FACULTAD DE INGENIERÍA ARQUITECTURA Y URBANISMO
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

TESIS

Método De Clasificación De Ataques Ransomware Utilizando Algoritmos A Través De Machine Learning

PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE SISTEMAS

Autor (es)

Bach. Bazan Carhuatanta Angel Junior

ORCID: https://orcid.org/0009-0004-7351-6793

Bach. Perez Arica Robert Frank

ORCID: https://orcid.org/0009-0008-4503-2758

Asesor(a)

Dr. Manuel Guillermo Forero Vargas

ORCID: https://orcid.org/0000-0001-9972-8621

Línea de Investigación

Tecnología e innovación en el desarrollo de la construcción y la industria en un contexto de sostenibilidad

Sublínea de Investigación

Innovación y tecnificación en ciencias de los materiales, diseño e infraestructura

Pimentel – Perú

2024

MÉTODO DE CLASIFICACIÓN DE ATAQUES RANSOMWARE UTILIZANDO ALGORITMOS A TRAVÉS DE MACHINE LEARNING

Aprobación del jurado

Mg. Asenjo Carranza Enrique David

Presidente del Jurado de Tesis

Mg. Guevara Alburqueque Laurita Belen

Secretario del Jurado de Tesis

Mg. Alva Zapata Juliana Del Pilar

Vocal del Jurado de Tesis

Universidad Señor de Sipán

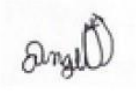
DECLARACIÓN JURADA DE ORIGINALIDAD

Quienes suscribimos la **DECLARACIÓN JURADA**, somos del Programa de Estudios de **Ingeniería de Sistemas** de la Universidad Señor de Sipán, declaramos bajo juramento que somos autores del trabajo titulado:

MÉTODO DE CLASIFICACIÓN DE ATAQUES RANSOMWARE UTILIZANDO ALGORITMOS A TRAVÉS DE MACHINE LEARNING

El texto de mi trabajo de investigación responde y respeta lo indicado en el Código de Ética de la Universidad Señor de Sipán, conforme a los principios y lineamientos detallados en dicho documento, en relación con las citas y referencias bibliográficas, respetando el derecho de propiedad intelectual, por lo cual informo que la investigación cumple con ser inédito, original y autentico.

En virtud de lo antes mencionado, firman:

Bazán Carhuatanta Ángel Junior	DNI: 74747774	
Pérez Arica Robert Frank	DNI: 72496295	

Pimentel, 20 de diciembre de 2023.

Dedicatoria

En primer lugar, deseo dedicar esta tesis a Dios, cuya guía y fortaleza han sido fundamentales en mi camino hacia este importante logro en mi vida profesional. Su divina orientación ha sido la luz que me ha guiado en cada paso de este proceso académico, siendo esta meta una de las más anheladas y significativas para mí. Expreso mi más profundo agradecimiento a mi madre, Lily Marilú Carhuatanta Leyva, y a mi tía, Flor Esperanza Leiva Serrano, cuyo amor incondicional y dedicación han sido pilares esenciales a lo largo de estos años. Su constante sacrificio y apoyo moral han sido un sostén invaluable en mi trayectoria académica, impulsándome a perseverar en busca de mis metas. Asimismo, no puedo pasar por alto el incondicional apoyo de mi pareja Yeraline Chavez, quien ha sido mi compañera constante durante este arduo camino. Su apoyo inquebrantable y aliento en los momentos más difíciles han sido un pilar fundamental en mi éxito académico. A todas las personas que han sido y son significativas en mi vida, les expreso mi más sincero agradecimiento por su inquebrantable apoyo a lo largo de estos cinco años de carrera. Cada palabra de aliento, gesto de ánimo y muestra de confianza ha sido un motor que me ha impulsado hacia adelante, permitiéndome alcanzar esta meta tan trascendental en mi vida. Su presencia y respaldo han sido un regalo invaluable que atesoro con gratitud, y sin su contribución, este logro no habría sido posible. A todos ellos, mi más sincero reconocimiento y agradecimiento por ser parte integral de mi camino hacia el éxito académico y profesional.

Angel Bazan

Dedico con todo mi corazón mi tesis a mis padres, quienes desde el primer día de esta travesía académica me mostraron el valor del esfuerzo y la perseverancia, su apoyo incondicional fue mi faro en los momentos de duda y desafío. A mi amada esposa, compañera incansable en este viaje, quien compartió mis alegrías y mis penas, siempre alentándome a dar lo mejor de mí. Y a mi hijo, cuya sonrisa radiante fue mi combustible en los días más agotadores. Juntos hemos caminado esta senda, enfrentando desafíos y celebrando triunfos. Este logro lleva impreso el amor y la dedicación de cada uno de ustedes.

Robert Perez

Agradecimientos

Expresamos nuestro sincero agradecimiento a los distinguidos docentes de la Universidad Señor de Sipán, cuya guía experta y dedicación han sido fundamentales en nuestra formación académica y en el desarrollo de esta tesis. Reconocemos el apoyo incondicional de nuestros padres, cuyo amor, comprensión y sacrificio han sido la fuerza motriz que nos ha impulsado a superar obstáculos y alcanzar nuestros sueños. También agradecemos profundamente a nuestro respetado asesor de tesis, Manuel Guillermo Forero Vargas, cuya orientación experta y compromiso con nuestro crecimiento académico y profesional han sido una inspiración constante. Este logro es resultado del esfuerzo conjunto de todos los que nos brindaron su apoyo y aliento a lo largo de esta travesía académica. Sin su ayuda, este proyecto no habría alcanzado la excelencia que hoy celebramos.

Los Autores

ÍNDICE

Índice de tablas. Figuras y fórmulas

Índice de Figuras

Índice de Tablas

Resumen

El ransomware es una seria amenaza para la seguridad cibernética, siendo conocido por su capacidad destructiva al cifrar datos de organizaciones y exigir rescates para su liberación. Esta tesis aborda la creciente sofisticación de estos ataques y propone un método de clasificación mediante algoritmos de Machine Learning con el objetivo de comprender mejor los archivos ransomware y mejorar la seguridad cibernética. Se seleccionó un dataset que incluye tanto muestras benignas como malignas de ransomware. Estos datos fueron sometidos a un análisis detallado para extraer características relevantes, como Machine, DebugSize, DebugRVA, MajorImageVersion, MajorOSVersion, ExportRVA, ExportSize, IatVRA, MajorLinkerVersion, MinorLinkerVersion, NumberOfSections, SizeOfStackReserve, DllCharacteristics, ResourceSize, con el objetivo de clasificarlos como 'Benign'. Posteriormente, el dataset se dividió en un 80% para entrenamiento y un 20% para pruebas. Se procedió al entrenamiento del modelo, ajustando los hiperparámetros y calculando métricas como accuracy, precisión también recall y por último F1-Score. Se implementó el método de Voting para la clasificación por votos y, finalmente, el modelo entrenado se guardó en un archivo joblib para su posterior uso en la clasificación. Los resultados mostraron un rendimiento excepcional para los algoritmos Decision Tree y Random Forest, alcanzando un 99.4% y 99.6% en cada una de las métricas evaluadas, respectivamente. En cuanto al algoritmo SVM, se observaron resultados variables con un 87.50% en accuracy y F1-Score, un 87.40% en recall y un sorprendente 99.96% en precisión.

Palabras Clave:

Ransomware, Ciberseguridad, Machine Learning, Amenaza, Clasificación

Abstract

Ransomware is a serious threat to cyber security, being known for its destructive ability to encrypt organizational data and demand ransoms for its release. This thesis addresses the increasing sophistication of these attacks and proposes a classification method using Machine Learning algorithms with the goal of better understanding ransomware files and improving cyber security. A dataset including both benign and malicious ransomware samples was selected. These data were subjected to detailed analysis to extract relevant features, such as Machine, DebugSize, DebugRVA, MajorImageVersion, MajorOSVersion, MajorOSVersion, ExportRVA, ExportSize, IatVRA, MajorLinkerVersion, MinorLinkerVersion, NumberOfSections, SizeOfStackReserve, DllCharacteristics, ResourceSize, with the aim of classifying them as 'Benign'. Subsequently, the dataset was divided into 80% for training and 20% for testing. We proceeded to train the model, adjusting the hyperparameters and calculating metrics such as accuracy, precision, recall and finally F1-Score. The Voting method was implemented for classification by votes and finally the trained model was saved in a joblib file for later use in classification. The results showed exceptional performance for the Decision Tree and Random Forest algorithms, reaching 99.4% and 99.6% in each of the evaluated metrics, respectively. As for the SVM algorithm, variable results were observed with 87.50% in accuracy and F1-Score, 87.40% in recall and an astonishing 99.96% in precision.

