JURNAL FASILKOM
Volume 15 No. 3 | Desember 2025: 648-654

P-ISSN : 2089-3353
E-ISSN : 2808-9162

Klasifikasi Algoritma Kriptografi pada Pesan Terenkripsi menggunakan *Support Vector Machine* (SVM)

Yulia Fatma[1*], Rahmad Gunawan[2], Nurkhairi Fitri[3], Rahmad Firdaus[4], Regiolina Hayami[5], Soni[6]
[123456]Teknik Informatika, Fakultas Ilmu Komputer, Universitas Muhammadiyah Riau, Riau, Indonesia
[1]yuliafatma@umri.ac.id[*], [2]goengoen78@umri.ac.id, [3]nurkhairiftr@gmail.com, [4]rahmadfirdaus@umri.ac.id, [5]regiolinahayami@umri.ac.id, [6]soni@umri.ac.id

Abstract

Data protection has become a highly critical aspect, particularly in addressing ransomware threats that illegally encrypt data. This study is important to evaluate the capability of machine learning techniques in identifying encryption algorithms used in encrypted data, especially in ransomware attacks. This work represents an initial step that can assist cybersecurity practitioners in more rapidly understanding attack patterns, determining appropriate response strategies, and enhancing proactive mitigation and response efforts to protect data against increasingly complex cyber threats. The machine learning algorithm employed in this study is the Support Vector Machine (SVM). The dataset consists of ciphertext generated using the AES, DES, and Vigenère Cipher cryptographic algorithms. The feature extraction process utilizes ten statistical features to capture the distinctive patterns of each type of ciphertext. The SVM model is developed using a data split of 90% for training and 10% for testing. Performance evaluation is conducted using a confusion matrix with accuracy, precision, recall, and F1-score metrics. The result demonstrate an average accuracy 0f 92,33%, with the vigenere cipher being perfectly classified (100% accuracy). Howefer, slight misclassifications occured beetween AES and DES duet o their similiar entropy chraracteristic. Experimental results demonstrate that the SVM model is capable of identifying encryption algorithms with high accuracy and balanced classification performance across the three algorithm classes. These findings highlight the potential of machine learning approaches for detecting encryption algorithms in cyber-attacks, thereby making a meaningful contribution to the improvement of proactive data security mitigation and response strategies.

Keywords: cryptanalysis, support vector machine (SVM), AES, DES, Vigenère

Abstrak

Perlindungan data menjadi aspek yang sangat krusial, terutama dalam menghadapi ancaman *ransomware* yang mengenkripsi data secara ilegal. Penelitian ini penting dilakukan untuk mengukur kemampuan *machine learning* dalam mengidentifikasi algoritma enkripsi yang digunakan pada data terenkripsi, khususnya dalam serangan *ransomware*. Ini merupakan langkah awal yang dapat membantu pihak keamanan siber memahami pola serangan lebih cepat, menentukan strategi penanganan yang tepat, serta meningkatkan upaya mitigasi dan respons proaktif dalam melindungi data dari ancaman siber yang semakin kompleks. Algoritma *machine learning* yang digunakan adalah *Support Vector Machine* (SVM). *Dataset* terdiri dari *ciphertext* algoritma kriptografi AES, DES, dan *Vigenère Cipher*. Proses ekstraksi fitur memanfaatkan sepuluh fitur statistik untuk menangkap pola unik dari setiap jenis *ciphertext*. Model SVM dibangun dengan skema pembagian data sebesar 90% untuk pelatihan dan 10% untuk pengujian. Evaluasi kinerja dilakukan menggunakan *confusion matrix* dengan metrik akurasi, presisi, *recall*, dan F1-*score*. Hasil pengujian menunjukkan bahwa model SVM berhasil mencapai akurasi rata-rata sebesar 92,33%. Secara spesifik, klasifikasi pada algoritma *vigenere* mencapai akurasi sempurna 100%. Sementara pada algoritma modern (*AES* dan *DES*) terdapat margin kesalahan tipis karena kemiripan karakteristik entropi pada hasil enskripsinya. Hasil eksperimen menunjukkan bahwa model SVM mampu mengidentifikasi algoritma enkripsi dengan tingkat akurasi yang tinggi serta kinerja klasifikasi yang seimbang di antara ketiga kelas algoritma. Temuan ini menunjukkan potensi penggunaan *machine learning* untuk mendeteksi algoritma enkripsi dalam serangan siber, sehingga dapat memberikan kontribusi nyata dalam meningkatkan strategi mitigasi dan *respons* keamanan data secara proaktif.

