

KOMBINASI JARINGAN SYARAF TIRUAN DENGAN ALGORITMA HYBRID CAESAR CIPHER DAN VIGENERE CIPHER UNTUK MENGOPTIMALKAN KEAMANAN INFORMASI

Stenly Windharta^{1*}, Lailany Yahya¹⁾ dan Djihad Wungguli¹⁾

¹⁾Program Studi Matematika, Jurusan Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Gorontalo
Gorontalo, 96128, Indonesia
E-mail korespondensi: *s1.math.stenly.w@gmail.com

ABSTRAK

Sejarah Artikel:

Diterima: 25 Maret 2026
Direvisi: 25 Agustus 2026
Diterima untuk
dipublikasikan: 28
September 2026

Kata Kunci:

Kriptografi, Jaringan
Syaraf Tiruan (JST), Tree
Parity Machine (TPM),
Algoritma Hybrid, Caesar
Cipher, Vigenere Cipher

Kriptografi merupakan proses untuk menyembunyikan informasi yang membuat suatu pesan hanya dapat dibaca atau dilihat oleh pengirim dan penerima pesan. Keamanan dalam mengirim pesan sudah menjadi masalah sejak zaman dahulu. Berbagai macam pendekatan telah dilakukan untuk mengembangkan algoritma kriptografi. Dalam penelitian ini, akan digunakan jaringan syaraf tiruan dengan algoritma hybrid. Dua Tree Parity Machine akan disinkronisasikan, dengan menggunakan parameter hidden neurons, input neurons, dan bobot, untuk menghasilkan kunci, yang akan digunakan untuk melakukan proses kriptografi. Proses Kriptografi menggunakan algoritma hybrid, dimana akan dilakukan enkripsi dan dekripsi ganda dengan menggunakan algoritma Caesar cipher dan Vigenere cipher. Penelitian ini memanfaatkan kombinasi jaringan syaraf tiruan dengan algoritma hybrid Caesar cipher dan Vigenere cipher dalam bentuk generator untuk melakukan optimasi keamanan pada pesan. Hasil uji ketahanan algoritma menunjukkan bahwa algoritma yang digunakan stabil, tidak berpengaruh jumlah karakter pada pesan dan nilai parameter. Hasil uji fungsionalitas menghasilkan nilai 100%.



This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

1. PENDAHULUAN

Kriptografi merupakan proses untuk menyembunyikan informasi, yang membuat suatu pesan hanya dapat dibaca atau dilihat oleh pengirim dan penerima pesan [1]. Pada Kriptografi, ada tiga skema kunci pada kriptografi, yaitu kriptografi kunci simetri (kunci rahasia), kunci asimetri (kunci publik), dan fungsi hash. Pada kunci rahasia, digunakan satu kunci untuk melakukan proses enkripsi dan dekripsi pada plainteks [2]. Kunci publik memiliki satu kunci yang dapat disebarluaskan secara tak terbatas sesuai keinginan pemilik kunci, dan fungsi hash atau disebut juga dengan enkripsi satu arah, dapat dikatakan tidak memiliki kunci. Fungsi hash menggunakan nilai hash yang panjangnya telah ditetapkan, yang apabila dikomputasikan dengan plainteks, menyebabkan isi dari plainteks mustahil untuk dipecahkan [3]. Caesar cipher dan Vigenere cipher menggunakan algoritma kunci Simetri. Secara terpisah, kedua algoritma ini memiliki tingkat keamanan yang lemah. Dengan menggabungkan kedua algoritma ini didapatkan algoritma yang memiliki tingkat keamanan yang lebih tinggi [4].

Masalah keamanan menjadi salah satu aspek yang paling penting dalam sebuah sistem informasi. Salah satunya adalah ketika melakukan pertukaran informasi dengan cara mengirimkan pesan [5]. Keamanan dalam mengirim pesan sudah menjadi masalah sejak zaman dahulu. Terbongkarnya pengamanan informasi dapat berdampak besar, bahkan terkadang menyangkut hidup matinya seseorang. Sebagai contohnya, terbongkarnya enkripsi *Enigma* (sebuah mesin yang digunakan pemerintah Jerman selama perang dunia II untuk mengenkripsi/dekripsi pesan-pesan militer) yang dianggap telah memperpendek perang dunia kedua [6].

Berbagai macam pendekatan telah dilakukan untuk mengembangkan algoritma kriptografi. Salah satunya adalah dengan menggunakan jaringan syaraf tiruan. Jaringan Syaraf Tiruan dapat diaplikasikan dengan kriptografi untuk meningkatkan keamanan informasi [1]. Keuntungan menggunakan jaringan syaraf tiruan pada kriptografi adalah tingkat keamanan yang lebih tinggi pada kunci, dikarenakan sistem kunci yang dibentuk oleh jaringan syaraf tiruan terbagi menjadi 2, yaitu beban dan arsitektur jaringan. Sehingga, tidak cukup hanya diketahui salah satu kuncinya saja. Keuntungan lainnya adalah pesan yang dienkripsi tidak dapat diubah sama sekali [7]. Pendekatan ini juga dapat digunakan untuk mengenkripsi gambar seperti format gambar JPEG2000. Dengan menggunakan metode jaringan syaraf tiruan multilayer, dapat dibuat sebuah kunci 128-bit. Setelah dienkripsi pun, enkripsi dari gambar dapat dienkripsi untuk kedua kalinya, meningkatkan keamanan gambar lebih jauh melalui enkripsi ganda. Sehingga, terbentuk sebuah kunci enkripsi dua lapis 256-bit [8]. Jaringan Syaraf Tiruan dapat dikombinasikan dengan cipher-cipher klasik seperti Caesar Cipher dan Vigenere Cipher, untuk meningkatkan daya tahan algoritma dan mengoptimalkan enkripsi dan dekripsi informasi.