Keywords:

Cybersecurity, Machine Learning, Threat, Classification

I. INTRODUCCIÓN

En la era digital actual, la amenaza del ransomware se ha convertido en un desafío crítico para la seguridad cibernética, actualmente encontrándose entre los ciberataques más ofensivos y dañinos que una organización puede experimentar, este tipo de ataques ha evolucionado de manera significativa en términos de complejidad y sofisticación, mediante estos ataques hacen uso de malware para poder infectar y poder acceder a los sistemas encriptando los datos que encuentran paralizando las organizaciones y poniéndose en peligro la identidad de las personas; y para colmo de males, los atacantes piden un pago a cambio de devolver los archivos a las manos de la organización [1]. En el año 2020, se presentaron algo más del 51% de ciberataques para los Sistemas de Información y en algunos de los servidores la cual contienen información sensible como nos menciona [2] También en un informe de Trend Micro, el 84 % de las organizaciones estadounidenses experimentaron ataques de phishing o ransomware en el último año equivaliendo a unos 65000 ataques solo en el mismo año [2]. Deloitte nos afirma que sólo el 10% de las empresas peruanas cuenta con indicadores de gestión de riesgos, y que el 51% de las empresas han sido afectadas por ataques como el phishing y malware [3]. El ransomware es el actual problema de las organizaciones a nivel mundial presentando una serie de dificultades técnicas. El autor [4] experto en Machine Learning y está trabajando en métodos avanzados para poder extraer características relevantes de cada archivo de ransomware por lo cual se podría mejorar la precisión de diferentes algoritmos, sin embargo, de acuerdo con [5] nos dice que los algoritmos de ML deben ser entrenados para determinar cuáles son los más efectivos a la hora de clasificar, también [6] utiliza el procesamiento de una cantidad de datos masivos relacionados al ransomware para que permitan mejores trabajos de mejores cantidades de datos, como también nos dice [7] utiliza interfaces intuitivas para que los usuarios finales puedan interactuar con las herramientas de clasificación para el ransomware. Este tipo de ataque representa una amenaza grave

para individuos, empresas e incluso gobiernos, porque presenta un aumento significativo en la frecuencia y sofisticación en estos últimos años. Nuestro objetivo es poder contribuir con la ciberseguridad, brindando un método nuevo que nos ayudara a comprender las características de los archivos que puedan contener ransomware, sus características generales, para mejorar los sistemas de la seguridad. Nos planteamos la siguiente pregunta de esta investigación ¿De qué manera se podrá clasificar ataques de ransomware?" Para responder a esta pregunta, Según nuestra hipótesis nos quiere decir que al implementar un método utilizando algoritmos de machine learning, entonces se podrá clasificar los ataques de ransomware de manera efectiva. Haciendo el uso adecuado de las técnicas de ML y análisis de características relevantes de los archivos ransomware, se busca desarrollar un método que permita identificar y clasificar estos ataques de forma precisa y eficiente. La implementación de algoritmos de machine learning ofrece una potente herramienta para la detección y clasificación de ransomware, brindando una mayor comprensión de las y características de estos ataques, lo cual contribuirá significativamente a fortalecer la seguridad cibernética y proteger a las organizaciones contra esta amenaza cada vez más sofisticada. Se ha seleccionado un dataset con diferentes características de archivos tanto benignos como malignos. Estos datos se someten a un riguroso análisis para extraer las diferentes características relevantes que nos servirán como base para la clasificación del ransomware. Las herramientas tecnológicas basadas en ML han tenido un gran impacto en la ciberseguridad como por ejemplo la clasificación del ransomware, para medir el rendimiento, la precisión entre otras métricas importantes. Diversos autores han realizado investigaciones sobre la clasificación de ransomware. En la investigación de [8], se abordó la clasificación de transacciones de ransomware en Bitcoin utilizando machine learning. Se equilibraron los datos mediante submuestreo y sobremuestreo, evaluando tres algoritmos de clasificación (LR, RF y XGBoost) en el dataset equilibrado. Los resultados mostraron que XGBoost tuvo el mejor rendimiento con un 99.9% de precisión, 99.8% de recall y 99.9% de F1-score.

RF obtuvo resultados ligeramente inferiores, y LR tuvo el peor rendimiento. Mientras tanto también [9] propuso un método novedoso de clasificación de ransomware utilizando una Red Neuronal Siamesa basada en aprendizaje de pocos ejemplos y características de entropía. Este método superó a técnicas previas como DNN, RNN y VGG, con una mejora de aproximadamente el 15%. Se obtuvo un puntaje promedio de F1 del 88%, demostrando una efectividad del 94% en la clasificación de ransomware. En conclusión, el enfoque de meta-aprendizaje y la incorporación de características basadas en entropía resultaron en una clasificación más efectiva de ransomware. Mientras tanto [10] realizó una investigación que se centró en desarrollar un método mejorado para clasificar y agrupar ransomware. Utilizaron características estáticas como peso de archivos, cadenas de caracteres y funciones API, combinadas con algoritmos de machine learning. Obtuvieron una precisión del 99.72% teniendo en cuenta que su tasa de FP es de 0.003% en un conjunto de datos de dos clases. Concluyeron que este método es prometedor en la lucha contra el ransomware en constante evolución, también [11] realizó una investigación sobre un nuevo enfoque llamado SIMPLE para la clasificación de malware en familias desconocidas. Utilizó técnicas de aprendizaje basado en instancias y agrupamiento supervisado para generar prototipos representativos. Los resultados mostraron precisiones de clasificación de hasta el 94,15% en ciertas tareas en el conjunto de datos VirusShare_00177 y hasta el 89,22% en el Dataset de APIMDS. En conclusión, el enfoque SIMPLE es innovador y efectivo en la clasificación de malware con pocos ejemplos de entrenamiento. También [12] realizó una investigación sobre la clasificación de ransomware mediante el uso de análisis de comportamiento. Utilizó un algoritmo de clasificación envolvente basado en PSO para seleccionar características óptimas, logrando una precisión del 98.03% en la clasificación binaria y 54.84% en la clasificación multiclase. En conclusión, el estudio propone un enfoque innovador para detectar ransomware mediante la selección de características relevantes utilizando PSO. También [13] investigó y desarrolló un método de detección de malware llamado

BHMDC, basado en n-gramas de bytes y hexadecimales, que combina extracción de características y prototipos para su clasificación a la hora de poder mejorar su precisión como también eficiencia. Se obtuvieron resultados de alta precisión, como 99.264%, 97.364%, 99.12%, 99.18%, 98.58%, 98.04%, 98.06%, 99.50%, 98.5%, 96.9%, y 96.6% en diversos conjuntos de datos, demostrando su efectividad en la clasificación de malware. En conclusión, este enfoque supera a otros métodos existentes en términos de precisión y es altamente efectivo en la detección y clasificación de malware. Mientras tanto el estudio de [14] se enfocó en la detección y clasificación de malware, particularmente ransomware, empleando diversas redes neuronales. Los resultados resaltaron una precisión del 100% en la detección binaria con el perceptrón multicapa, mientras que la red neuronal convolucional alcanzó un 94% de precisión en la clasificación de las nueve familias de ransomware. Estos hallazgos subrayan la relevancia de elegir cuidadosamente el tipo de red neuronal para futuras tareas de detección y clasificación de malware. También [15] nos menciona que su investigación se enfocó en clasificar familias de ransomware mediante el uso de N-gramas de opcodes y TF-IDF para la extracción de características. Se empleó un modelo MLP para la clasificación, obteniendo una precisión del 91,43% y una medida F1 del 99% para WannaCry, así como una precisión del 99,3% en la detección binaria. Los clasificadores basados en Random Forest lograron una precisión entre el 85,15% y el 91,43% utilizando 2-gramos, 3-gramos y 4-gramos, mientras que los basados en Naive Bayes oscilaron entre el 45,89% y el 70,34% con los mismos gramajes. Estos resultados respaldan la efectividad del enfoque propuesto para mejorar la detección y defensa contra ransomware. A través de la investigación de [16] se desarrolló un enfoque innovador para identificar pagos de ransomware en redes de Bitcoin heterogéneas utilizando técnicas de análisis de transacciones de Bitcoin y algoritmos de ML. Los resultados mostraron un error del 0.10% en la clasificación binaria y del 0.60% en la clasificación multiclase, con una precisión del 93.91% en la clasificación binaria y del 99.40% en la

clasificación multiclase. Además, el modelo logró un recall del 99.90% en la clasificación binaria y del 99.30% en la clasificación multiclase, con un F1-Score del 96.82% en la clasificación binaria y del 99.35% en la clasificación multiclase. Estos resultados respaldan la eficacia del modelo propuesto para la detección de pagos de ransomware en redes de Bitcoin heterogéneas. Mientras tanto [17] realizó una investigación sobre la clasificación de familias de malware empaquetado utilizando técnicas de ML y análisis dinámico. Extrajo características de un amplio dataset, utilizó análisis dinámico para comprender el comportamiento del malware, empleó redes neuronales profundas (DNN) para la clasificación y una red generativa adversarial (GAN) para generar muestras de malware empaquetado. Los resultados incluyeron una precisión del 94,1% en el conjunto de datos original, 66,60% sin GAN y 71,25% con GAN, demostrando una alta capacidad de detección y clasificación de malware empaquetado.

II. MATERIALES Y MÉTODO

2.1 MATERIALES

En esta investigación se requiere una infraestructura tecnológica, y está compuesta por los siguientes materiales y componentes como se muestra en la siguiente tabla:

Material	Descripción
Computador 1	El procesador que se utilizó es Intel(R) Core(TM) de i7-7500U con un CPU de 2.70GHz, 24GB RAM y también un SSD con 1TB de espacio, con una tarjeta de video AMD Radeon de 4 GB, SSD de 1TB.
Computadora 2	El procesador utilizado es el Intel(R) Core(TM) de i3-10100 de CPU 3.60 GHz, 8 GB de RAM y un SSD de 1TB, Tarjeta de Vídeo Nvidia 1050 ti.
Sistema Operativo	Windows 10 y 11.
Softwares	Visual Studio Code, Google Colab, Balsamiq.

