JSI : Jurnal Sistem Informasi *(E-Journal), Vol.15, No.1, 2023*
ISSN Print : 2085-1588 / ISSN Online :2355-4614
http://ejournal.unsri.ac.id/index.php/jsi/index
email: jsifasilkom@unsri.ac.id

Perbandingan Kinerja Kernel RBF dan *Linear* pada Algoritma *Support Vector Machine* (SVM) untuk Prediksi Serangan *Ransomware Locker*

Nurul Afifah[1], Deris Stiawan*[2], Ali Bardadi[3]
[1] Program Studi Teknik Komputer, Fakultas Ilmu Komputer, Universitas Sriwijaya
[2] Program Studi Sistem Komputer, Fakultas Ilmu Komputer, Universitas Sriwijaya
[3] Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Sriwijaya
e-mail: *[1]nurul@unsri.ac.id, [2]deris@unsri.ac.id, [3]alibardadi@unsri.ac.id

Abstrak

Ransomware salah satu jenis Malware yang sangat berbahaya. Cara kerja Ransomware yaitu menyusup lalu mampu menduplikasi file yang dijalankan di sistem operasi windows. Ransomware melakukan enkripsi dan mengunci ke sistem file dengan sangat cepat. Ransomware akan memberikan notice yaitu dengan cara memberitahu cara membuka sistem file dengan melakukan pembayaran melalui Criptocurrency. Ransomware sangat merugikan user. Salah satu jenisnya yaitu Ransomware Locker. Ransomware locker sangat berbahaya, maka dari itu perlu dibuat model prediksi mengenai serangan Ransomware Locker. Beberapa metode Machine Learning mampu menyelesaikan permasalahan sistem prediksi, salah satunya yaitu metode Support Vector Machine (SVM). Untuk mendapatkan model terbaik dalam prediksi, maka perlu melakukan perbandingan antara dua kernel SVM yaitu RBF dan Linear. Hasilnya, metode SVM RBF mampu menghasilkan performa yang sangat baik yaitu 93.27% untuk AUC Train dan 93.41 untuk AUC Test.

Kata Kunci — Malware, Ransomware Locker, Criptocurrency, SVM, RBF, AUC Train, AUC Test

Abstract

Ransomware is one type of Malware that is very dangerous. The way Ransomware works is to infiltrate and then be able to duplicate files that run on the Windows operating system. Ransomware encrypts and locks onto a file system very quickly. Ransomware will give a notice by telling how to open the file system by making a payment via cryptocurrency. Ransomware is very detrimental to users. One type is Ransomware Locker. Ransomware locker is very dangerous, therefore it is necessary to make a prediction model about Ransomware Locker attacks. Several Machine Learning methods are able to solve prediction system problems, one of which is the Support Vector Machine (SVM) method. To get the best model in prediction, it is necessary to do a comparison between two SVM kernels namely RBF and Linear. As a result, the SVM RBF method was able to produce excellent performance, namely 93.27% for AUC Train and 93.41 for AUC Test.

Keywords— *Malware, Ransomware Locker, Criptocurrency, SVM, RBF, AUC Train, AUC Test*

1. PENDAHULUAN

Ransomware adalah jenis malware berbahaya yang diam-diam menginfeksi perangkat korban dengan melakukan enkripsi dan tiba-tiba menuntut tebusan untuk mendekripsi file tersebut. Dalam beberapa pendekatan, banyak penelitian telah dilakukan untuk mendeteksi ransomware pada aplikasi dan mencari kode berbahaya, tetapi masih tetap tidak efisien. Ransomware Locker mengunci seluruh sistem dan menuntut pembayaran untuk membuka kunci sistem dan tidak akan didekripsi kecuali tebusan dibayarkan. Umumnya ransomware terdeteksi menggunakan pendekatan statis dan dinamis [1]. Terkadang, ransomware mengubah perilaku malware untuk membuat proses deteksi lebih sulit atau karena proses filter *private key* yang mengakifkan dekripsi file [2]. Sistem pembayaran tebusan, ransomware menyuruh korban membayar uang tebusan ke rekening bank yang ditunjuk. Metode pembayaran elektronik seperti Bitcoin, MoneyPak, Paysafecard, dan cashU merupakan metode anonim, sehingga sulit untuk melacak asal dan tujuan akhir pembayaran.