Pada kriptografi, algoritma hybrid telah banyak digunakan untuk meningkatkan level keamanan. Algoritma hybrid Caesar cipher dan Vigenere cipher yang dibuat oleh [4] memiliki tingkat keamanan yang lebih tinggi, apabila dibandingkan dengan Caesar cipher, Vigenere cipher dan Hill cipher, berdasarkan parameter seperti frekuensi alfabet dan perilaku grafik. Algoritma hybrid Vigenere cipher dan Polybius cipher yang dibuat oleh [9] menggunakan Vigenere cipher untuk mengenkripsi teks. Selanjutnya, teks cipher akan dienkripsikan lagi dengan menggunakan Polybius cipher. Vigenere cipher yang digunakan memiliki tabel 36x36 dengan menambahkan angka 0 sampai 9, yang umumnya hanya menggunakan alfabet saja dengan tabel 26x26, dan Polybius cipher yang digunakan memiliki tabel 6x6, dimana umumnya hanya menggunakan tabel 5x5 saja. Algoritma hybrid Hill cipher dan Caesar cipher yang dibuat oleh [10] didapatkan bahwa tingkat keamanan dengan menggunakan gabungan kedua algoritma kriptografi lebih tinggi, apabila dibandingkan dengan Caesar cipher atau Hill cipher saja, dengan waktu yang diperlukan untuk melakukan proses enkripsi dan dekripsi untuk kunci matriks 5x5 lebih banyak, daripada menggunakan kunci matriks 3x3.

Jaringan Syaraf Tiruan telah digunakan untuk mengoptimalkan keamanan informasi dengan mengkombinasikan jaringan syaraf tiruan dengan algoritma Vigenere Cipher. Kunci yang dihasilkan melalui *bidirectional learning*, dan dua Tree Parity Machine yang saling tersinkronisasi digunakan untuk melakukan enkripsi dan dekripsi dengan algoritma Vigenere cipher pada teks. Dengan memanfaatkan kombinasi jaringan syaraf tiruan dan Vigenere cipher, dibuatlah sebuah generator untuk mengoptimasi enkripsi dan dekripsi pada pesan [11]. Jaringan Syaraf Tiruan juga digunakan untuk

melakukan enkripsi pada gambar dengan menggunakan algoritma Vigenere Cipher.

Untuk melakukan proteksi pada gambar, diperkenalkan tree parity machine dan Vigenere cipher. Dengan menggunakan pendekatan ini, kunci rahasia tidak akan perlu disebarkan dan tidak diperlukan kemampuan komputasi yang besar untuk melakukan prosen enkripsi dan dekripsi. Untuk mengatasi kelemahan dari Vigenere cipher, dibuat perubahan pada algoritma Vigenere cipher dengan mengalikan nilai kunci dengan axis y dari gambar. Efektivitas diuji dengan korelasi pearson yang mendekati 0 [12].

Berdasarkan uraian di atas, penelitian ini akan menggunakan jaringan syaraf tiruan untuk melakukan optimasi keamanan informasi, yang akan dikombinasikan dengan algoritma hybrid Caesar cipher dan Vigenere cipher.

2. METODE PENELITIAN

2.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan di laboratorium komputer Jurusan Matematika dan Ilmu Pengetahuan Alam, Universitas Negeri Gorontalo. Penelitian dilakukan pada tahun 2025 selama 1 bulan.

2.2 Metode Penelitian

Metode yang akan diterapkan dalam penelitian ini adalah studi literatur.

2.3 Tahapan Penelitian

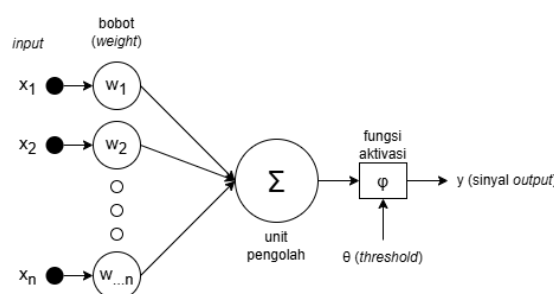
Proses yang dilakukan pada penelitian ini meliputi:

- 1) Studi Pustaka
- 2) Melakukan Analisa Kinerja Sinkronisasi Jaringan Syaraf Tiruan
- 3) Melakukan Analisa Metode Algoritma Hybrid Caesar Cipher dan Vigenere Cipher
- 4) Melakukan Modifikasi Kombinasi Jaringan Syaraf Tiruan dan algoritma hybrid Caesar Cipher dan Vigenere Cipher
- 5) Melakukan Pengujian
- 6) Kesimpulan dan saran

3. KAJIAN PUSTAKA

3.1 Jaringan Syaraf Tiruan (*Artificial Neural Network*)

Jaringan Syaraf Tiruan (*Artificial Neural Network*) adalah metodologi dalam memproses informasi yang dilakukan dengan menggunakan struktur dan aspek fungsional dari jaringan syaraf biologis, memproses informasi seperti otak pada manusia. Jaringan syaraf tiruan adalah salah satu cabang dalam kecerdasan buatan yang menggambarkan sistem pemrosesan informasi yang terdiri atas komponen-komponen yang melakukan proses saling terhubung, yang dikenal sebagai neuron atau *artificial neurons*. Elemen atau bagian dari Jaringan Syaraf Tiruan memiliki 3 komponen utama, yaitu: bobot, *threshold*, dan fungsi aktivasi. Model dasar dari Jaringan Syaraf Tiruan dapat dilihat di Gambar 3.1.