2.2. MÉTODO

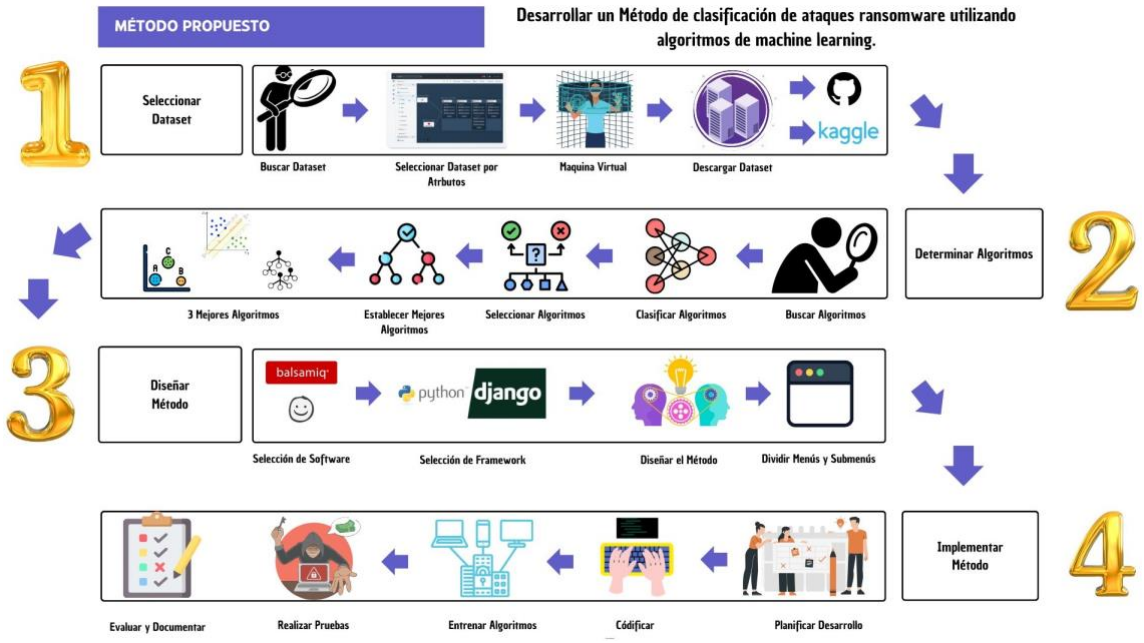


Figura 1: - Método principal.

Para realizar la clasificación de ransomware, mediante algoritmos de machine learning, fue necesario identificar los dataset disponibles en el ámbito del ransomware, la necesidad de contar con un dataset principalmente es por la cantidad de datos que se ofrecen a futuro a diversos algoritmos para posteriormente ser entrenados y tener una predicción exacta para tener los mejores resultados a comparación de otros proyectos de investigación. Sin embargo se sabe que existe una cantidad ilimitada de dataset de distintas temáticas. Para ello fue necesario realizar una búsqueda exhaustiva entre diferentes artículos científicos descargados de base de datos confiables entre ellos scopus, IEEExplore, Science Direct y el repositorio seleccionado Kaggle siendo el más usado por la comunidad científica. con la finalidad de identificar los dataset existentes sobre la temática de la clasificación de ransomware por lo cual se logró identificar 6 dataset los cuales se muestran en la **Tabla 1**, de estos dataset identificados se seleccionó un dataset "RandomFore_ransomware_detection_and_classification", esto siguiendo los criterios de calidad de los datos siendo un criterio de vital importancia al seleccionar un dataset para la clasificación de ransomware. Nos basamos en dos pilares fundamentales como

es la precisión y confiabilidad de los datos que nos garantizan que los modelos de clasificación sean efectivos en la toma de decisiones. Como segundo criterio consideramos la disponibilidad del dataset siendo de crucial importancia para nuestra investigación y el desarrollo del entrenamiento. Acceder a un conjunto de datos ampliamente disponible facilita la replicación de estudios, la colaboración entre investigadores y las novedades en diferentes campos. Mientras en el tercer criterio consideramos el volumen de datos medio, por la actual capacidad computacional que se tiene acceso. Esto implica determinar un equilibrio que permita el manejo eficiente de datos sin sobrecargar los recursos computacionales disponibles, asegurando así un rendimiento óptimo en la clasificación. Por último consideramos el criterio de etiquetado preciso, lo cual nos da mayor seguridad que las muestras estén etiquetadas correctamente para el posterior entrenamiento ver **Tabla 2**. El dataset seleccionado fue sometido a un proceso de limpieza con la finalidad de estandarizar el dataset.

N°	*Nombre del Dataset*	*Referencia*
1	RandomFore_ransomware_detection_and_classification	[18], [19]
2	Virus Total	[20]
3	Virus Share	[21]
4	Ransap	[22]
5	TheZoo	[23]
6	Cukoo Sandbox	[24]

Tabla 1: Listado de los Datasets encontrados.

N°	*Dataset*	*Calidad de datos*	*Disponibilidad*	*Volumen de datos*	*Etiquetado*
1	RandomFore ransomware detection and classification	x	x	x	x
2	Virus Total	x		x	x
3	Virus Share	x		x	
4	Ransap	x		x	
5	TheZoo	x			x
6	Cukoo Sandbox	x	x	x	

Tabla 3: Selección de bajo criterios para el Dataset.

Es necesario realizar la selección de algoritmos de ml que presenten mejores resultados de rendimiento para ser considerados en este trabajo de investigación, para lograrlo se llevó a cabo una exhaustiva revisión sistemática de la literatura teniendo como fuente las bases de datos Scopus, IeeeXplore, ScienceDirect de los cuales utilizando estrategias de búsqueda se logró identificar 182 artículos científicos relacionados al tema, sin embargo luego de un proceso de selección utilizando criterios de inclusión y exclusión se logró identificar 74 artículos científicos relevantes de los cuales después de una lectura minuciosa se identificó los algoritmos de ml que había empleado, lo cual permitió elaborar un listado preliminar de estos algoritmos ver **Anexo 6** – listado de algoritmos a fin de ser seleccionados utilizando criterios como los de Precisión, Accuracy, Recall, F1-Score ver **Tabla 3**. esto con la finalidad de contar con los mejores algoritmos para ser implementados en este trabajo de investigación. Como resultado de este procedimiento se logró seleccionar a los algoritmos (svm, Decisión Tree, Random Forest) los cuales presentaron los mejores resultados en la **Tabla 4**.

Arquitectura Decision Tree

La arquitectura del árbol de decisión se representa visualmente como un gráfico dirigido (digraph). Cada nodo interno del árbol es representado por cuestiones de las características del algoritmo, Como también las hojas del árbol representa la predicción de la clase. Cada nodo tiene una etiqueta que describe la condición de división del nodo y otros detalles relevantes, como el coeficiente de impureza Gini, el número de muestras y el recuento de clases. Los nodos se conectan mediante aristas que representan las ramas del árbol y las condiciones de división, A continuación se muestra la arquitectura en la Figura 2. La fórmula de Decision Tree es:

$$IG(T, a) = H(T) - \sum_{v \in valores(a)} \frac{T_v}{T} H(T)_v$$

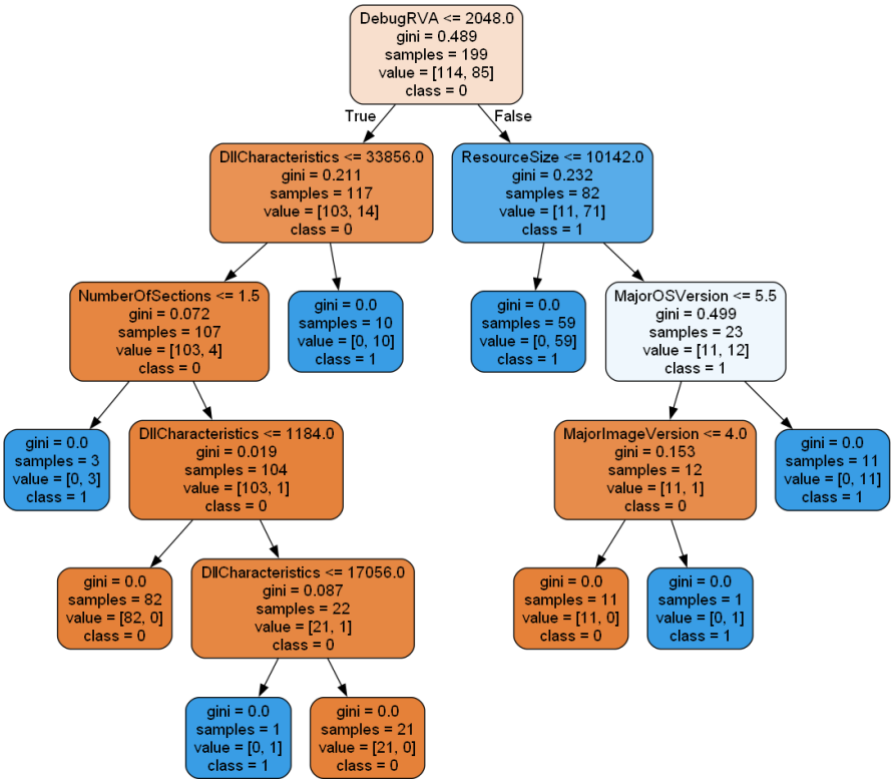


Figura 2: Arquitectura del Algoritmo Decision Tree.