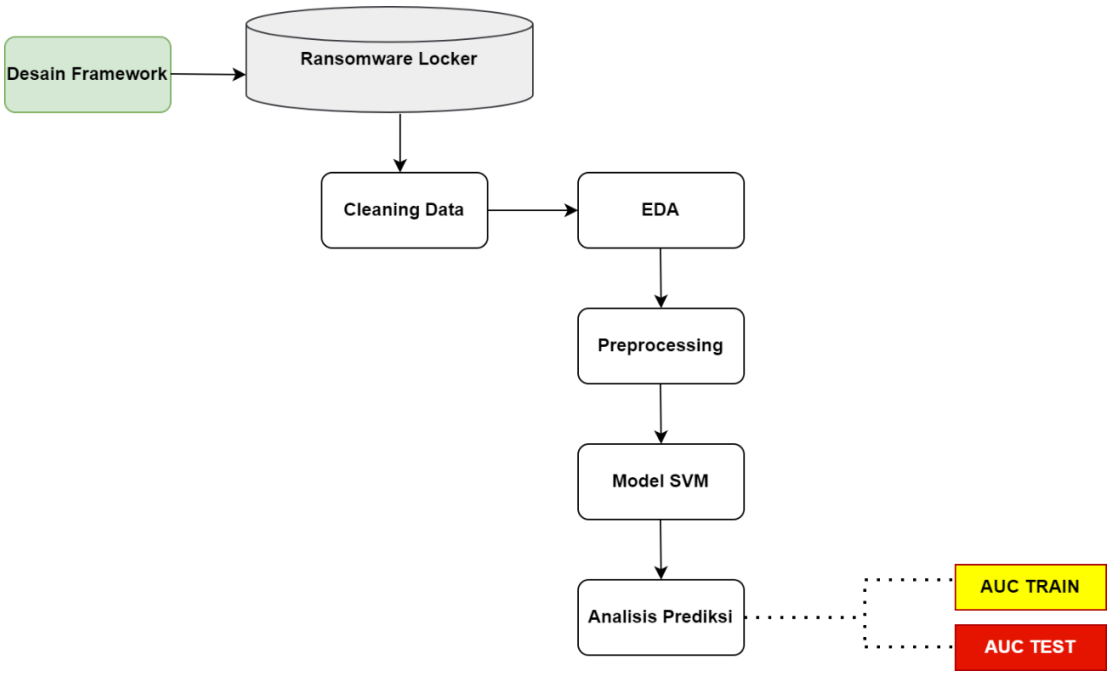
Sejak 2013, Bitcoin telah menjadi salah satu opsi pembayaran alternatif paling populer bagi banyak penyerang ransomware dan cenderung menjadi salah satu faktor kunci dalam kesuksesan ransomware yang

semakin meningkat [3]. Salah satu peneliti [4] melakukan investigasi mengenai teknik pembelajaran mesin untuk klasifikasi malware.

Peneliti [5] melakukan klasifikasi malware android menggunakan algoritma SVM dengan memilih 94 atribut hanya mampu menghasilkan akurasi sebesar 91.25%. Peneliti [6] juga melakukan klasifikasi ransomware menggunakan salah satu algoritma machine learning yaitu SVM. Namun penelitian tersebut belum menerapkan beberapa kernel dari algoritma SVM, sehingga belum ditemukan hasil yang optimal dalam melakukan deteksi ransomware. Mengingat ransomware mengunci seluruh sistem, karena pertumbuhan yang cepat terlihat pada ransomware, ada kebutuhan untuk mengembangkan solusi yang efektif yaitu model yang mampu melakukan prediksi jika ada serangan ransomware dalam suatu sistem baik itu web ataupun mobile. Tetapi, untuk mendapatkan model terbaik dalam suatu algoritma, diperlukan sebuah komparasi sehingga mampu melihat bagaimana perbandingan kinerja dari dua kernel SVM yaitu RBF dan Linear. Kontribusi utama dari paper ini adalah membuat perbandingan kinerja kernel RBF dan Linear menggunakan algoritma SVM dalam melakukan prediksi serangan Ransomware Locker. Paper ini terdiri dari section 1 Pendahuluan, Section 2 Metodologi Penelitian, Section 3 Hasil dan Analisis, dan terakhir Section 4 Kesimpulan.