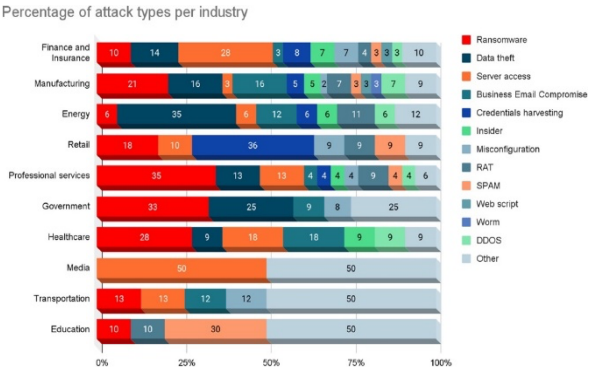
Kata kunci: kriptanalisis, *support vector machine* (SVM), AES, DES, *Vigenère*

1. Pendahuluan

Seiring dengan pesatnya perkembangan teknologi digital, keamanan siber telah menjadi prioritas strategis bagi berbagai organisasi di seluruh dunia. Di antara beragam ancaman siber yang mendapat perhatian global, serangan *ransomware* menonjol karena dampaknya yang luas serta potensi gangguan yang

P-ISSN : 2089-3353
E-ISSN : 2808-9162

signifikan. *Ransomware* merupakan jenis perangkat lunak berbahaya (*malicious software*) yang secara khusus dirancang untuk mengenkripsi atau membatasi akses terhadap data dan sistem komputer milik korban. Setelah proses enkripsi berhasil dilakukan, pelaku umumnya menuntut pembayaran tebusan dalam bentuk mata uang kripto sebagai syarat untuk memulihkan akses terhadap data tersebut. Salah satu insiden ransomware terbesar di Indonesia terjadi ketika Pusat Data Nasional Sementara (PDNS) mengalami kompromi keamanan, yang mengakibatkan terganggunya layanan publik dan keimigrasian [1]. Peristiwa ini menegaskan urgensi pengembangan metode untuk mengidentifikasi kunci kriptografi atau algoritma enkripsi yang digunakan dalam serangan *ransomware*. Dampak kebocoran data tidak hanya terbatas pada kerugian finansial, tetapi juga menimbulkan ancaman serius terhadap kepercayaan publik serta kerahasiaan informasi sensitif. Ransomware pertama kali diidentifikasi pada 12 Mei 2017 dan dengan cepat menyebar, menginfeksi lebih dari 200.000 komputer di lebih dari 150 negara [2]. Sebagaimana ditunjukkan pada Gambar 1, pada sektor *Professional Services*, Pemerintahan, dan Kesehatan, persentase serangan ransomware terhadap keseluruhan serangan siber masing-masing mencapai 35%, 33%, dan 28%, menjadikannya sebagai jenis serangan yang paling dominan secara keseluruhan.



Gambar 1. Persentase Tipe Serangan di Industri [3]

Kriptografi merupakan bidang ilmu yang mempelajari teknik untuk menjamin kerahasiaan dan/atau keaslian informasi. Penelitian dalam kriptografi pada umumnya berfokus pada dua area utama, yaitu perancangan kriptografi (*cryptographic design*) dan kriptanalisis (*cryptanalysis*) [3]. Kriptanalisis, yang sering disebut sebagai *code-breaking*, mencakup berbagai teknik untuk menguraikan informasi yang telah dienkripsi. Secara khusus, kriptanalisis merupakan teknik yang digunakan untuk memecahkan pesan terenkripsi tanpa memiliki pengetahuan mengenai proses enkripsi yang digunakan [4]. Teknik kriptanalisis konvensional tidak lagi efektif terhadap algoritma-algoritma kriptografi modern [5]. Metode kriptanalisis tradisional umumnya membutuhkan waktu dan sumber daya yang besar sehingga kurang kompatibel dengan algoritma baru, sehingga diperlukan pendekatan kriptanalisis yang lebih mutakhir [6]. Perkembangan teknologi *Machine Learning* (ML) membuka arah baru dalam bidang kriptografi dan kriptanalisis [7]. Keterkaitan antara kriptografi dan ML pertama kali diperkenalkan pada tahun 1991 [8]. Sejak saat itu, banyak peneliti mengeksplorasi penggunaan teknik ML untuk melakukan kriptanalisis terhadap *block cipher*. Sejumlah penelitian ML di bidang kriptanalisis dilakukan untuk menebak atau mengidentifikasi algoritma kriptografi yang digunakan, memperkirakan S-Box, menemukan kunci rahasia, hingga memulihkan pesan asli (*plaintext*) [6],[9],[10]. Berdasarkan penelitian-penelitian sebelumnya, kemampuan kecerdasan buatan, khususnya *deep learning* dalam kriptanalisis, dinilai cukup efektif terhadap algoritma yang ada [6], [5], [11].

Teknik kriptanalisis tradisional seperti *brute-force* attack, analisis frekuensi, dan *known-plaintext attack* terbukti efektif terutama terhadap algoritma kriptografi klasik seperti Caesar *Cipher*. Namun, pendekatan tersebut tidak lagi memadai untuk menghadapi algoritma kriptografi *modern* yang secara khusus dirancang untuk bertahan dari serangan-serangan tersebut [12]. Penelitian yang melibatkan kecerdasan buatan dalam kriptanalisis, khususnya penggunaan *artificial neural network* untuk memulihkan *plaintext* dari data yang dienkripsi menggunakan DES, menunjukkan potensi besar dalam meningkatkan efektivitas serangan dan menjadi arah penelitian yang menjanjikan dalam keamanan data di masa depan [13]. Krishna [14] melakukan penelitian yang bertujuan untuk mengidentifikasi jenis *ciphertext* klasik menggunakan dua pendekatan utama. Pendekatan pertama menggunakan klasifikasi langsung dengan algoritma *Support Vector Machine* (SVM) tanpa menambahkan fitur khusus dari *ciphertext*. Meskipun *ciphertext* diasumsikan tidak memiliki struktur bermakna, pendekatan ini menghasilkan tingkat akurasi yang cukup tinggi. Pendekatan kedua memanfaatkan *Hidden Markov Model* (HMM) untuk menangkap pola statistik tersembunyi antar karakter, yang selanjutnya diproses menggunakan *Convolutional Neural Networks* (CNN) dan SVM. Hasil penelitian menunjukkan bahwa SVM memiliki kinerja paling unggul dalam mengklasifikasikan *ciphertext* klasik dengan rata-rata akurasi mencapai 99%. Temuan ini mengonfirmasi bahwa klasifikasi berbasis SVM merupakan metode yang sangat efektif untuk mendeteksi dan mengidentifikasi skema enkripsi yang digunakan dalam serangan *ransomware*. Oleh karena itu, pengembangan model berbasis SVM untuk mengidentifikasi algoritma enkripsi pada serangan tersebut menjadi aspek penting dalam penelitian ini.