Arquitectura Random Forest

La arquitectura de Random Forest se compone a través de diversos árboles de decisiones, ya que cada uno de los árboles son entrenados de forma independiente realizado por un grupo aleatorio y diferentes de datos y también con sus diferentes características. A la hora de realizar la predicción final se obtiene por votación Es importante tener en cuenta que los gráficos dirigidos generados por export_graphviz son solo representaciones visuales de la arquitectura de los modelos. Para utilizar y hacer predicciones con los modelos de Decision Tree y Random Forest, se deben entrenar con un conjunto de datos y luego utilizar el modelo entrenado para realizar predicciones en nuevos datos, A continuación se muestra la arquitectura en la Figura 3. La fórmula de Random Forest es $Predicción\ Final = Clase\ mayoritaria$

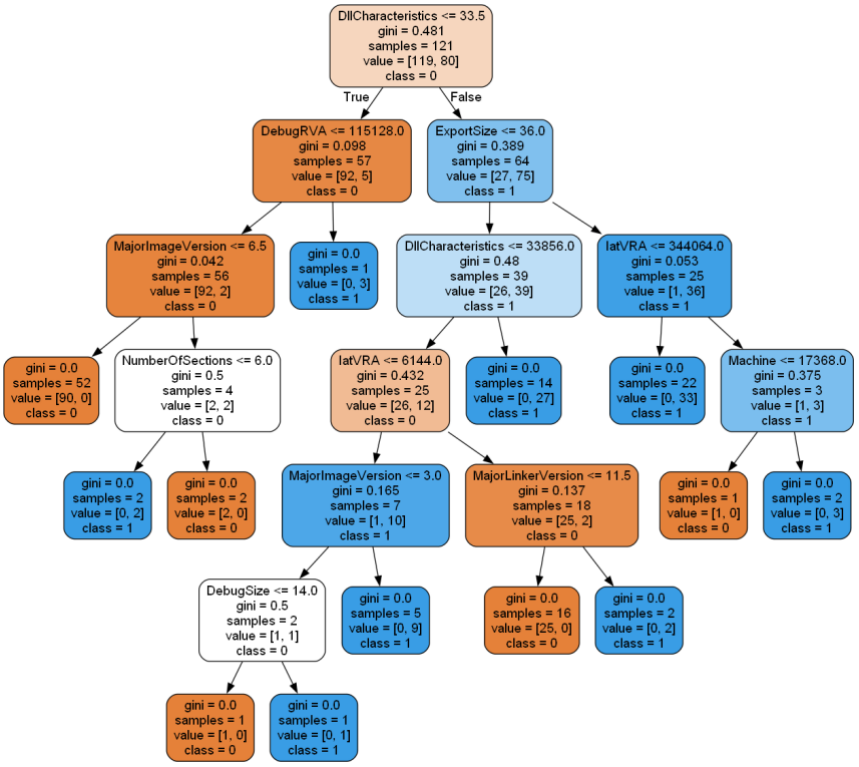


Figura 3: Arquitectura del Algoritmo Random Forest.

Arquitectura SVM

SVM es utilizado tanto como para realizar clasificaciones y regresiones. El propósito que tiene es encontrar un hiperplano óptimo para poder dividir las clases obtenidas en el espacio de características. Un hiperplano que sea óptimo es definido como el que puede maximizar una distancia entre las diferentes clases que son más cercanas, también es conocido como los vectores para el soporte ya que esos son los puntos obtenidos más cercanos hacia el hiperplano porque así se utiliza a la hora de determinar la arquitectura del SVM. El algoritmo SVM utiliza diferentes funciones a la hora de darle uso al kernel y poder mapear los datos a diferentes espacios para sus características con una dimensión mayor. Ya que eso nos puede ayudar a encontrar el hiperplano más óptimo que mejore a la hora de dividir mejor sus clases. Uno de los ejemplos más claros de las diferentes funciones que tiene el kernel es que incluye un kernel lineal, también puede ser kernel polinomial y sobre todo y más importante un kernel radial basis function (RBF). Una vez que se encuentra el hiperplano óptimo, ya que puede utilizar las predicciones para poder clasificar nuevos datos. El algoritmo

SVM clasifica los puntos de datos según el lado del hiperplano en el que se encuentren. A continuación se muestra la arquitectura en la Figura 4. La fórmula de Support Vector Machine es $f(x) = \beta_0 + \sum_{i=1}^{n} a_i . x_i$

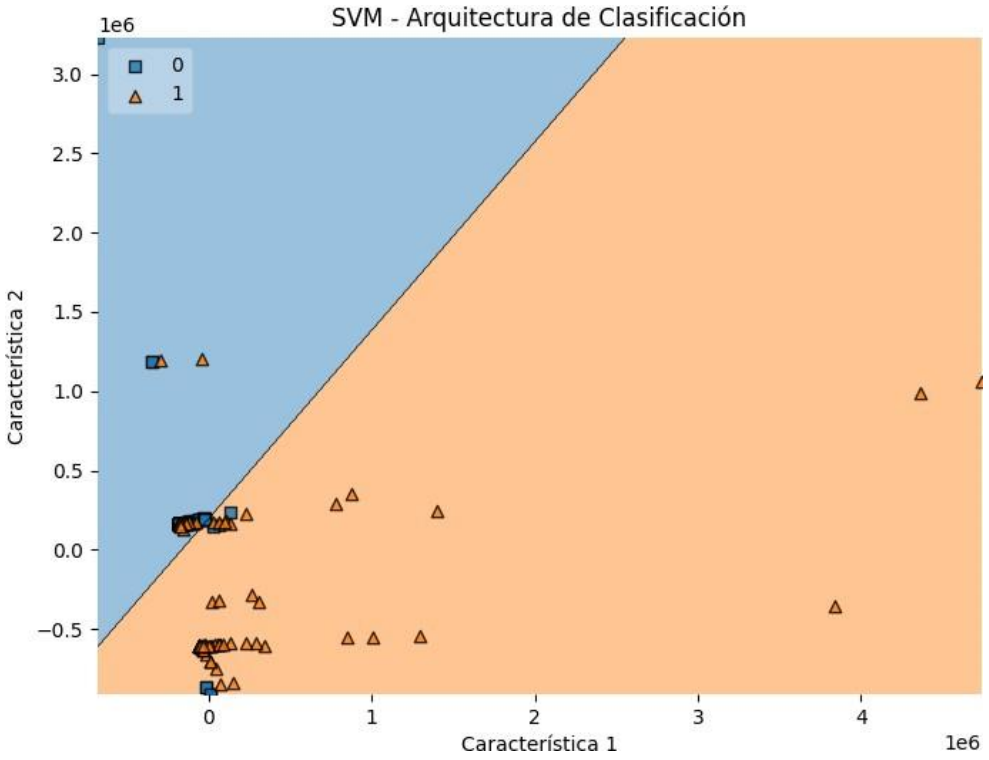


Figura 4: Arquitectura del Algoritmo SVM.

ITEM	ALGORITMOS	CITAS	MÉTRICAS			
			PRECISIÓN	ACCURACY	RECALL	F1-SCORE
1	Random forest	[25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42]	98.50%	98.50%	98.50%	98.50%
2	Super Vector Machine	[38][39][25][40][36][30][32][29] [28][43]	98.50%	98.50%	98.50%	
3	Decisión Tree	[39][25][40][36][35][32][33] [29][28] [41]	95.50%	95.50%	95.50%	
4	Naive Bayes	[25], [27], [28], [30], [32], [36], [39],				
5	Logistica Regresión					
6	CNN					

Tabla 4: Primera versión de la lista de Algoritmos.

ITEM	ALGORITMO	CITAS	MÉTRICAS			
			PRECISIÓN	ACUARACY	RECALL	F1-SCR
1	Random forest	[38][37][41][39] [34][30][35][32] [28]				
	Machine Tree	[29][28]				

Tabla 5:Versión final de la lista de algoritmos con sus métricas.

En la búsqueda de alcanzar el tercer objetivo de investigación centrado en el diseño de un método para la clasificación de ransomware, se han tomado decisiones clave. El proceso implica una cuidadosa evaluación de herramientas y recursos disponibles. Se realiza una evaluación de software, frameworks y programas de terceros, presentando en la **Tabla 5** un catálogo de software popular para la maquetación y visualización del resultado final. Esta selección es fundamental para identificar herramientas que se ajusten a los requerimientos del proyecto, ofreciendo una integración eficiente. Además, se utilizó el lenguaje de programación Python, buscando garantizar la versatilidad y capacidad efectiva para entrenamientos. La **Tabla 6** desglosa los mejores frameworks en Python respaldando esta elección. Se identifica un programa de terceros que contribuirá al proceso de diseño, proporcionando una guía clara. La culminación de esta fase se presenta ofreciendo una visión tangible del sistema de clasificación. Los mockups finales, elaborados con el software seleccionado y mostrados en la **Figura 1**, son cruciales para la visualización y planificación detallada del método, allanando el camino hacia su implementación. La **Figura 2** proporciona una visión detallada del diseño del método de clasificación de ataques ransomware, detallando cada punto que se seguirá para el desarrollo del método propuesto.

N°	*Softwares Más utilizados*
1	Balsamiq
2	Figma
3	Canva Pro

Tabla 6: Softwares más utilizados para maquetas Mockups.

N°	Frameworks y Lenguajes seleccionados
1	Django
2	Python
3	Oracle

Tabla 7: Frameworks y Lenguajes seleccionados.