2. METODOLOGI PENELITIAN

Metodologi yang dilakukan pada penelitian ini terdiri dari beberapa tahapan. Tahapan pertama yaitu melakukan persiapan dataset. Dataset didapat dari VX Heavens Virus Collection database [3]. Tahapan kedua yaitu melakukan proses Exploratory Data Analysis (EDA). Tahapan EDA yaitu melakukan analisis fitur agar mudah dipahami dalam melakukan proses komputasi. Tahapan selanjutnya yaitu melakukan proses training dan testing terhadap metode yang diusulkan yaitu SVM dan menerapkan dua kernel RBF dan Linear. Setelah dilakukan proses validasi lalu hasil dan analisis dan yang terakhir kesimpulan. Pada gambar 1 merupakan proses metodologi penelitian pada paper ini.



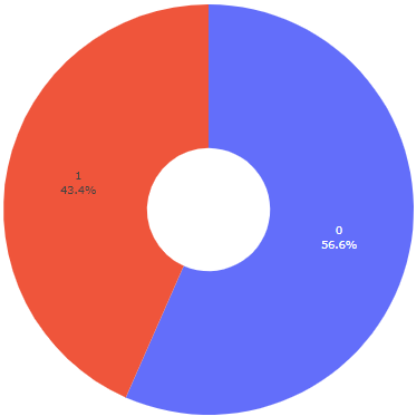
Gambar 1. Metodologi Penelitian

2.1 Persiapan Dataset

Dataset yang digunakan dalam penelitian ini adalah VX Heavens Virus Collection database [3]. Jenis malware yang digunakan dalam penelitan ini adalah Ransomware Locker berjumlah 62.486 sampel.

2.2 EDA

Exploratory Data Analysis (EDA) bertujuan untuk melakukan pendekatan analisis seluruh fitur data guna mendapatkan *summary data* sehingga data dapat dengan mudah untuk dipahami [7]. Pada gambar 2 merupakan hasil EDA distribusi data kelas ransomware (1) dan normal (0). Kelas ransomware berjumlah 43.4% dan kelas normal 56.6%.



Gambar 2. Distribusi Data kelas Ransomware & Normal

2.3 Model SVM

Sistem prediksi ransomware berbasis SVM yang diusulkan menggunakan dua kernel yaitu RBF dan Linear. Algoritma SVM akan berusaha mencari hyperplane terbaik dengan menggunakan kernel dan hyperparameter yang sesuai dengan ketentuan [8]. SVM mampu menghasilkan *decision plane nonlinier* dan memprediksi data yang memiliki distribusi non-reguler. Ini guna menghindari fitur dengan rentang numerik yang lebih besar, mendominasi fitur yang memiliki rentang numerik yang lebih kecil, dan menghindari kesulitan numerik selama perhitungan. SVM bekerja dalam dua fase yaitu *training* dan *testing* [8]. Pemilihan kernel yang tepat, berperan penting dalam menentukan Feature Space untuk menemukan fungsi prediksi. Terdapat beberapa kernel pada penelitian ini yaitu: RBF dan Linear. Penentuan parameter dapat dilakukan dengan beberapa cara, salah satunya ialah cross validation. Nilai C yang besar akan memberikan keluaran yang lebih besar terhadap nilai error prediksi. Nilai dari parameter Gamma berisikan koefesien dan dapat pula berisikan 1/n fitur secara otomatis apabila nilai gamma tidak diberikan [9].