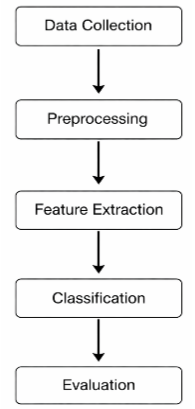
Penelitian kriptanalisis lainnya menerapkan teknik ekstraksi fitur pada *ciphertext* [15] dan menunjukkan bahwa pendekatan machine learning berbasis rekayasa fitur sangat efektif dalam mengidentifikasi jenis cipher klasik hanya berdasarkan *ciphertext*. Dengan merancang dan memanfaatkan fitur-fitur statistik yang merepresentasikan pola unik dari berbagai cipher, pendekatan ini berhasil mencapai akurasi klasifikasi

P-ISSN : 2089-3353
E-ISSN : 2808-9162

hingga 80,24%, yang secara signifikan melampaui metode sebelumnya yang tidak menggunakan fitur eksplisit. Namun demikian, sebagian besar penelitian yang ada masih berfokus pada algoritma kriptografi klasik. Oleh karena itu, penelitian ini bertujuan untuk mengeksplorasi dan mengevaluasi efektivitas teknik ekstraksi fitur dalam kriptanalisis algoritma kriptografi modern. Secara khusus, penelitian ini mengkaji apakah pendekatan machine learning yang diperkaya dengan fitur-fitur statistik yang diekstraksi dari ciphertext dapat diterapkan secara efektif untuk mengidentifikasi, memecahkan, atau menganalisis algoritma enkripsi modern yang dirancang untuk menahan serangan kriptanalisis tradisional. Hasil dari penelitian ini diharapkan dapat berkontribusi terhadap pengembangan metode kriptanalisis yang lebih adaptif dan akurat dalam menghadapi kompleksitas keamanan informasi digital yang terus meningkat.

2. Metode Penelitian

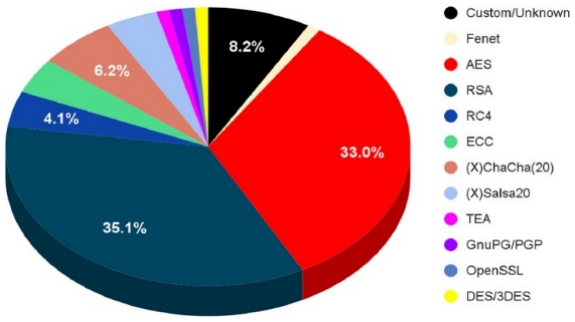
Tahapan pada penelitian ini diawali dengan pengumpulan data publik berupa *plaintext*, tahap *preprocessing* yang dilakukan untuk memastikan bahwa data siap untuk diolah, ekstraksi fitur untuk mendapatkan informasi dan pola unik dari data, proses klasifikasi untuk mempelajari pola data dan evaluasi untuk mengukur prediksi kelas dari hasil pembelajaran model. Tahapan ini dipaparkan pada Gambar 2.



Gambar 2. Metode Penelitian

2.1. *Data Collection*

Dataset yang digunakan dalam penelitian ini terdiri dari 1.000 entri *plaintext* yang diambil dari penelitian sebelumnya, yang menggunakan algoritma Data *Encryption* Standard (DES) dan *Vigenère Cipher*. Penulis memastikan bahwa setiap kelas memiliki jumlah sampel data yang sama, sehingga menghasilkan dataset yang seimbang. Pada tahap ini, penulis menyiapkan tiga kelompok data yang diproses menggunakan algoritma DES, *Vigenère Cipher,* dan *Advanced Encryption Standard* (AES). Penambahan algoritma AES didasarkan pada laporan distribusi berbagai algoritma enkripsi yang digunakan dalam serangan *ransomware* sejak insiden pertama yang tercatat pada tahun 1989 hingga akhir tahun 2021, sebagaimana ditunjukkan pada Gambar 3. Berdasarkan laporan tersebut, AES merupakan algoritma enkripsi yang paling sering digunakan dalam serangan *ransomware*.