Gambar 3.1. Struktur neuron tunggal jaringan syaraf tiruan

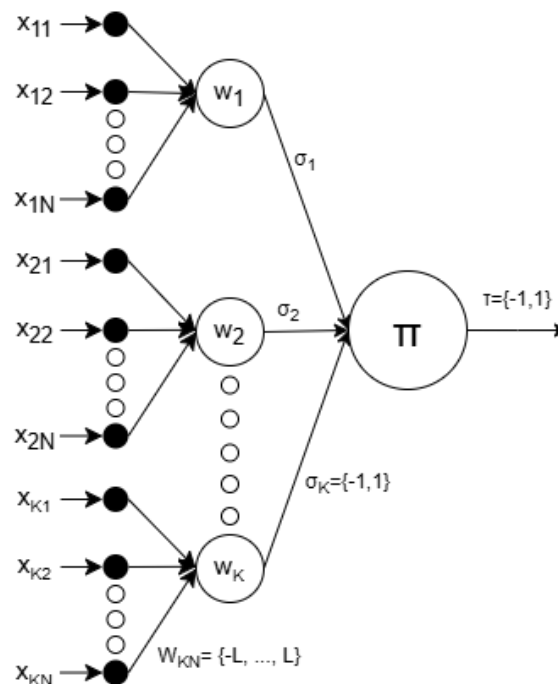
Dari Gambar 3.1, dijelaskan bahwa $input (x_{\{1\}}, x_{\{2\}}, \dots, x_{\{n\}})$ akan berfungsi sebagai sinyal yang diterima oleh sel syaraf. Nilai $w_{\{1\}}, w_{\{2\}}, \dots, w_{\{n\}}$ merupakan faktor dari bobot yang menghubungkan masing-masing node, dimana setiap $input$ akan dikalikan dengan tiap bobot yang terhubung, disesuaikan dengan fungsi aktivasi. Selanjutnya, dihasilkan $threshold$ yang menjadi nilai amban internal, yang mempengaruhi besarnya nilai $offset$ aktivasi node dari $output y$. Dengan persamaan

$$y = \sum_{i=1}^n x_i w_i - \theta$$

Fungsi aktivasi berperan sebagai operasi fungsi matematis yang dimanfaatkan untuk menghasilkan y sebagai nilai $output$ [11].

3.2 Tree Parity Machine

Tree Parity Machine merupakan Jaringan Syaraf Tiruan dengan tipe *multi-layer feed forward* yang terdiri dari 3 bagian: *hidden neuron*, *input neuron*, dan bobot [13].

**Gambar 3.2. Arsitektur tree parity machine**

Pada Gambar 3.2, arsitektur Tree Parity Machine hanya memiliki 1 *output neuron*. *Hidden layer* dari Tree Parity Machine yang terbentuk oleh K *hidden neuron*, dimana setiap *hidden neuron* saling terhubung dengan N *input neuron* sehingga membentuk jaringan yang berukuran $K \times N$. Neuron-neuron dalam Tree Parity Machine dibangun melalui model McCulloch-Pitts dengan representasi data bipolar. Oleh karena itu, nilai dari *input neuron* x_{ij} , *output* dari *hidden neuron* σ_i , dan *output neuron* τ hanya memiliki dua nilai saja, yaitu 1 dan -1 [12].

$$x_{ij} \in \{-1, +1\}, \sigma_i \in \{-1, +1\}, \tau \in \{-1, +1\}$$

Weight merupakan nilai bobot yang saling berhubungan dengan *input neuron* dan *hidden neuron*, dengan angka yang berada dalam *range* -L ke L.

$$w_{ij} \in \{-L, \dots, 0, \dots, +L\}$$

Di sini, i ($1 \leq i \leq K$) menotasikan *hidden neuron* ke- i di dalam *hidden layer* dan j ($1 \leq j \leq N$) menotasikan elemen dari vektor-vektor. Dalam Tree Parity Machine, digunakan fungsi aktivasi yang spesifik, dan yang diketahui sebagai *signum* (sgn) atau langkah fungsi aktivasi. Formula dari fungsi aktivasi adalah sebagai berikut.

$$\text{sgn}(x) = \begin{cases} 1, & \text{jika } x > 0 \\ 0, & \text{jika } x = 0 \\ -1, & \text{jika } x < 0 \end{cases}$$

Jika *output* fungsi di atas adalah 0, nilainya akan dipetakan ke -1 untuk memastikan bahwa *output* yang dikeluarkan hanya berada di nilai 1 atau -1. Secara berturut-turut, *output* dari *hidden neuron* σ_i , dan *output neuron* τ dapat ditemukan dengan menggunakan formula berikut.

$$\sigma_i = \text{sgn} \left(\sum_{j=1}^N w_{ij} x_{ij} \right)$$

$$\tau = \prod_{i=1}^K \tau_i$$

3.3 Caesar Cipher

Caesar Cipher adalah salah satu algoritma yang paling simpel dan paling umum digunakan, yang merupakan kriptografi simetri. Enkripsi pesan dilakukan dengan menggeser posisi yang spesifik dan menggesernya kembali untuk melakukan dekripsi pesan. Pada umumnya, keuntungan dari Caesar cipher adalah kemudahan dalam melakukan aplikasi terhadap algoritma enkripsi lainnya yang lebih kompleks. Kerugian paling utama dari menggunakan Caesar cipher ialah teks cipher sangat mudah didekripsi oleh orang yang membaca teks dikarenakan oleh distribusi alfabet yang berbeda pada teks cipher [4]. Fungsi matematis Caesar cipher ialah sebagai berikut [14].

Untuk enkripsi

$$E_k(x) = (x + k) \bmod 26$$

Dan, untuk dekripsi

$$D_k(y) = (y - k) \bmod 26$$

3.4 Vigenere Cipher

Cara kerja Vigenere cipher melibatkan rangkaian huruf yang disandikan berdasarkan dengan metode Caesar cipher yang bergantung pada karakter yang terdapat di dalam kunci. Huruf-huruf yang digunakan adalah A hingga Z yang dalam prosesnya diubah menjadi angka dari 0 hingga 25. Vigenere cipher adalah salah satu tipe dari sandi substitusi polialfabetik. Apabila dibandingkan dengan Caesar cipher, pendekatan Caesar cipher lebih rentan dibandingkan dengan Vigenere cipher ketika diretas dengan menggunakan analisis frekuensi [11]. Fungsi Matematis Vigenere cipher untuk enkripsi ialah sebagai berikut.

$$C_i = (P_i + K_i) \bmod 26$$

Atau

$$C_i = (P_i + K_i)$$

Apabila jumlah P_i , dan K_i di bawah 26 dan dikurangi 26 jika hasil jumlah di atas 26.