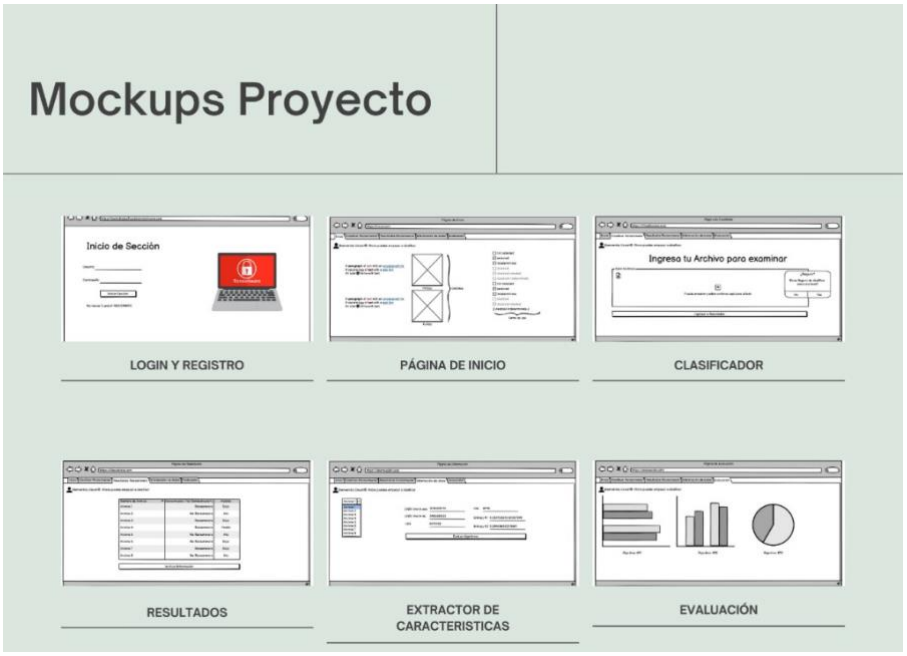


Figura 5: Mockups realizados con el software Balsamiq.

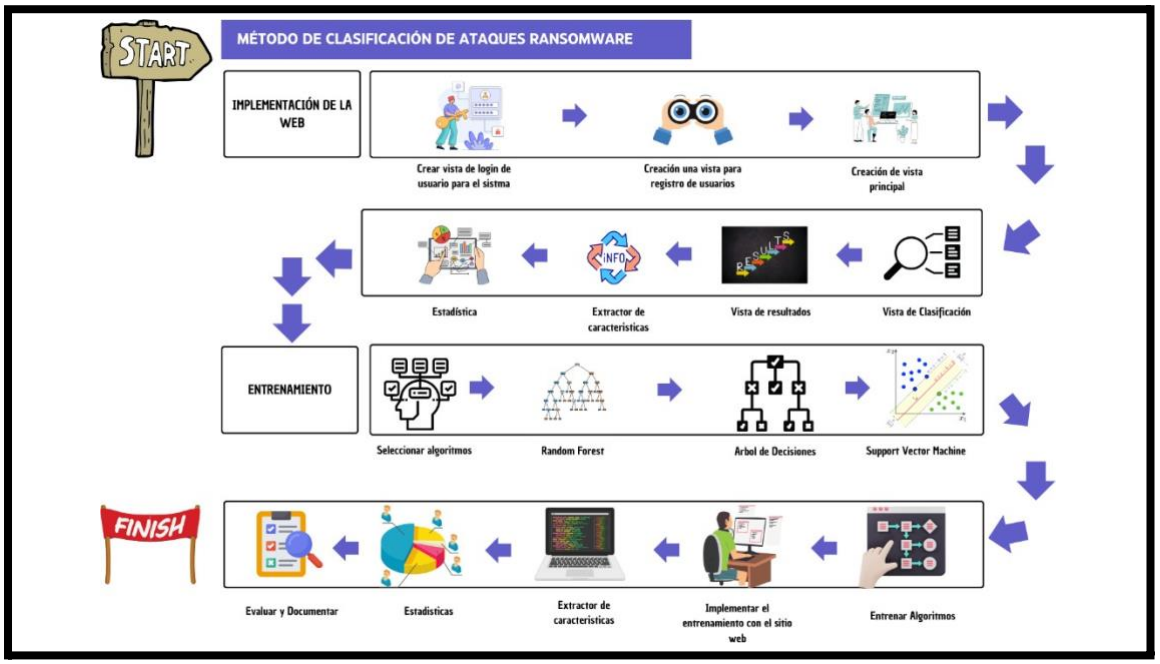


Figura 6: Método de Clasificación de Ataques Ransomware.

La implementación de la vista de Login de usuario emerge como un componente cardinal dentro de la arquitectura del sitio web dedicado a la clasificación de ransomware.

Este aspecto reviste una importancia crítica al proporcionar una barrera de seguridad que asegura el acceso controlado a diversas funciones, al tiempo que resguarda la integridad de la información vital para el funcionamiento del sistema. A pesar de la necesidad de esta medida de seguridad, se destaca la ausencia de restricciones en el proceso de registro para nuevos usuarios, posibilitando que cualquier individuo con las credenciales adecuadas pueda acceder al sitio web. La creación de una vista específica para el registro de usuarios complementa este sistema de seguridad al facilitar la adquisición de credenciales mediante un proceso estructurado de registro. Esta funcionalidad no solo simplifica el acceso para los usuarios, sino que también contribuye al establecimiento de un control meticuloso sobre la totalidad de los participantes en la plataforma. Mientras tanto, la vista principal del sitio web se consolida como una prioridad máxima en el proceso de desarrollo. En esta instancia, se proporciona información detallada sobre el método de clasificación en pocos pasos, constituyendo así un punto de verificación esencial que permite a los usuarios explorar y confirmar el contenido fundamental del sitio. La vista de clasificación, que se revela como una piedra angular del sistema, permite a los usuarios cargar archivos con extensiones específicas (.exe, .dll) para su clasificación. Este paso crítico está respaldado por un riguroso proceso de validación en el Backend que salvaguarda contra la manipulación de archivos maliciosos. No obstante, cuando se sube un archivo con otra extensión durante el proceso de clasificación, se implementa una alerta que notifica al usuario sobre la imposibilidad de llevar a cabo la clasificación debido a un inconveniente en el archivo. Posterior a la fase de clasificación, la vista de resultados desempeña un papel crucial al presentar una tabla informativa que incluye columnas como "Archivo" y "Resultado". Estos resultados son almacenados tanto en la interfaz del usuario como en el administrador de Django, permitiendo su acceso y utilización en diversas secciones del método de clasificación. La vista del extractor de características se posiciona como un elemento invaluable en el proceso de clasificación, ofreciendo una funcionalidad esencial para comprender la naturaleza intrínseca de los archivos cargados. En esta instancia, se exhibe un combobox que enumera todos los archivos previamente subidos al clasificador de

archivos, proporcionando a los usuarios la capacidad de seleccionar el archivo del cual desean conocer las características. La implementación de estadísticas en la web se erige como una función crucial para evaluar la efectividad del entrenamiento de los algoritmos. Esta implementación visualiza los resultados del entrenamiento mediante gráficos detallados que representan métricas asociadas a cada algoritmo. A través de un combobox, los usuarios pueden seleccionar uno de los algoritmos entrenados para visualizar la correspondiente gráfica, brindando así una visión profunda de la eficacia de cada algoritmo en términos de clasificación. La selección de algoritmos se presenta como un paso estratégico en la construcción de la base del trabajo. La cuidadosa elección de los tres mejores algoritmos se fundamenta en métricas y porcentajes altos, asegurando un rendimiento óptimo en la clasificación. Este proceso no solo se limita a consideraciones técnicas, sino que también incorpora la ponderación de algoritmos que han demostrado eficacia en contextos específicos, como imágenes o enfermedades. Tras la selección, se inicia el entrenamiento de los 3 algoritmos seleccionados. Este entrenamiento se lleva a cabo con conocimientos sólidos de aprendizaje automático, ajustando cuidadosamente los hiperparámetros para obtener métricas destacadas. Se enfatiza la importancia de utilizar conjuntos de datos significativos, y en este caso, se realiza una selección que abarca más de 60 mil muestras, equilibrando benignas y malignas para garantizar una representación adecuada. La implementación del entrenamiento en el sitio web adquiere un protagonismo significativo, ya que otorga vida al clasificador. En esta fase, se incorpora el hiperparámetros Joblib al entrenamiento para almacenar los resultados en archivos con extensión .joblib. El método de votación se introduce, permitiendo que los tres algoritmos tomen decisiones colaborativas basadas en entrenamientos previos. Si al menos dos algoritmos están de acuerdo en que un archivo es ransomware, el resultado será 1; de lo contrario, será 0. Cada algoritmo cuenta con su propio archivo joblib, y se implementa una función de extracción de características internas que permite subir archivos al sistema, extraer características internamente y realizar predicciones, cuyos resultados se envían a la página de resultados. La implementación del extractor de características se alinea con el proceso anteriormente

descrito. Esta fase implica una extracción interna de características de cada archivo subido a través de la primera pestaña del sitio web, destacando la importancia de correlacionar las extracciones con las características específicas de cada archivo. Finalmente, la implementación de estadísticas en el web cumple una función vital para evaluar la efectividad del entrenamiento de los algoritmos. La visualización de los resultados a través de gráficos detallados ofrece información valiosa sobre la efectividad del entrenamiento, permitiendo ajustes en hiperparámetros o la celebración de resultados excepcionales. La documentación cierra este intrincado proceso, asegurando el registro detallado de cada paso y decisión. Este compendio de información se convierte en un recurso esencial para futuras referencias, garantizando la trazabilidad y comprensión del desarrollo y la implementación de cada componente del sistema de clasificación de ransomware.