Gambar 3. Algoritma Enkripsi Yang Digunakan Pada Kasus Ransomware 1989 – 2021 [16]

2.2. *Preprocessing*

Tahap pra-pemrosesan dilakukan untuk memastikan bahwa dataset siap digunakan pada tahap analisis selanjutnya. Langkah-langkah pra-pemrosesan meliputi penghapusan karakter non-alfabet untuk menstandarkan analisis, konversi seluruh huruf menjadi huruf kapital guna menghindari perbedaan akibat sensitivitas huruf besar dan kecil, serta normalisasi panjang *ciphertext* untuk memastikan keseimbangan pada saat proses ekstraksi fitur. Selanjutnya, data yang telah dibersihkan dan disebut sebagai *plaintext* dienkripsi menggunakan tiga algoritma yang berbeda, yaitu AES, DES, dan *Vigenère Cipher.* Keluaran dari proses enkripsi ini disebut sebagai *ciphertext*, yang tidak dapat dipahami tanpa proses dekripsi atau pengetahuan terhadap kunci yang sesuai. Setiap *ciphertext* yang dihasilkan memiliki karakteristik yang berbeda-beda, bergantung pada metode enkripsi yang digunakan.

2.3. *Feature Extraction*

Setelah tahap pra-pemrosesan selesai, dilakukan proses ekstraksi fitur statistik untuk mengidentifikasi pola unik dari setiap *ciphertext* yang dihasilkan oleh algoritma AES, DES, dan *Vigenère Cipher*. Proses ekstraksi ini bertujuan untuk menangkap karakteristik khas dari masing-masing *ciphertext* dengan memanfaatkan sejumlah fitur statistik utama, antara lain SDD (*Sequential Digraph Distribution*), DIC (*Digraph Index of Coincidence*), MIC (*Monograph Index of Coincidence)*, dan MKA (*Mean Kappa Analysis*), yang digunakan untuk mengukur distribusi karakter dan tingkat kebetulan (*coincidence*). Selain itu, fitur ROD (*Rate of Digraphs*), LDI (*Letter Distribution Index*), dan LR (*Letter Repetition*) digunakan untuk menilai tingkat pengulangan dan pola distribusi karakter dalam *ciphertext*. Fitur tambahan seperti SHAN (*Shannon Entropy*), IoC (*Index of Coincidence*), dan REP (*Repetition Factor*) digunakan untuk menghitung karakteristik entropi dan pengulangan, sehingga memungkinkan pendeteksian

P-ISSN : 2089-3353
E-ISSN : 2808-9162

pola khas yang melekat pada masing-masing algoritma enkripsi [15], masing-masing fitur dijelaskan pada Tabel 1.

Tabel 1. Fitur Yang Digunakan

No.	Fitur	Deskripsi
1	*Single letter-digraph discrepancy score* (SDD)	Fitur ini menggunakan tabel perbedaan antara unigram dan bigram. Skor dihitung dengan menambahkan setiap nilai pada posisi huruf pertama dalam alfabet dikali 26 ditambah posisi huruf kedua dalam alfabet dari tabel SDD. Skor kemudian dibagi dengan panjang teks dikurangi 1. Untuk normalisasi, skor dibagi dengan 10.
2	*Digraphic index of coincidence* (DIC)	Jumlah semua probabilitas kemunculan dua pasangan karakter identik dalam sebuah teks dikalikan 1000.
3	*Max IC for periods 1-15* (MIC)	Teks dibagi menjadi periode 1-15. IoC tertinggi dari semua subkelompok dihitung dengan membagi teks menjadi p kelompok. Setiap kelompok terdiri dari semua karakter yang diberi jarak p. Jika p = 3 ada 3 kelompok, dengan demikian kelompok pertama berisi setiap karakter ketiga yang dimulai dengan 0; kelompok kedua setiap karakter ketiga yang dimulai dengan 1 dan kelompok ketiga setiap karakter ketiga yang dimulai dengan 2. Nilai tertinggi dikembalikan.
4	*Max kappa for periods 1-15* (MKA)	Teks digeser p ke kanan untuk Periode 1-15. Karakter p yang tersisa diisi dengan nilai yang tidak terdapat dalam teks (misalnya -1). Hasil statistik ini adalah persentase kecocokan maksimum antara teks yang dipindahkan dan teks asli.
5	*Percentage of odd-spaced repeats* (ROD)	Persentase karakter berulang dengan spasi ganjil terhadap jumlah karakter berulang. Untuk tujuan ini, semua karakter yang sama dihitung untuk setiap karakter dari posisi +1. Hasilnya adalah sum odd /sumal.
6	*Log digraph score* (LDI)	Bigram dalam teks dicari dalam daftar frekuensi huruf bahasa Inggris yang telah dihitung sebelumnya dan dijumlahkan. Rata-rata dari jumlah ini adalah skornya. Di Bion, angka riil digunakan sebagai gantinya, tetapi ini adalah nilai yang terlalu besar, itulah sebabnya probabilitas kejadian dibagi 10 lebih cocok.
7	*Long Repeat* (LR)	Persentase karakter yang diulang tepat tiga kali. Untuk tujuan ini, semua karakter yang sama dihitung untuk setiap karakter dari posisi +1. Akar hasil ini dibagi dengan panjang teks.
8	*Shannon's Entropy Equation* (SHAN)	Entropi adalah ukuran untuk menentukan konten informasi suatu teks. Pada dasarnya, entropi yang lebih tinggi menunjukkan bahwa data dienkripsi, nilai ini dibagi 10.
9	*Index of Coincidence* (IoC)	Jumlah semua kemungkinan munculnya dua karakter identik dalam sebuah teks.
10	*Repetition Feature* (REP)	Fitur ini terdiri dari jumlah karakter identik yang muncul tepat n kali untuk $2 \leq n \leq 5$. Normalisasi dihitung dengan membagi jumlah total pengulangan.