Selanjutnya, fungsi untuk melakukan dekripsi pada Vigenere Cipher

$$C_i = (P_i - K_i) \bmod 26$$

Atau

Dengan $A=0, B=1, \dots, Z=25$

Untuk mencapai sinkronisasi, kesamaan bobot-bobot dari Tree Parity Machine harus selalu dicek dan diperbarui. Bobot-bobot ini akan diperbarui hanya ketika kedua Tree Parity Machine sama. Salah satu dari aturan-aturan pembelajaran yang dapat digunakan untuk sinkronisasi [15]:

- *Hebbian Learning Rule*

$$w_{ij}^+ = g\{w_i + \sigma_i x_i \Theta(\sigma_i \tau) \Theta(\tau^a \tau^b)\}$$

- *Anti-Hebbian learning rule*

$$w_{ij}^+ = g\{w_i - \sigma_i x_i \Theta(\sigma_i \tau) \Theta(\tau^a \tau^b)\}$$

- *Random walk*

$$w_{ij}^+ = g\{w_i + x_i \Theta(\sigma_i \tau) \Theta(\tau^a \tau^b)\}$$

Dimana,

Θ adalah Fungsi Tangga *Heaviside*, jika *input* positif, maka *output* adalah 1, dan jika *input* negatif, maka fungsi membuat nilai menjadi 0,

g menjaga agar nilai bobot tetap berada diantara $\{-L, +L\}$,

x adalah vektor *input*

w adalah bobot

3.7 Pengujian

3.7.1 Uji Ketahanan Algoritma

Pengujian ini dilakukan agar dapat diketahui nilai efektivitas dan daya tampung karakter ketika melakukan proses enkripsi dan dekripsi [11].

3.7.2 Uji Fungsionalitas

Uji fungsionalitas dilakukan dengan menyusun berbagai jenis skenario ketika melakukan enkripsi dan dekripsi, seperti menggunakan kunci yang benar atau salah, untuk mengetahui jika metode kombinasi dapat digunakan[11].

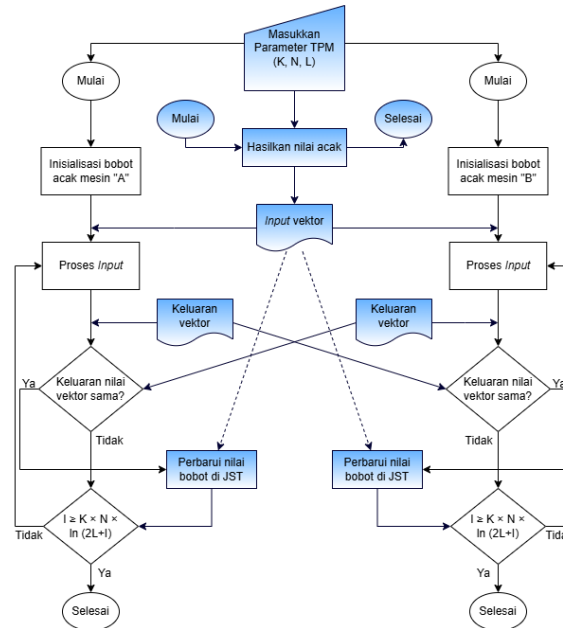
4. HASIL DAN PEMBAHASAN

4.1 Kombinasi Jaringan Syaraf Tiruan dan Kriptografi Menggunakan Tree Parity Machine

Percobaan diawali dengan terminal "A", dan "B", dimana masing-masing terminal menggunakan *tree parity machine* berbeda. Dikombinasikan 2 *tree parity machine* berbeda untuk menghasilkan kunci kriptografi, "A" dan "B" Caesar cipher dan Vigenere cipher. Sinkronisasi tercapai melalui langkah-langkah berikut ini.

- 1) Menginisialisasi nilai-nilai bobot acak
- 2) Melakukan proses-proses berikut sampai sinkronisasi penuh tercapai
 - a) Menghasilkan nilai vektor acak X , melihat apabila keluaran vektor sesuai dengan parameter *input*. Jika belum sesuai, maka nilai bobot akan diperbarui. Jika sudah sesuai, maka akan dilanjutkan dengan proses perhitungan selanjutnya.
 - b) Melakukan komputasi nilai-nilai dari hidden neuron
 - c) Melakukan komputasi nilai-nilai dari output neuron
 - d) Membandingkan nilai-nilai dari kombinasi jaringan *tree parity machine*. Apabila output yang dihasilkan berbeda, maka proses diulang ke a. Apabila output yang dihasilkan sama, maka aplikasikan salah satu jaringan ke bobot

Dengan menggunakan Metode *bidirectional learning*, yang dimana dua tree parity machine tersinkronisasi penuh, bobot-bobot dari "A", dan "B" dapat digunakan sebagai kunci yang dapat digunakan untuk melakukan kriptografi [15]. Sehingga didapatkan diagram alir pada gambar 4.1 berikut.