En el cuarto y último objetivo, el proceso inicia con la elección precisa de las tecnologías para el entrenamiento de los algoritmos DT, RF y SVM, ajustando cuidadosamente los hiperparámetros para lograr una clasificación efectiva del ransomware y generando un archivo Joblib, esencial para futuras predicciones y clasificaciones. Antes del entrenamiento se realiza una exploración detallada de los datos para comprender su estructura, identificar valores atípicos y gestionar datos faltantes, garantizando la coherencia del conjunto de datos. Posteriormente, se optimizan los datos a través de la selección de atributos, eliminando los datos irrelevantes para reducir la complejidad del entrenamiento. Con los datos depurados y las características seleccionadas, se procede a la fase de prueba con un conjunto de datos independiente, "Dataset_De_Pruebas", que representa el 20% del conjunto original y excluye la columna Benign, permitiendo evaluar la eficacia de la predicción. Tras confirmar la alta eficacia del modelo, se implementan los modelos Joblib en el framework Django previamente establecido, desarrollando las vistas del Frontend y añadiendo diversas funciones al archivo view.py en el Backend. Entre estas funciones, destacan la extracción de características y la clasificación mediante el método Voting, el cual toma una decisión sacando un promedio de las votaciones de cada algoritmo para llegar a una decisión final. Además, se implementa una función que garantiza el

funcionamiento del select en la tercera vista, mostrando todos los archivos subidos en el clasificador y permitiendo la extracción de todas sus características, asegurando un análisis robusto y completo. Donde en la **Figura 3** nos muestra parte del entrenamiento de los datos como se mencionó previamente y detallado en el cronograma de tareas en la **Figura 4**, garantiza una ejecución ordenada y efectiva del proceso de clasificación, abarcando fases como la planificación hasta la exhibición de los resultados de cada algoritmo entrenado en la **Tabla 7** y su comparación con investigaciones anteriores para una visión mejorada, concluyendo con la visualización del código de las vistas del proyecto en la **Figura 5**, todo en línea con el diseño del método planteado anteriormente.

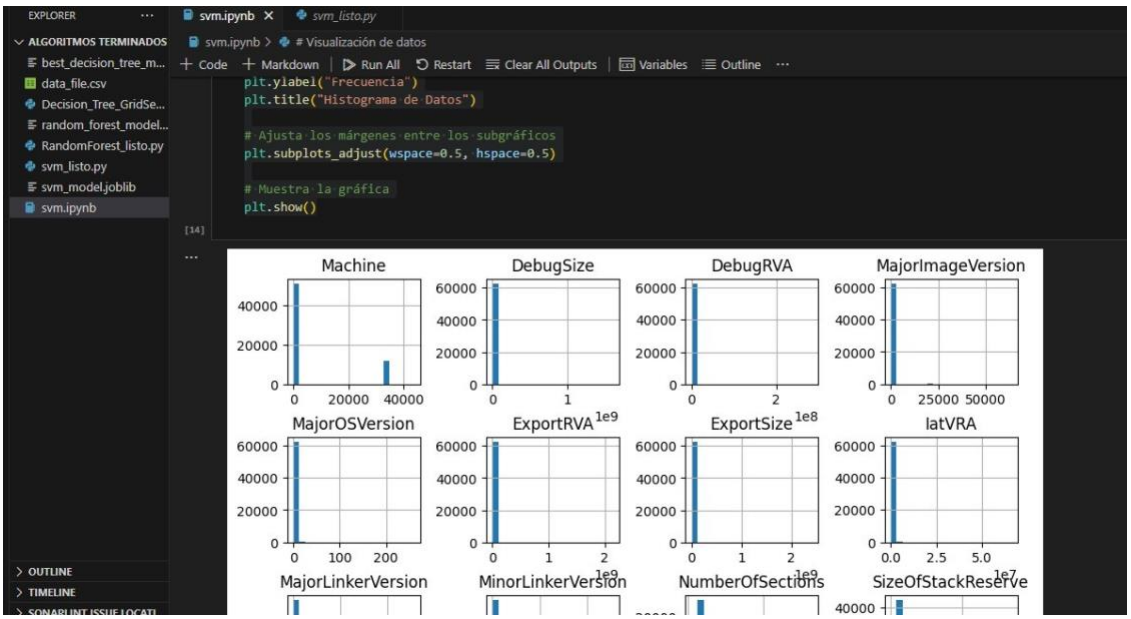


Figura 7: Vista general del entrenamiento de algoritmos.

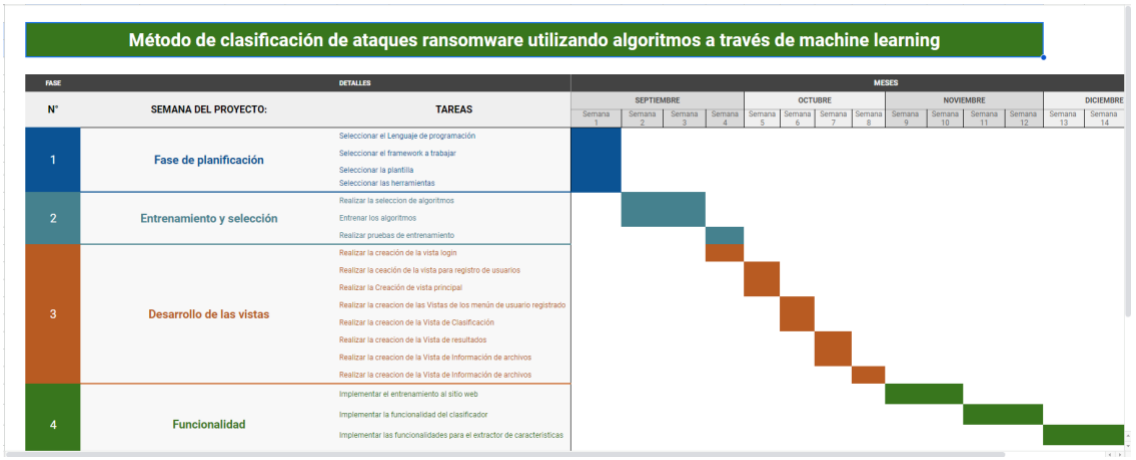


Figura 8: Cronograma de actividades para el sistema.

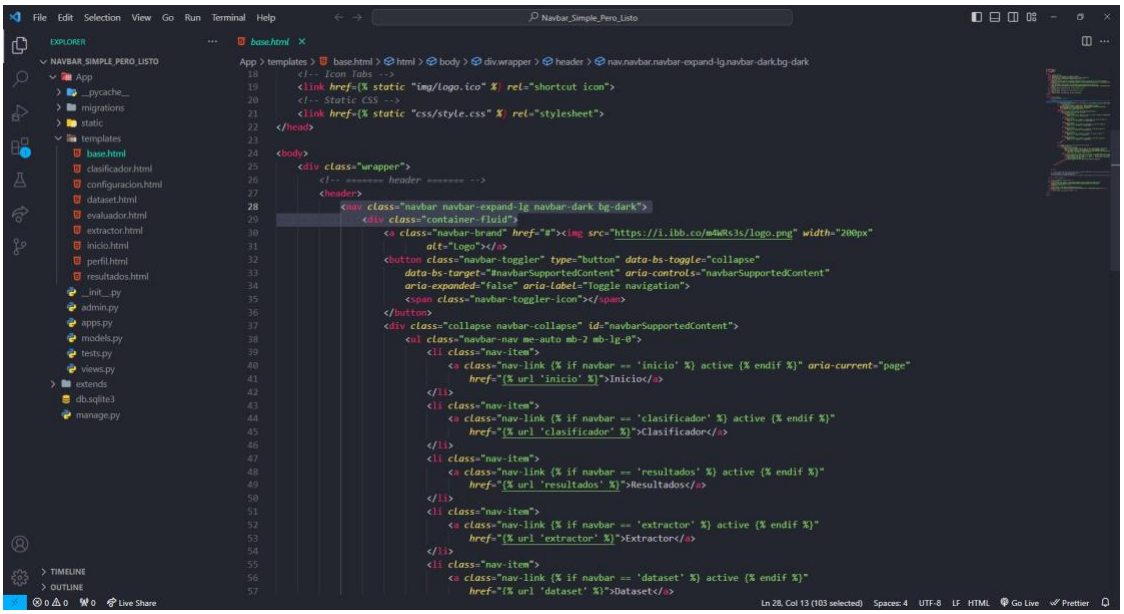


Figura 9: Vista General de la estructura del proyecto.

ITEM	ALGORITMO	MÉTRICAS			
		PRECISIÓN	ACCURACY	RECALL	F1-SCRO
1	Random forest	99.60%	9		
2	Decisión Tree	99.40%	9		
	Machine				

Tabla 8: Resultados de los algoritmos entrenados.

III. RESULTADOS Y DISCUSIÓN

3.1 RESULTADOS

En esta sección, se muestran las métricas del gasto a través de recursos computacionales y el tiempo que nos da una respuesta, primordialmente en relación con el procesamiento que se requiere el dataset. Luego, se profundiza al entrenar los algoritmos de ML, específicamente el Decision Tree, el Random Forest y el SVM, desglosando su desempeño en términos de eficiencia temporal y utilización de recursos. La meticulosa evaluación de las métricas que utilizaremos nos muestra una visión de su rendimiento global que tiene el sistema, sino que también se da a identificar áreas de mejora potencial y optimización. En consecuencia, este análisis detallado se convierte en una herramienta

esencial y así poder comprender y perfeccionar la ejecución de los procesos que tiene el aprendizaje automático, contribuyendo a unas mejores decisiones informadas y a la eficacia continua de las operaciones computacionales.

Tabla 9: Especificaciones técnicas del computador utilizado para el entrenamiento.