Terdapat beberapa hyperparameter dalam proses validasi proses komputasi model yang tersaji pada tabel 1 berikut.

Tabel 1. Hyperparameter Tunning

Hyperparameter SVM Model
Kernel = RBF & Linear
C=10
Gamma=1
Random state=0
Train Test Split = 60:40, 70:30, 80:20, 90:10

2.3.1 Kernel Linear

Linier kernel SVM merupakan fungsi kernel yang baik digunakan ketika data sudah terpisah secara linier [10].

$$K(x_1, x_2) = x_1^T x_2 \quad \text{... (1)}$$

2.3.1 Kernel *Radial Basis Function* (RBF)

Kernel RBF merupakan fungsi kernel yang digunakan ketika data tidak dapat terpisah secara linier, dimana dalam melakukan analisis dengan RBF akan dilakukan optimasi parameter cost dan gamma [10] [11]. RBF yang dapat menentukan nilai serta lokasi dari center dan nilai pembobot secara otomatis dan mampu memiliki rentang yang tak terhingga. Parameter ini berguna untuk mengontrol trade off yang terjadi antara margin dan error prediksi [10].

$$K(x_1, x_2) = \exp\left(-\frac{\|x_1 x_2\|^2}{2\sigma^2}\right) \quad \text{... (2)}$$

2.4 Confusion Matrix

Pengukuran peforma seberapa efektif model seperti dijelaskan pada peneltiian [12] [13] dapat diukur menggunakan Confusion Matrix. Confusion Matrix merupakan standar ukur performa dalam proses prediksi dalam machine learning [14].

Tabel 2. Confusion Matrix

	Predicted 0	Predicted 1
Actual 0	TP	FP
Actual 1	FN	TN

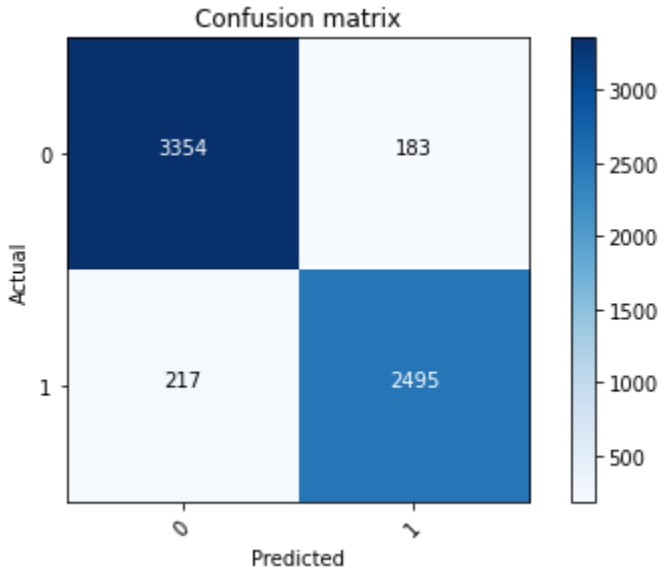
Confusion Matrix merupakan hasil perhitungan dari nilai prediksi dan aktual. TN adalah singkatan dari True Negative yang menunjukkan jumlah sampel negatif yang diprediksi secara akurat [13]. Demikian pula, TP adalah singkatan dari True Positive yang menunjukkan jumlah contoh positif yang diklasifikasikan secara akurat. Istilah "FP" menunjukkan nilai False Positive, yaitu, jumlah contoh negatif aktual yang diklasifikasikan sebagai positif; dan "FN" berarti nilai False Negative yang merupakan jumlah contoh positif aktual yang diklasifikasikan sebagai negative [15].

3. HASIL DAN ANALISIS

Hasil performa algoritma SVM menunjukkan kernel RBF lebih baik dibanding kernel Linear. Kernel RBF mampu menghasilkan performa AUC Training sebesar 93.27% dan AUC Testing 93.41%. Sedangkan kernel Linear hanya mampu menghasilkan performa AUC Training maksimal 84.03% dan AUC Testing 83.84%. Ini menunjukkan bahwa kernel RBF mampu mengontrol trade off yang terjadi antara margin dan error prediksi sehingga mampu menghasilkan performa prediksi terbaik. Pada tabel 3 disajikan hasil performa SVM kernel RBF dan Linear.