Gambar 4.1. Diagram alir kriptografi jaringan syaraf tiruan

Keterangan:

- K adalah nilai parameter *hidden neurons*
- N adalah nilai parameter *input neurons*
- L adalah nilai parameter bobot
- A adalah TPM (*tree parity machine*) 1
- B adalah TPM (*tree parity machine*) 2

4.2 Kriptografi dengan Algoritma Hybrid Caesar Cipher dan Vigenere Cipher

Alur Kriptografi dengan menggunakan algoritma hybrid Caesar cipher dan Vigenere cipher dapat dilihat pada Gambar 3.3. Konsep ini didasari oleh algoritma Caesar dan Vigenere yang ditingkatkan performanya dengan menambahkan pergeseran yang berbeda setiap beberapa baris. Simulasi kerja algoritma hybrid Caesar cipher dan Vigenere cipher, misalnya diketahui:

Plainteks(P):

KRIPTOGRAFI ADALAH ILMU SAINS YANG MENGGUNAKAN PERHITUNGAN MATEMATIKA UNTUK MELAKUKAN ENKRIPSI DAN DEKRIPSI DATA KRIPTOGRAFI MEMUDAHKAN KITA DALAM MENYIMPAN INFORMASI SENSITIF ATAU MENGIRIMKANNYA LEWAT JARINGAN-JARINGAN YANG TIDAK AMAN SEPerti INTERNET SEHINGGA INFORMASI TERSEBUT TIDAK DAPAT DIBACA OLEH SIAPAPUN KECUALI OLEH PIHAK YANG DIMAKSUDKAN KRIPTOGRAFI ADALAH STUDI DARI TEKNIK MENGAMANKAN KOMUNIKASI YANG HANYA MEMPERBOLEHKAN PENGIRIM DAN PIHAK YANG DITUJUKAN UNTUK MEMBACA ISI DARI PESAN

Kunci(K): MATEMATIKA

Untuk melakukan enkripsi akan ditentukan terlebih dulu jumlah baris pada plainteks, selanjutnya akan diubah setiap huruf pada plainteks menjadi angka dengan $A=0, B=1, C=2, \dots, Z=25$. Setelah diubah, akan dilakukan pergeseran Caesar pada teks setiap 2 baris, naik dari $n = +1$ sampai $n = +25$, kemudian turun dari $n = +25$ sampai $n = +1$, siklus berulang.

Tabel 4.1. Tabel Proses Enkripsi

Baris	P:	K	R	I	dst...
1	P_i	$10+11=11$	$17+1=18$	$8+1=9$	
Baris	P:	P	E	R	dst...
2	P_i	$15+1=16$	$4+1=5$	$17+1=18$	
Baris	P:	D	A	N	dst...
3	P_i	$3+2=5$	$0+2=2$	$13+2=15$	
Baris	P:	M	E	N	dst...
4	P_i	$12+2=14$	$4+2=6$	$13+2=15$	
Baris	P:	L	E	W	dst...
5	P_i	$11+3=14$	$4+3=7$	$22+3=25$	
Baris	P:	I	N	T	dst...
6	P_i	$8+3=11$	$13+3=16$	$19+3=22$	
Baris	P:	D	I	B	dst...
7	P_i	$3+4=7$	$8+4=12$	$1+4=5$	
Baris	P:	D	I	M	dst...
8	P_i	$3+4=7$	$8+4=12$	$12+4=16$	
Baris	P:	M	E	N	dst...
9	P_i	$12+5=17$	$4+5=9$	$13+5=18$	
Baris	P:	P	E	N	dst...
10	P_i	$15+5=20$	$4+5=9$	$13+5=18$	
Baris	P:	D	A	R	dst...
11	P_i	$3+6=9$	$0+6=6$	$17+6=23$	

Setelah semua baris telah dienkripsi, dilanjutkan dengan melakukan enkripsi Vigenere cipher, dengan menggunakan $C_i = (P_i + K_i) \bmod 26$.

Kunci: M-A-T-E-M-A-T-I-K-A=12-0-19-4-12-0-19-8-10-0

(K): $C_i = (11 + 12) \bmod 26 = 23$ atau X

(R): $C_i = (18 + 0) \bmod 26 = 18$ atau S

(I): $C_i = (9 + 19) \bmod 26 = 2$ atau C, dst...

Sehingga didapatkan cipherteks berikut.

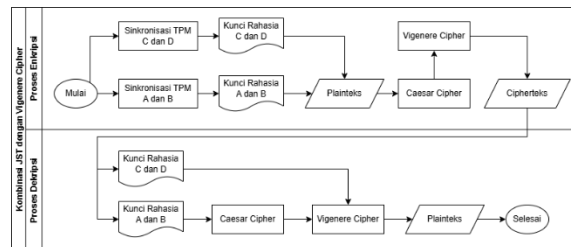
XSCUGPAALGV BXYFBB RWNH TUNAT SJYH ZFHLTVHJVBA
 QYWUJNDYHNO GFGFGJEJXB OSGVE VPMNLOPNO YWVSVQMN
 RCI NQMFKKYW FVDM MFKKZCIMKRK AGHARCCUMP YKOG RCGKY
 OSPTOARVX UPTQMSOUD CQPGKOOT COKG OSPBOFKHUMPBVA
 STZWE WDGLJNPQ FLELCJWU NDJR GLSDG HBDJ DRSTUPP
 XQPPEQ TW OLWLJRTD XQBVGPWDV WTUOLQXP EVGPN ZHEDP
 PWFQGX WBIE EWEFEMCD OBOIEBM LTUL MUVEA CXVW
 HFYOOIYASQR HDWTJSDZQJF MREBEE AJYAU REHM QMARFW
 BJELYVRSINC PFRSWZPYFX DRSE QRSWN BJDUCASTJRWPRS
 NNELGEXR UFL YZMYX NFEL BRKZHHZFE ZLCLP KRBGRHY RJN
 COHO HKRKF

Untuk melakukan proses dekripsi, huruf diubah kembali menjadi angka sesuai dengan $A=0, B=1, C=2, \dots, Z=25$. Setelah diubah akan dilakukan pergeseran Caesar kembali setiap 2 baris, naik dari $n =$

-1 sampai $n = -2$, kemudian turun dari $n = -25$ sampai $n = -1$, siklus berulang, yang dilanjutkan dengan melakukan dekripsi Vigenere cipher, dengan menggunakan $C_i = (P_i - K_i) \bmod 26$.