2.4. *Classification*

Keberhasilan proses klasifikasi diukur berdasarkan tingkat ketepatan model dalam mengelompokkan data ke dalam kategori yang benar [17]. Dalam penelitian ini, proses klasifikasi dilakukan menggunakan algoritma *Support Vector Machine* (SVM) karena kemampuannya dalam menangani data berdimensi tinggi serta menghasilkan kinerja optimal pada permasalahan klasifikasi nonlinier. Keunggulan utama SVM terletak pada performanya yang kuat terhadap dataset berdimensi tinggi serta fleksibilitasnya dalam mengelola volume data yang kecil maupun besar [18]. Implementasi SVM melibatkan beberapa tahapan, yaitu pemilihan kernel, pelatihan model, dan validasi model. Model SVM dikembangkan menggunakan pustaka *Scikit-Learn* pada platform *Google Colab* dengan menerapkan kernel linear dan parameter regularisasi C = 1.0. Dataset dibagi menjadi 90% data pelatihan dan 10% data pengujian. Kinerja beberapa jenis kernel, seperti *linear* dan *radial basis function* (RBF), dievaluasi untuk menentukan konfigurasi yang paling sesuai. Selanjutnya, model dilatih menggunakan dataset yang telah melalui tahap pra-pemrosesan dan ekstraksi fitur. Model kemudian divalidasi menggunakan metode *cross-validation* untuk memastikan konsistensi hasil dan mencegah terjadinya *overfitting*. Pendekatan ini memungkinkan evaluasi yang lebih robust terhadap kemampuan generalisasi model pada berbagai pembagian data.

2.5. *Evaluation*

Evaluasi model dilakukan untuk menilai kinerja klasifikasi algoritma *Support Vector Machine* (SVM) dalam mengidentifikasi algoritma kriptografi yang digunakan pada *ciphertext*. Metode evaluasi yang digunakan adalah *confusion matrix*, yang mencakup pengukuran akurasi, presisi, *recall*, dan *F1-score*. *Confusion matrix* digunakan untuk menganalisis distribusi prediksi yang benar dan salah. Akurasi mengukur tingkat keberhasilan keseluruhan dari proses klasifikasi. Presisi mengevaluasi kemampuan model dalam mengklasifikasikan *ciphertext* secara benar, sedangkan *recall* mengukur sensitivitas model dalam mendeteksi setiap kelas algoritma. *F1-score* mengombinasikan nilai presisi dan recall ke dalam satu metrik tunggal untuk memberikan gambaran komprehensif mengenai kinerja model [19].

3. Hasil dan Pembahasan

Bagian ini menyajikan hasil dari proses klasifikasi *ciphertext* yang dilakukan menggunakan algoritma *Support Vector Machine* (SVM). Contoh *ciphertext* yang dihasilkan oleh algoritma kriptografi AES, DES, dan Vigenère ditampilkan pada Tabel 2.

Tabel 2. Potongan *Ciphertext* yang digunakan sebagai dataset

Algoritma	Ciphertext
AES	JiX5CNhFCDwShAOAaWzWFNJh7YPVdN MwB4VLjL5H6VdXqJ/o3JMgTzFRfBUfr+I0 m93T6DILHTMsFy+1Hfp1JjSDlSonDhPjJFR

P-ISSN : 2089-3353
E-ISSN : 2808-9162

	BvkWQWmgyobsHqivSpisPMYsQqug/QFx1 yQcpUeP5kDuC4Q9Q2c0QJqYT
DES	elzDOHBFXu0fK+cBrU0FcCI1kx5LassRyjT YE/8oDdXHzMUeycqKlrk8uuFGP+BsOLOH KiU+lbAIQ+hwPJFA2Ngk04PAVzIKMvCcih 2ZQkRsp8efhSH2/8qPLLgz3fqEETQRnD73G AHms2Yo1SQ8BxsEDtWg1gm
Vigenère	€qu|,ƒr…o|zƒ€}d…pw~^zwxg}~v|m^h„xqq€Œ uvhŠxr…{sy€{h„ƒdŠ~swrh‹tr, €|„u‹trŠ ppzxq‹q …q}k†hŠq{ŒytuŠ,w mŒttjƒ|€m}r‰~i‹tr{uƒ€…ud‹xn†€pm…wqz yz}l^xqŽtv{tƒo{to|o {}~w

Data *ciphertext* selanjutnya dianalisis dan melalui proses ekstraksi fitur untuk memperoleh informasi yang bermakna dan representatif, sehingga mampu membedakan karakteristik dari masing-masing algoritma kriptografi. Sebanyak sepuluh fitur diekstraksi dari setiap *ciphertext*, yaitu SDD (*Sequential Digraph Distribution*), DIC (*Digraph Index of Coincidence*), MIC (*Monograph Index of Coincidence*), MKA (*Mean Kappa Analysis*), ROD (*Rate of Digraphs*), LDI (*Letter Distribution Index*), LR (*Letter Repetition*), SHAN (*Shannon Entropy*), IoC (*Index of Coincidence*), dan REP (*Repetition Factor*). Hasil sampel ekstraksi fitur untuk *ciphertext* AES, DES dan Vigenère disajikan pada Tabel 3.

