell. Rev.*, 2025, DOI: 10.1007/s10462-024-11055-z.

[15] H. Pippalla, "*Malicious URLs Dataset (40k Samples)*," Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/himadri07/malicious-urls-dataset-15k-rows

.