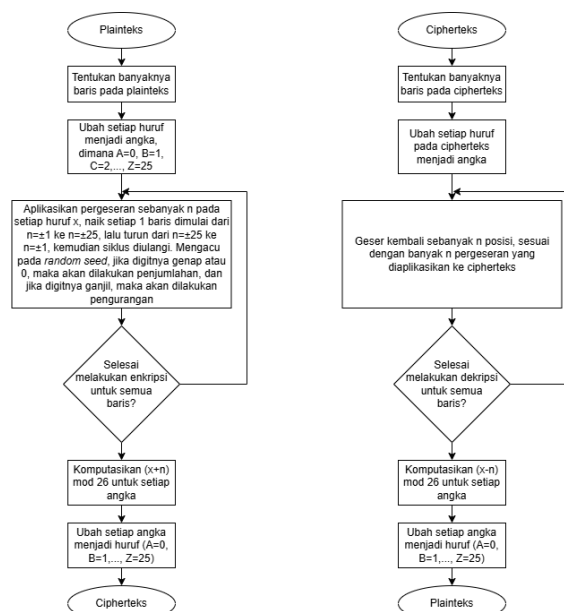
4.3 Modifikasi Jaringan Syaraf Tiruan dengan Algoritma Cipher Hybrid Caesar Cipher dan Vigenere Cipher

Diawali dengan memasukkan nilai K, N, dan L, Sinkronisasi dari TPM A dan B dapat menghasilkan kunci untuk Caesar cipher, dan sinkronisasi dari TPM C dan D dapat menghasilkan kunci untuk Vigenere cipher. Gambar 4.2 menjelaskan alur kerja kombinasi JST dengan algoritma hybrid Caesar cipher dan Vigenere cipher.



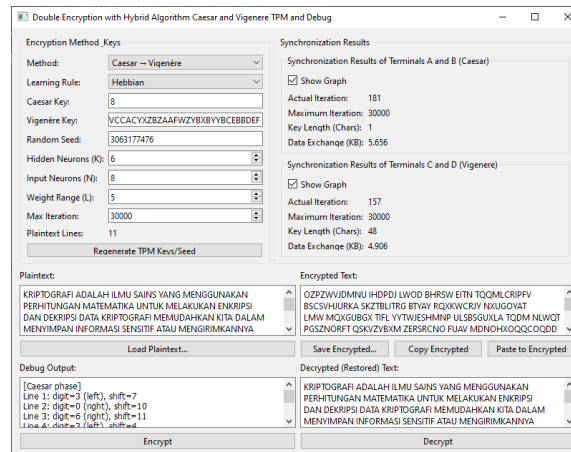
Gambar 4.2. Diagram alir kriptografi kombinasi jaringan syaraf tiruan dengan algoritma hybrid caesar cipher dan vigenere cipher

Adapun dengan menggunakan TPM A dan B, algoritma hybrid ini dapat ditingkatkan lagi. Dengan menghasilkan *random seed*, kunci yang telah dihasilkan untuk melakukan enkripsi dan dekripsi pada Caesar cipher, yang sebelumnya hanya dilakukan pergeseran sebanyak $n = +1$ sampai dengan $n = +25$, kemudian $n = +25$ sampai dengan $n = +1$, dengan siklus berulang untuk setiap 2 baris, dapat dilakukan untuk setiap 1 baris, dengan pergeseran $key \pm 1$ sampai dengan $key \pm 25$, kemudian $key \pm 25$ sampai dengan $key \pm 1$, dengan siklus berulang. Jika digit pada *random seed* adalah genap atau 0, maka akan dilakukan penjumlahan. Sedangkan, jika digit pada *random seed* adalah ganjil, maka akan dilakukan pengurangan. Adapun alur algoritma hybrid yang telah ditingkatkan dapat dilihat pada Gambar 4.3.



Gambar 4.3. Diagram alir algoritma hybrid caesar cipher dan vigenere cipher yang telah ditingkatkan

Implementasi Modifikasi Jaringan Syaraf Tiruan dengan algoritma hybrid Caesar Cipher dan Vigenere Cipher menggunakan generator dengan *user interface* dapat dilihat pada Gambar 4.4.



Gambar 4.4. User interface dari generator jst dengan algoritma hybrid caesar cipher dan vigenere cipher

Dalam generator ini, pengguna (*user*) diharapkan untuk dapat memasukkan nilai parameter *hidden neurons* (K), *input neurons* (N), dan *weight range* atau bobot (L), sehingga dapat dihasilkan kunci untuk melakukan enkripsi dan dekripsi. Setelah kunci dihasilkan, cukup dengan menekan tombol "*Encrypt*" dapat dihasilkan teks cipher, dan dengan menekan tombol "*Decrypt*" teks cipher dapat didekripsi dengan kunci yang sama. Pengguna (*user*) juga dapat memilih metode kriptografi, *learning rules* yang digunakan pada bobot ketika melakukan sinkronisasi TPM, dan iterasi maksimum.

Selain itu, pengguna (*user*) juga dapat mengambil teks dari file .txt, sehingga pengguna dapat melakukan enkripsi pada teks yang diinginkan, dan dapat menyimpan hasil enkripsi dalam bentuk file .txt.

4.4 Pengujian

Akan ada 2 skenario pengujian yang akan dilakukan, yaitu uji ketahanan algoritma dimana akan digunakan nilai-nilai parameter K, N, L yang berbeda-beda, dan uji fungsionalitas.

4.4.1 Uji Ketahanan Algoritma

Pengujian ini akan membandingkan iterasi dan ukuran data yang saling ditukarkan (*data exchange*), berdasarkan nilai parameter K, N, L yang berbeda, dan panjang karakter tertentu. Berikut adalah tabel hasil pengujian ketahanan algoritma hybrid.