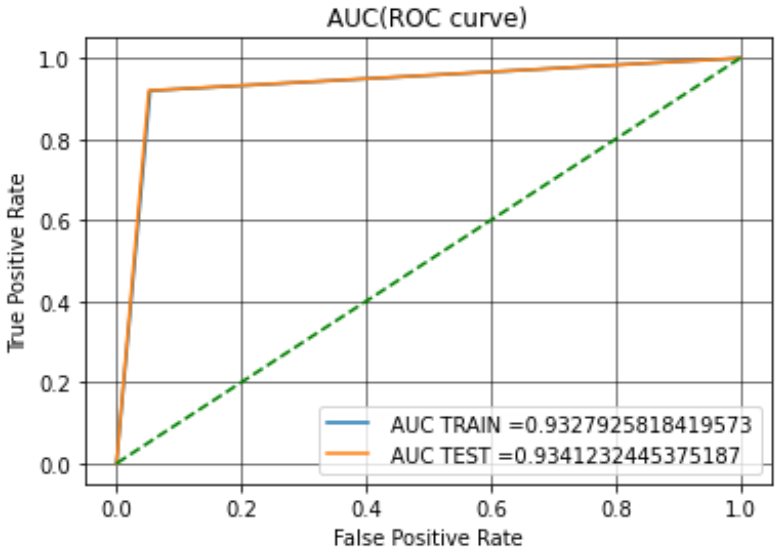
Table 3. *Hasil Performa SVM kernel RBF dan Linear*

Model	***Perbandingan***		***AUC Training***	***AUC Testing***
	Jumlah Data Training	***Jumlah Data Testing***		
SVM kernel RBF	31242	31243	92.57	93.11
	37491	24994	92.76	93.19
	43739	18746	92.84	93.19
	49988	12497	92.97	93.09
	56236	6249	**93.27**	**93.41**
SVM kernel Linear	31242	31243	83.86	84.41
	37491	24994	83.84	84.16
	43739	18746	83.94	84.12
	49988	12497	83.97	84.12
	56236	6249	84.03	83.84



Gambar 3. Hasil Confusion Matrix

Berdasarkan hasil confusion matrix pada gambar 3 menunjukkan kernel RBF mampu menghasilkan performa terbaik, yaitu dengan nilai TP sebesar 3354, TN 2495, FP 183 dan FN 217. Dapat dilihat bahwa dengan FP dan FN yang didapat, performa kernel RBF mampu menghasilkan sistem prediksi yang maksimal yaitu dengan cara meminimalisir serangan ransomware locker yang akan masuk.



Gambar 4. Kurva AUC Kernel RBF

Berdasarkan pengujian yang telah dilakukan diatas, dapat diketahui bahwa kernel Gaussian RBF memiliki rata-rata performansi AUC tertinggi bila dibandingkan jenis kernel linear, yaitu AUC Train 93.27% dan AUC Test 93.41%. Hal ini membuktikan bahwa kernel RBF ternyata mampu melakukan persebaran data yang lebih baik pada saat proses pemetaan datanya

4. KESIMPULAN

Berdasarkan hasil dari keseluruhan penelitian yang dilakukan, dapat disimpulkan bahwa kernel RBF mendapat performa terbaik dalam sistem prediksi yaitu mendapat nilai AUC Training sebesar 93.27% dan AUC Testing sebesar 93.41%. dan konsisten dengan range 90% keatas, berbeda dengan kernel Linear yang hanya mendapat performa AUC maksimal sebesar 84.03%. Kernel RBF mampu membuat sebaran data yang lebih baik dengan melakukan optimasi parameter cost dan gamma sehingga mampu mendapatkan nilai prediksi terbaik.