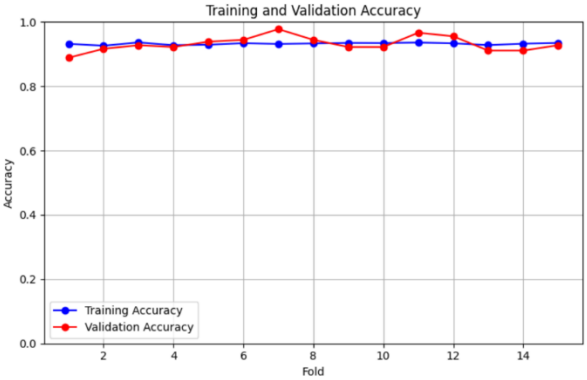
Tabel 3. Sampel Hasil Ekstraksi Fitur dari Ciphertext

Algoritma	Nilai Ekstraksi Fitur		
	SDD	DIC	SHAN
AES	1.135.459.874	272.747.929	5.969.541.471
	1.009.293.238	280.747.929	5.971.058.809
	1.067.725.748	3.305.940.828	5.962.212.086
	1.031.635.562	1.665.940.828	5.989.648.107
	1.125.011.664	2.502.863.905	5.976.338.453
	1.096.954.716	2.333.633.136	5.979.039.104
	1.105.746.133	2.361.325.444	5.977.878.725
DES	1.095.345.094	297.808.284	5.967.362.417
	1.097.705.491	2.248.852.071	5.977.425.843
	1.081.039.559	2.565.775.148	5.972.476.017
	9.935.377.438	2.448.852.071	5.973.226.141
	1.062.861.287	2.821.159.763	5.967.776.641
	1.095.410.288	2.999.621.302	5.964.376.957
	1.080.381.484	2.624.236.686	5.971.455.752
Vigenère	2.548.377.567	1.985.283.951	5.182.552.926
	2.099.569.319	2.424.256.198	5.107.419.456
	2.189.408.768	2.487.950.243	5.070.545.885
	2.250.550.931	2.050.275.825	5.121.382.751
	2.230.624.859	2.647.892.562	5.092.747.847
	2.383.947.735	2.157.081.612	5.151.645.072
	2.317.241.225	2.292.438.017	5.114.652.327

Tahapan berikutnya adalah membagi dataset menjadi data pelatihan (*training set*) dan data pengujian (*testing set*), yang selanjutnya digunakan sebagai masukan bagi model SVM. Data pelatihan dimanfaatkan untuk melatih model dalam mengenali pola-pola yang terdapat pada data, sedangkan data pengujian digunakan untuk mengevaluasi kinerja model dalam melakukan klasifikasi secara akurat.

Gambar 4. Visualisasi *Loss Training* dan *Testing*

Gambar 4 menampilkan grafik visualisasi *loss* selama proses pelatihan dan validasi model. Sumbu X merepresentasikan jumlah *fold* pada proses *cross-validation*, sedangkan sumbu Y menunjukkan nilai *loss* (kesalahan) yang dihasilkan oleh model. Kurva berwarna biru menunjukkan *Training Loss*, yang menggambarkan tingkat kesalahan model terhadap data pelatihan selama proses pembelajaran, sementara kurva berwarna merah menunjukkan *Validation Loss*, yang merefleksikan tingkat kesalahan model pada data validasi selama *cross-validation*. Grafik tersebut membandingkan *Training Loss* dan *Validation Loss* pada 15 *fold*. Nilai *loss* pada kedua kurva berada pada rentang yang relatif rendah dan stabil, yaitu antara 0,02 hingga 0,12. Tidak terdapat perbedaan yang signifikan antara *training loss* dan *validation loss*, yang mengindikasikan bahwa model tidak mengalami *overfitting* dan memiliki kemampuan generalisasi yang baik pada seluruh *fold*.



Gambar 5. Visualisasi Akurasi *Training* dan *Validation*

Gambar 5 menunjukkan bahwa akurasi pelatihan (*training accuracy*) dan akurasi validasi (*validation accuracy*) berada pada kisaran yang tinggi di seluruh 15 *fold*. Penggunaan 15 *cross validation* untuk memastikan bahwa model SVM memiliki Tingkat generalisasi yang tinggi dan meminimalkan resiko overfitting memingat karakteristik ataset yang memiliki variasi fitur yang cukup rapat antar kelas algoritma modern. dipilih Kedua kurva tampak stabil dan saling berdekatan, yang menandakan bahwa model mempertahankan kinerja yang konsisten serta memiliki kemampuan generalisasi yang baik tanpa indikasi *overfitting* maupun *underfitting*. Hasil pelatihan menunjukkan rata-rata akurasi pelatihan sebesar 90,98% dan rata-rata akurasi validasi sebesar 90,00%.