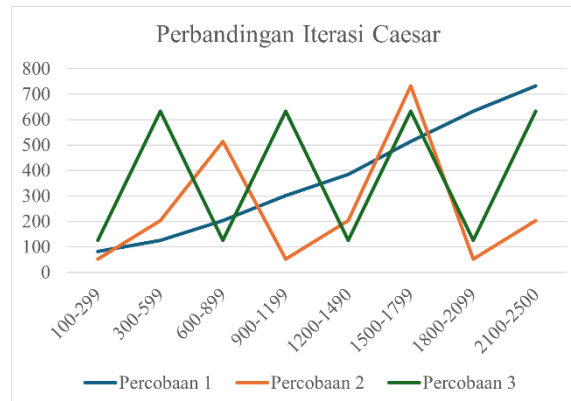
Tabel 4.2. Hasil Pengujian Ketahanan Algoritma Hybrid

Panjang Karakter	Percobaan 1			Panjang Kunci*	Percobaan 2			Panjang Kunci*	Percobaan 3			Panjang Kunci*	Status
	K	N	L		K	N	L		K	N	L		
100-299	4	4	4	16	3	3	3	9	5	5	5	25	Berhasil
300-599	5	5	5	25	6	6	6	36	10	10	10	100	Berhasil
600-899	6	6	6	36	9	9	9	81	5	5	5	25	Berhasil
9000-1199	7	7	7	49	3	3	3	9	10	10	10	100	Berhasil
1200-1499	8	8	8	64	6	6	6	36	5	5	5	25	Berhasil
1500-1799	9	9	9	81	9	9	9	121	10	10	10	100	Berhasil

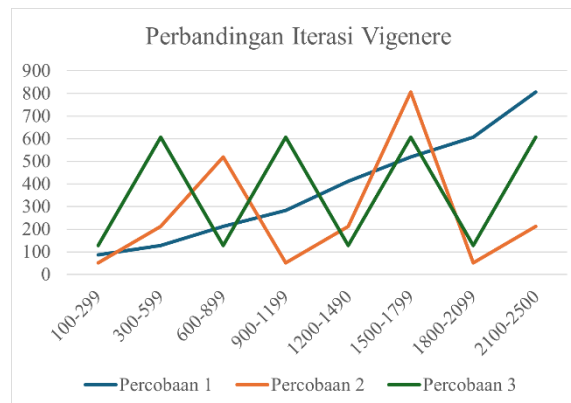
1800-2099	10	10	10	100	3	3	3	9	5	5	5	25	Berhasil
2100-2500	11	11	11	121	6	6	6	36	10	10	10	100	Berhasil

*untuk panjang kunci hanya diperlihatkan panjang kunci algoritma Vigenere cipher saja, dikarenakan panjang kunci Caesar cipher selalu sama, yaitu 1

Selanjutnya dari Tabel 4.2, dibuat grafik perbandingan iterasi dan ukuran data yang saling ditukarkan (*data exchange*) untuk masing-masing algoritma, berdasarkan nilai-nilai parameter pada percobaan 1, 2, dan 3.

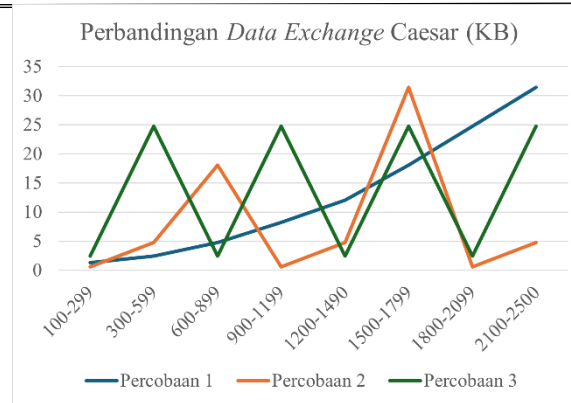


Gambar 4.5. Perbandingan iterasi algoritma caesar

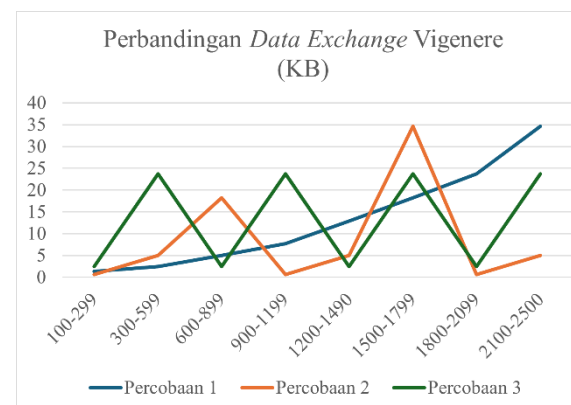


Gambar 4.6. Perbandingan iterasi algoritma vigenere

Dari Gambar 4.5 dan Gambar 4.6 terlihat bahwa iterasi naik secara stabil, seiring dengan naiknya nilai parameter yang digunakan, dan tidak terlihat ada perbedaan yang signifikan pada kedua grafik iterasi.



Gambar 4.7. Perbandingan *data exchange* algoritma caesar



Gambar 4.8. Perbandingan *data exchange* algoritma vigenere

Dari Gambar 4.7 dan Gambar 4.8, sama halnya dengan perbandingan iterasi kedua algoritma pada Gambar 4.5 dan Gambar 4.6, tidak terlihat adanya perbedaan yang signifikan pada kedua grafik. Seiring dengan bertambahnya nilai parameter, bertambah besar pula ukuran *data exchange*. Sehingga, dapat disimpulkan bahwa Sinkronisasi TPM stabil.

Adapun, kunci yang dihasilkan cenderung memiliki panjang yang sama. Hal ini dikarenakan kunci yang digunakan dihitung dengan mengalikan *hidden neurons* K, dan *input neurons* N. Melihat kembali Tabel 4.2, misalnya untuk percobaan 1, nilai K dan N yang digunakan adalah berturut-turut 4 dan 4, sehingga tinggal menghitung $K \times N$ untuk mengetahui bahwa panjang kuncinya adalah 16.