REFERENCES

[1] U. Desai, “A Survey on Android Ransomware and its Detection Methods,” pp. 3081–3087, 2019.

[2] E. Berrueta, D. Morato, E. Magana, and M. Izal, “Open Repository for the Evaluation of Ransomware Detection Tools,” *IEEE Access*, vol. 8, pp. 65658–65669, 2020, doi: 10.1109/ACCESS.2020.2984187.

[3] J. Chen, C. Wang, Z. Zhao, K. Chen, R. Du, and G. J. Ahn, “Uncovering the Face of Android Ransomware: Characterization and Real-Time Detection,” *IEEE Trans. Inf. Forensics Secur.*, vol. 13, no. 5, pp. 1286–1300, 2018, doi: 10.1109/TIFS.2017.2787905.

[4] S. Sharma, C. Rama Krishna, and S. K. Sahay, “Detection of advanced malware by machine learning techniques,” *Adv. Intell. Syst. Comput.*, vol. 742, pp. 333–342, 2019, doi: 10.1007/978-981-13-0589-4_31.

[5] I. Fitriani, S. Basuki, and A. E. Minarno, “Seleksi Fitur Relieff Pada Klasifikasi Malware Android Menggunakan Support Vector Machine(SVM),” *J. Repos.*, vol. 2, no. 11, p. 1529, 2020, doi: 10.22219/repositor.v2i11.901.

[6] E. S. Lamdompak Sistem Komputer and F. Ilmu Komputer, “Klasifikasi Malware Trojan Ransomware Dengan Algoritma Support Vector Machine (SVM),” vol. 2, no. 1, 2016.

[7] R. W. Emerson, “Exploratory factor analysis,” *J. Vis. Impair. Blind.*, vol. 111, no. 3, pp. 301–302,

2017, doi: 10.1177/0145482x1711100313.
[8] S. Ranveer and S. Hiray, “SVM Based Effective Malware Detection System,” vol. 6, no. 4, pp. 3361–3365, 2015.
[9] J. Mathew, C. K. Pang, M. Luo, and W. H. Leong, “Classification of Imbalanced Data by Oversampling in Kernel Space of Support Vector Machines,” *IEEE Trans. Neural Networks Learn. Syst.*, vol. 29, no. 9, pp. 4065–4076, 2018, doi: 10.1109/TNNLS.2017.2751612.
[10] S. Ali and K. A. Smith, “On learning algorithm selection for classification,” *Appl. Soft Comput. J.*, vol. 6, no. 2, pp. 119–138, 2006, doi: 10.1016/j.asoc.2004.12.002.
[11] K. Thurnhofer-Hemsi, E. López-Rubio, M. A. Molina-Cabello, and K. Najarian, “Radial basis function kernel optimization for Support Vector Machine classifiers,” 2020, [Online]. Available: http://arxiv.org/abs/2007.08233.
[12] A. Luque, A. Carrasco, A. Martín, and A. De Las Heras, “The impact of class imbalance in classification performance metrics based on the binary confusion matrix,” *Pattern Recognit.*, vol. 91, pp. 216–231, 2019, doi: 10.1016/j.patcog.2019.02.023.
[13] Y. Sun, A. K. Bashir, U. Tariq, and F. Xiao, “Effective malware detection scheme based on classified behavior graph in IIoT,” *Ad Hoc Networks*, vol. 120, no. April, p. 102558, 2021, doi: 10.1016/j.adhoc.2021.102558.
[14] A. Charim, S. Basuki, and D. R. Akbi, “Detect Malware in Portable Document Format Files (PDF) Using Support Vector Machine and Random Decision Forest,” *J. Online Inform.*, vol. 3, no. 2, p. 99, 2019, doi: 10.15575/join.v3i2.196.
[15] Y. Ye, T. Li, D. Adjeroh, and S. S. Iyengar, “A Survey on Malware Detection Using Data Mining Techniques,” *ACM Comput. Surv.*, vol. 50, no. 3, pp. 1–40, 2017, doi: 10.1145/3073559.