P-ISSN : 2089-3353
E-ISSN : 2808-9162

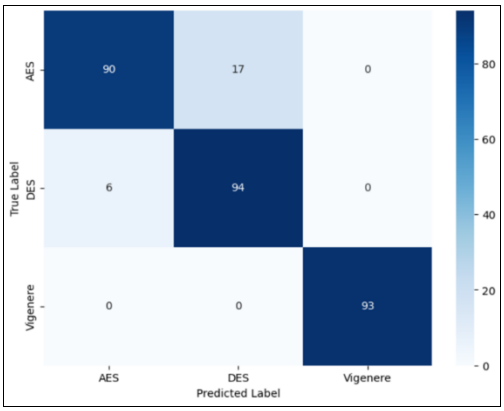
Selain itu, stabilitas yang terlihat pada grafik *Training Loss* dan *Validation Loss* serta grafik *Training Accuracy* dan *Validation Accuracy* mengonfirmasi bahwa model mampu mempelajari pola data secara efektif selama proses pelatihan.

```
Classification Report (Test Data):
Class | Precision | Recall | F1-Score
-------------------------------------
AES    | 0.9375     | 0.8411 | 0.8867
DES    | 0.8468     | 0.9400 | 0.8910
Vigenere | 1.0000     | 1.0000 | 1.0000

Accuracy: 0.9233
```

Gambar 6. Evaluasi Model

Selanjutnya, model dievaluasi dalam kemampuannya untuk mengklasifikasikan *ciphertext* dan menghasilkan akurasi keseluruhan sebesar 92,33%. Matriks kebingungan (*confusion matrix*) menunjukkan bahwa model berhasil mengklasifikasikan sebagian besar data dengan benar, meskipun terdapat beberapa kesalahan klasifikasi antara kelas AES dan DES. Berdasarkan *classification report*, model mencapai nilai presisi dan *recall* yang sangat baik pada kelas Vigenère (presisi = 1,00 dan *recall* = 1,00). Untuk kelas AES, model memperoleh nilai presisi sebesar 0,94 dan *recall* sebesar 0,84, sedangkan untuk kelas DES diperoleh nilai presisi sebesar 0,85 dan *recall* sebesar 0,94. Meskipun nilainya sedikit lebih rendah dibandingkan kelas Vigenère, kinerja pada kelas AES dan DES masih berada dalam kategori memuaskan. Nilai *F1-score* untuk kelas AES dan DES masing-masing sebesar 0,89, yang menunjukkan keseimbangan yang baik antara presisi dan *recall*.



Gambar 7. *Confusion Matrix*

Gambar 7 menampilkan matriks kebingungan untuk dataset pengujian yang terdiri dari 300 sampel. Berdasarkan matriks tersebut, model berhasil mengklasifikasikan sebagian besar data dengan benar, khususnya pada kelas Vigenère yang mencapai akurasi sempurna, yaitu 93 prediksi benar dari 93 sampel. Pada kelas AES dan DES masih terdapat beberapa kesalahan klasifikasi, di mana 17 sampel AES salah diprediksi sebagai DES, dan 6 sampel DES salah diklasifikasikan sebagai AES. Visualisasi matriks kebingungan memberikan gambaran yang lebih jelas mengenai distribusi hasil prediksi model. Meskipun terdapat sejumlah kesalahan klasifikasi antara AES dan DES, tingkat kesalahan tersebut relatif rendah dan tidak berdampak signifikan terhadap kinerja keseluruhan model. Hasil evaluasi ini menegaskan bahwa model SVM memiliki kemampuan yang kuat dalam mengidentifikasi pola kriptografi serta menghasilkan tingkat akurasi yang tinggi dalam klasifikasi jenis *ciphertext*.

4. Kesimpulan

Penelitian ini menunjukkan bahwa klasifikasi *ciphertext* menggunakan algoritma *Support Vector Machine* (SVM) yang dikombinasikan dengan ekstraksi fitur statistik mampu mengidentifikasi jenis algoritma kriptografi secara akurat tanpa memerlukan informasi kunci maupun *plaintext*. Kontribusi utama penelitian ini terletak pada pemanfaatan sepuluh fitur statistik yang efektif dalam merepresentasikan karakteristik khas algoritma AES, DES, dan Vigenère, sehingga mendukung kriptanalisis berbasis *ciphertext* dengan kinerja yang tinggi.

Implikasi dari penelitian ini relevan dalam konteks keamanan siber, khususnya untuk mitigasi serangan *ransomware*, di mana identifikasi algoritma enkripsi secara cepat dapat membantu menentukan strategi respons dan penanganan yang lebih tepat. Untuk pengembangan kedepan, disarankan untuk memperluas cakupan penelitian dengan melibatkan algoritma enkripsi asimetris seperti RSA atau *Elliptic Curve Cryptography* (ECC), serta menguji ketangguhan model terhadap *ciphertext* yang telah melalui proses kompresi atau *encoding* ambahan seperti *base64* yang umum ditemukan dalam serangan siber nyata.