4.4.2 Uji Fungsionalitas

Pada bagian ini, akan diuji fungsionalitas dari algoritma hybrid Caesar cipher dan Vigenere cipher dalam melakukan proses enkripsi dan dekripsi. Tabel di bawah ini menunjukkan beberapa macam skenario dekripsi dengan menggunakan teks cipher dan kunci yang benar ataupun salah.

Tabel 4.3. Hasil pengujian fungsionalitas algoritma

No.	Skenario Input	Output	Hasil
1.	Input teks cipher benar dan kunci benar	Menampilkan hasil dekripsi benar	Valid
2.	Input teks cipher benar dan kunci salah	Menampilkan hasil dekripsi salah	Valid

3.	Input teks cipher salah dan kunci benar	Menampilkan hasil dekripsi salah	Valid
4.	Input teks cipher salah dan kunci salah	Menampilkan hasil dekripsi salah	Valid

Dari Tabel 4.3, dapat disimpulkan bahwa algoritma bekerja dengan benar. Adapun, pengguna (*user*) tidak dapat memasukkan nilai parameter yang bukan angka atau kurang dari 1 di dalam GUI, sehingga mustahil untuk gagal melakukan *generate* kunci.

5. KESIMPULAN DAN SARAN

Kombinasi Jaringan Syaraf Tiruan dengan algoritma hybrid Caesar cipher dan Vigenere cipher menunjukkan mudahnya mengoptimalkan keamanan informasi. Proses enkripsi dan dekripsi hanya memerlukan waktu beberapa detik saja, tidak peduli berapa banyak karakter yang ada pada teks. Adapun, kunci yang dihasilkan bisa bervariasi panjangnya, dan menghasilkan kunci hanya memerlukan waktu yang singkat. Dari pengujian ketahanan algoritma dapat dilihat bahwa sinkronisasi Tree Parity Machine cukup stabil dalam menghasilkan kunci, dan pada uji fungsionalitas, algoritma yang digunakan menghasilkan nilai 100%. Dengan membuat generator juga meningkatkan kemudahan dalam melakukan proses Kriptografi.

Berdasarkan hasil penelitian tentang kombinasi jaringan syaraf tiruan dengan algoritma hybrid Caesar cipher dan Vigenere cipher untuk mengoptimalkan keamanan informasi, disarankan agar penelitian selanjutnya dapat menggunakan algoritma yang berbeda selain Caesar cipher dan Vigenere cipher, dan walaupun bentuk generator sudah lebih mudah digunakan, diharapkan untuk kedepannya pendekatan ini dapat digunakan dalam kehidupan sehari-hari.

DAFTAR PUSTAKA

- [1] S. Pal, B. Datta, and A. Karmakar, "An Artificial Neural Network Technique of Modern Cryptography," *Journal of Scientific Research*, vol. 14, no. 2, pp. 471–481, May 2022, doi: 10.3329/jsr.v14i2.55669.
- [2] N. Parmonangan Simanjuntak, "Penggunaan Artificial Neural Network pada Kriptografi Kunci Simetri," 2012.
- [3] V. Gujral and S. K. Pradhan, "Cryptography using Artificial Neural Networks," 2009.
- [4] C. M. S. Tan, G. P. Arada, A. C. Abad, and E. R. Magsino, "A Hybrid Encryption and Decryption Algorithm using Caesar and Vigenere Cipher," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Aug. 2021. doi: 10.1088/1742-6596/1997/1/012021.
- [5] D. Ariyus and U. Amikom, *Pengantar Ilmu Kriptografi: Teori Analisis & Implementasi*. Penerbit Andi, 2008. [Online]. Available: <https://books.google.co.id/books?id=3SSTJONEmX0C>.
- [6] H. Mukhtar, "Kriptografi untuk Keamanan Data," 2018.
- [7] K. G. . Troitzsch, Michael. Mohring, and Ulf. Lotzmann, *Cryptography Based on Neural Network*. European Council for Modelling and Simulation, 2012.
- [8] Q. A. Memon, "Neural network-based double encryption for JPEG2000 images," *Journal of Information and Communication Technology*, vol. 16, no. 1, pp. 137–155, Jun. 2017, doi: 10.32890/jict2017.16.1.8226.
- [9] M. D. V. Krishna, P. M. Rakesh, C. Harshavardhan, S. K. Gupta, and S. Das, "Improved Cryptography System Using Modified Vigenère and Polybius Cipher," *Transactions of the Indian National Academy of Engineering*, 2025, doi: 10.1007/s41403-025-00535-5.

-
- [10] A. Serdano, M. Zarlis, and E. B. Nababan, "Performance of Combining Hill Cipher Algorithm and Caesar Cipher Algorithm in Text Security," in *2021 International Conference on Artificial Intelligence and Mechatronics Systems (AIMS)*, 2021, pp. 1–5. doi: 10.1109/AIMS52415.2021.9466039.
- [11] A. Fadlil, I. Riadi, A. Nugrahanoro, A. Dahlan, and J. Soepomo, "Kombinasi Sinkronisasi Jaringan Syaraf Tiruan dan Vigenere Cipher untuk Optimasi Keamanan Informasi," 2020, doi: 10.31849/digitalzone.v11i1.3945ICCS.
- [12] M. A. Budiman, Handrizal, and William, "A neural cryptography approach for digital image security using Vigenere cipher and tree parity machine," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jun. 2021. doi: 10.1088/1742-6596/1898/1/012039.
- [13] A. Ruttur, "Neural Synchronization and Cryptography," Nov. 2007, [Online]. Available: <http://arxiv.org/abs/0711.2411>.
- [14] R. Focardi and F. L. Luccio, "Neural Cryptanalysis of Classical Ciphers," 2019.
- [15] S. Man and S. Shrestha, "C++ Implementation of Neural Cryptography for Public Key Exchange and Secure Message Encryption with Rijndael Cipher," 2013.