ETAPA	CONSUMO CPU	CONSUMO RAM	TIEMPO DE RESPUESTA
Procesamiento de caracteristicas	35.09%	12MB	240 Segundos

Nota: Los datos fueron calculados considerando las especificaciones técnicas del computador designado para el entrenamiento. **Fuente: Elaboración propia**

La Tabla 8 muestra el consumo del CPU y la memoria RAM y el tiempo de respuesta necesario para el procesamiento de las características. Esta etapa abarca desde la carga de los datos para el entrenamiento hasta la validación. Primero, se carga el dataset a entrenar. A continuación, se realiza la eliminación de etiquetas que no son de tipo entero para el entrenamiento (FileName, md5Hash, BitcoinAddresses), utilizando el método df.drop de la librería Pandas. También se asignan las variables (x, y). Luego, se realiza una división del dataset que se divide en el 80% para entrenar los algoritmos de ML y solo el 20% que es para realizar las pruebas y realizar los resultados. A continuación, se definen los parámetros para la búsqueda cuadrícula y se crea la instancia del clasificador. Posteriormente, se crea la instancia de Grid Search y se entrena el modelo. En cuanto al análisis del rendimiento, podemos observar que de los 2.70 GHz que tiene actualmente el computador, se utilizó el 35.09% para el procesamiento de características. Asimismo, de los 24 GB de memoria RAM, se utilizaron 12 MB para completar dicha etapa. Por último, en cuanto al tiempo de respuesta, el proceso completo pasó los 4 minutos con 53 segundos para procesar el dataset con 62486 características. Las unidades de medida existentes y consideradas se encuentran en Tabla 10 y Tabla 11.

Tabla 10: Especificaciones técnicas del computador entrenando cada algoritmo.

Medida de consumo de CPU, memoria RAM y tiempo de respuesta empleado por los 3 algoritmos seleccionados de aprendizaje automático.

ALGORITMOS EMPLEADOS	CONSUMO CPU	CONSUMO RAM	TIEMPO DE RESPUESTA
Decision Tree	8%	0.82MB	106.98 seg
Random Forest	29.5%	6.12 MB	5.87 seg
Support Vector Machine	15.0%	6.09MB	82.37seg

Nota: Los datos fueron calculados considerando las especificaciones técnicas del computador designado para el entrenamiento. **Fuente: Elaboración propia**

La **Tabla 9** nos muestra los datos relativos al consumo del CPU, el uso de memoria RAM y los tiempos de respuesta necesarios durante la ejecución de los diferentes algoritmos de ML escogidos. De las 62,486 características que contiene el dataset, se asignó el 80% para el proceso de entrenar los algoritmos, también se tomó el 20% de todos los datos para poder realizar diversas pruebas. En la fase de entrenamiento de los algoritmos, se implementó el aprendizaje automático.

Durante el proceso de análisis, se empleó el algoritmo Decision Tree para evaluar el conjunto de datos. Los resultados revelaron un consumo de CPU del 8%, lo que representa que se usó de manera eficiente, de acuerdo a los recursos disponibles que se encuentran a través del computador. Además, únicamente se utilizó 0.82 MB de memoria RAM, lo que indica una baja demanda de memoria para completar el proceso. Sin embargo, se observó que el tiempo de respuesta fue de 106.98 segundos, lo que podría considerarse un tiempo relativamente largo en comparación con otros algoritmos evaluados. Estos hallazgos destacan la eficacia en el uso de recursos, pero también indican una posible necesidad de optimización en el tiempo de respuesta.

En el análisis del conjunto de datos, se aplicó el algoritmo Random Forest para obtener resultados comparativos. Durante la ejecución, se evidenció un consumo de CPU del 29.5%, lo que implica un mayor uso de los recursos disponibles en el computador.

Asimismo, se utilizó un total de 6.12 MB de memoria RAM, lo que representa una demanda significativa en comparación con otros algoritmos. Sin embargo, los resultados mostraron un tiempo de respuesta de tan solo 5.87 segundos, lo que indica una eficiencia notable en términos de velocidad de procesamiento. Estos resultados resaltan la capacidad del algoritmo Random Forest para realizar cálculos rápidos a costa de un mayor consumo de recursos.

Durante el análisis del conjunto de datos, se implementó el algoritmo Support Vector Machine para evaluar su desempeño. Durante la ejecución, se observó un consumo de CPU del 15.0%, lo que indica un uso moderado de los recursos del computador. En términos de memoria, se utilizó un total de 6.09 MB, lo que representa una demanda similar a la del algoritmo Random Forest. Sin embargo, el tiempo de respuesta fue de 82.37 segundos, lo que indica un rendimiento intermedio en comparación con los otros algoritmos evaluados. Estos resultados evidencian que el algoritmo Support Vector Machine ofrece un equilibrio entre lo que consumen los recursos y también el tiempo que responde la máquina, siendo una opción a considerar cuando se busca un compromiso entre eficiencia y velocidad.

Tabla 11: Tabla de unidades de medida del consumo del GPU.

VALOR	SÍMBOLO	NOMBRE
$10^3 Hz$	kHz	Kilohercio
$10^6 Hz$	MHz	Megahercio
$10^9 Hz$	GHz	Gigahercio

Tabla de Unidades de medida del consumo del GPU. Fuente [47]

Tabla 12: Tabla de unidades de medida del consumo de RAM.

UNIDAD	SÍMBOLO	CANTIDAD EN BITS	EQUIVALENCIA
1 Byte	B	$8 = 2^8$	8 Bits

1 Kilobyte	KB	2^{10}	1024 Bytes
1 Megabyte	MB	2^{20}	1024 Kilobytes
1 Gigabyte	GB	2^{30}	1024 Megabytes
1 Terabyte	TB	2^{40}	1024 Gigabytes

Tabla de Unidades de medida del consumo de RAM. Fuente: [48]

A continuación, mostramos los resultados obtenidos, de acuerdo con las métricas de rendimiento por cada algoritmo para clasificar ransomware. Específicamente, se presenta información relacionada con los resultados obtenidos del entrenamiento realizado a cada algoritmo.

Tabla 13: Resultados de los algoritmos entrenados.

ALGORITMOS	ACCURACY	F1-SCORE	RECALL	PRECISIÓN
Decision Tree	99.4%	99.4%	99.4%	99.4%
Random Forest	99.6%	99.6%	99.6%	99.6%
Super Vector Machine	87.50%	87.50%	87.40%	99.96%

Nota: Los datos calculados en la tabla son parte de los resultados del entrenamiento y validación. Fuente: **Elaboración propia.**

A través de la Tabla 12 Tiene como medidas de rendimiento para cada uno de estos 3 algoritmos de ML utilizados. Estos resultados fueron obtenidos con la base de una matriz de confusión para poder obtener los resultados de los algoritmos propuestos.

Decision Tree

En la **Figura 10** Se puede observar la matriz de confusión, que nos muestra un resultado detallando las predicciones obtenidas con el algoritmo Decision Tree. utilizando Aprendizaje Automático.

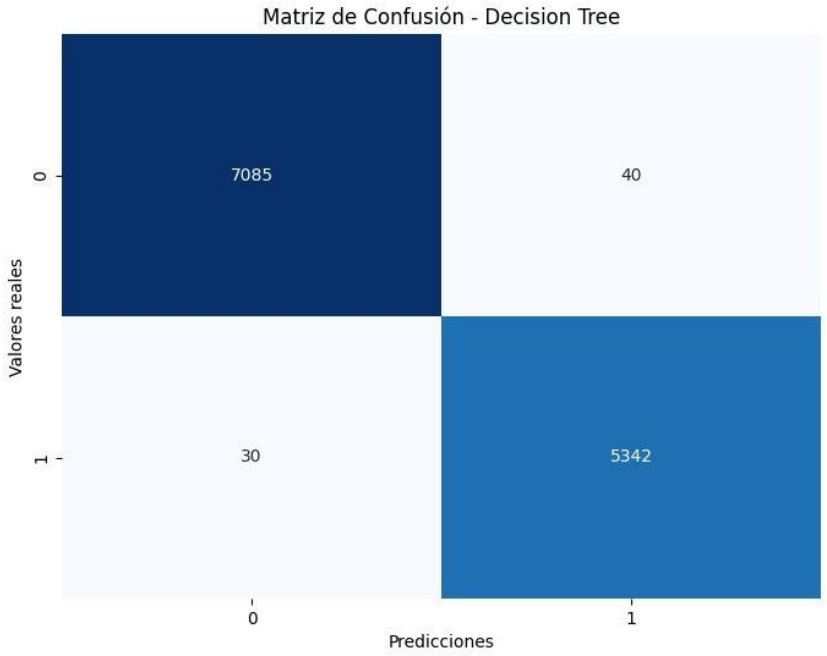


Figura 10: Matriz de Confusión de Predicciones del Algoritmo Decision Tree.

Los resultados que se tienen de acuerdo al análisis a la hora de aplicar el algoritmo Decision Tree, evidenció un comportamiento altamente efectivo y preciso en cuanto a la clasificación de datos. Las métricas obtenidas, es decir, la precisión, Accuracy, recall y F1-Score, todas ellas con un valor de 99.4%, manifiestan una gran capacidad del algoritmo para identificar y clasificar correctamente tanto los casos benignos como los malignos. En una muestra total de 12498 instancias, el algoritmo fue capaz de clasificar correctamente 5340 casos como benignos y 7085 casos como malignos. Este grado de efectividad a través de su clasificación es indicativo de un rendimiento excepcionalmente alto, lo que sugiere una robustez y fiabilidad significativa del algoritmo en este contexto de aplicación específico. Sin embargo, aunque la tasa de error fue mínima, se presentaron algunos casos de clasificación incorrecta. Concretamente, 32 muestras fueron clasificadas erróneamente como no ransomware y 40 como ransomware. Aunque estas inexactitudes representan un porcentaje muy pequeño del total de casos, su existencia subraya la importancia de seguir refinando y optimizando el algoritmo para mejorar aún más su precisión y confiabilidad. En resumen, los resultados indican que el algoritmo Decision Tree ha demostrado ser altamente efectivo, pero como con cualquier modelo de aprendizaje automático, siempre hay espacio para mejoras y refinamientos adicionales.