Daftar Rujukan

[1] L. Septiani, “Kronologi Pusat Data Nasional Diretas hingga Pejabat Kominfo Mundur.” [Online]. Available: https://katadata.co.id/digital/teknologi/66862b8b7f375/kronologi-pusat-data-nasional-diretas-hingga-pejabat-kominfo-mundur

[2] G. W. Wahidin, S. Syaifuddin, and Z. Sari, “Analisis Ransomware Wannacry Menggunakan Aplikasi Cuckoo Sandbox,” *J. Repos.*, vol. 4, no. 1, pp. 83–94, 2022, doi: 10.22219/repositor.v4i1.1373.

[3] Y. Zhao and S. Fan, “Analysis of cryptosystem recognition scheme based on Euclidean distance feature extraction in three machine learning classifiers,” *J. Phys. Conf. Ser.*, vol. 1314, no. 1, 2019, doi: 10.1088/1742-6596/1314/1/012184.

[4] W. Stallings, *the William Stallings Books on Computer Data and Computer Communications , Eighth Edition*, 5th ed. New York: Pearson, 2011.

[5] A. Benamira, D. Gerault, T. Peyrin, and Q. Quan Tan, “A Deeper Look at Machine Learning-Based Cryptanalysis,” in *Advances in Cryptology – EUROCRYPT 2021*, Springer, Cham, 2021, pp. 805–835. doi: https://doi.org/10.1007/978-3-030-77870-5_28.

[6] J. So, “Deep Learning-Based Cryptanalysis of Lightweight Block Ciphers,” *Secur. Commun. Networks*, vol. 2020, 2020, doi: 10.1155/2020/3701067.

[7] Y. Fatma, R. Wardoyo, and H. Mukhtar, “An Approach to Cryptography Based on Neural Network,” *AIP Conf. Proc.*, vol. 2601, no. 1, pp. 1–9, 2023, doi: 10.1063/5.0130464.

P-ISSN : 2089-3353
E-ISSN : 2808-9162

[8] R. L. Rivest, "Cryptography and machine learning," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 739 LNCS, pp. 412–426, 1993, doi: 10.1007/3-540-57332-1_36.

[9] S. Baek and K. Kim, "Recent Advances of Neural Attacks against Block Ciphers," in *Symposium on Cryptography and Information Security (SCIS 2020)*, Kochi, Japan: IEICE Technical Committee on Information Security, 2020. [Online]. Available: https://caislab.kaist.ac.kr/publication/paper_files/2020/scis2020_SG.pdf

[10] E. M. Meno, "Neural Cryptanalysis for Cyber-Physical System Ciphers," Virginia Polytechnic Institute and State University, 2021. [Online]. Available: https://vtechworks.lib.vt.edu/handle/10919/103373%0Ahttps://vtechworks.lib.vt.edu/bitstream/handle/10919/103373/Meno_EM_T_2021.pdf?sequence=1

[11] B. Y. Chong and I. Salam, "Investigating deep learning approaches on the security analysis of cryptographic algorithms," *Cryptography*, vol. 5, no. 4, pp. 1–20, 2021, doi: 10.3390/cryptography5040030.

[12] U. J. Ningsih., et al., "Pendekripsian Caesar Chiper Dengan Menggunakan Teknik-Teknik Kriptanalisis," *J. Ilmu Komput. dan Multimed.*, vol. 1, no. 1, pp. 11–15, 2024, doi: 10.46510/ilkomedia.v1i1.10.

[13] M. W. Kurniaga, A. Yulianto, and T. Setya Aji Kumoro, "Kriptanalisis DES menggunakan Jaringan Syaraf Tiruan," *Fidel. J. Tek. Elektro*, vol. 4, no. 2, pp. 40–44, 2022, doi: 10.52005/fidelity.v4i2.89.

[14] N. R. Krishna, "Classifying Classic Ciphers using Machine Learning," 2019.

[15] Ernst Leierzopf, Nils Kopal, Bernhard Esslinger, Harald Lampesberger, and Eckehard Hermann, "A Massive Machine-Learning Approach For Classical Cipher Type Detection Using Feature Engineering," *Proc. 4th Int. Conf. Hist. Cryptol. HistoCrypt 2020*, vol. 183, pp. 111–120, 2021, doi: 10.3384/ecp183164.

[16] K. Begovic, A. Al-Ali, and Q. Malluhi, "Cryptographic ransomware encryption detection: Survey," *Comput. Secur.*, vol. 132, no. February 2022, 2023, doi: 10.1016/j.cose.2023.103349.

[17] R. Nanda, E. Haerani, S. K. Gusti, and S. Ramadhani, "Klasifikasi Berita Menggunakan Metode Support Vector Machine," *J. Nas. Komputasi dan Teknol. Inf.*, vol. 5, no. 2, pp. 269–278, 2022, doi: 10.32672/jnkti.v5i2.4193.

[18] F. Abdusyukur, "Penerapan Algoritma Support Vector Machine (Svm) Untuk Klasifikasi Pencemaran Nama Baik Di Media Sosial Twitter," *Komputa J. Ilm. Komput. dan Inform.*, vol. 12, no. 1, pp. 73–82, 2023, doi: 10.34010/komputa.v12i1.9418.

[19] F. Putrawansyah, "Penerapan Metode Support Vector Machine Terhadap Klasifikasi Jenis Jambu Biji," *JIKO (Jurnal Inform. dan Komputer)*, vol. 8, no. 1, p. 193, 2024, doi: 10.26798/jiko.v8i1.988.