Random Forest

En la **Figura 7** Se puede observar la matriz de confusión, que nos muestra un resultado detallando las predicciones obtenidas con el algoritmo Random Forest. utilizando Aprendizaje Automático.

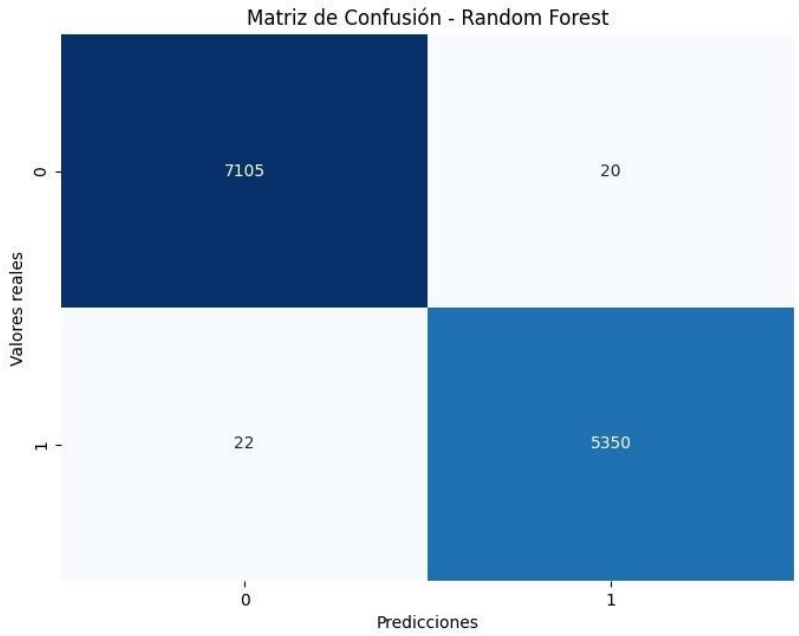


Figura 11: Matriz de Confusión de Predicciones del Algoritmo Random Forest.

La aplicación del algoritmo Random Forest en nuestra muestra de datos arrojó resultados igualmente impresionantes, destacando su capacidad para clasificar eficientemente los datos en cuestión. Las métricas obtenidas, a saber, la precisión, accuracy, recall, F1-Score, todas registraron un valor de 99.6%, lo que indica una gran eficacia del algoritmo para discriminar adecuadamente entre los casos benignos y malignos. En términos numéricos, de las 12498 muestras utilizadas, el algoritmo logró clasificar correctamente 7103 casos como benignos y 5349 casos como malignos. Este alto grado de precisión en la clasificación reafirma la excelencia del rendimiento del algoritmo Random Forest en este contexto específico. Sin embargo, a pesar de su alta efectividad, aún se registraron pequeños márgenes de error. En concreto, 23 muestras fueron clasificadas de manera incorrecta como no ransomware y 22 muestras como ransomware. Aunque estos errores representan una fracción mínima del total de casos, sirven para recordar la continua

necesidad de optimización y perfeccionamiento del algoritmo para mejorar su precisión y fiabilidad.

Super Vector Machine

En la Figura 8 Se puede observar la matriz de confusión, que nos muestra un resultado detallando las predicciones obtenidas con el algoritmo Decision Tree. utilizando Aprendizaje Automático.

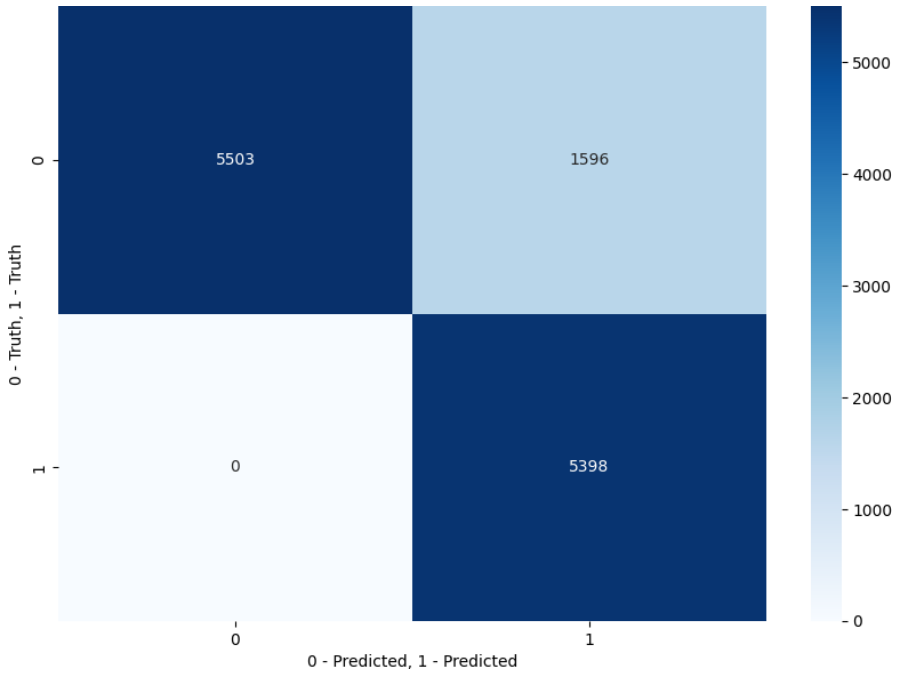


Figura 12: Matriz de Confusión de Predicciones del Algoritmo SVM.

El análisis de los resultados derivados del algoritmo Super Vector Machine (SVM) sus resultados son relativamente diferentes comparando con los otros algoritmos. Las métricas de Accuracy 87.50%, Precisión: 87.50%, Recall: 87.40%, F1-Score: 99.96%. Aunque estos porcentajes aún indican una capacidad razonable de clasificación, son notablemente inferiores a los diversos resultados que se obtuvieron previamente en los algoritmos anteriormente mencionados, los cuales son: Decision Tree y Random Forest. En términos absolutos, de las 12498 muestras utilizadas, el algoritmo SVM logró clasificar correctamente 5503 casos como benignos y 5398 casos como malignos. Sin embargo, se observó un margen de error más amplio, con 1596 muestras que fueron erróneamente clasificadas como no ransomware y 0 muestras incorrectamente identificadas como

ransomware. Estos errores representan una proporción significativa del total de casos, lo que subraya las limitaciones inherentes al algoritmo SVM en este contexto particular. Estos resultados resaltan la necesidad de una mayor optimización y ajustes para mejorar la precisión y fiabilidad del algoritmo Super Vector Machine.

Figura 13: Evaluación de la efectividad de los algoritmos

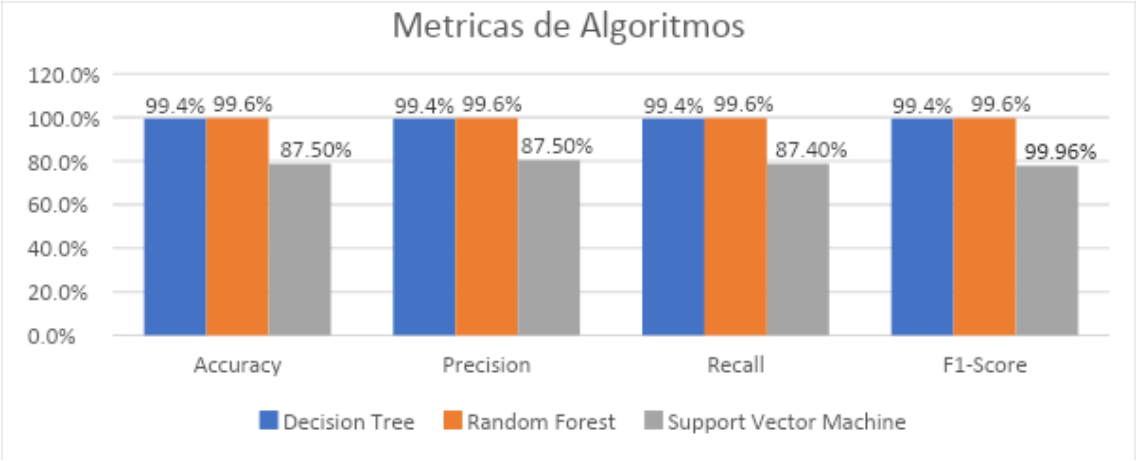


Figura 9. Se presenta la evaluación de la efectividad de los algoritmos anteriormente entrenados para la clasificación. Fuente: Elaboración propia.

Adicionalmente, en este apartado se proporciona un análisis detallado de las métricas de rendimiento correspondientes a cada uno de los algoritmos de Machine Learning (ML) implementados. Este análisis se realiza tomando en cuenta cada una de las clases, es decir, para cada tipo de muestra en nuestro conjunto de datos.

Resultados para la predicción con el algoritmo Decision Tree

Figura 14: Resultados eficaces de predicciones del Algoritmo Decisión Tree.

RESULTADOS FINALES DE LA PREDICCIÓN	
Decision Tree	
Total de datos Ingresados	12497
Total de datos Predecidos como Benignos	5401
Total de datos Predecidos como Ransomware	7096

Total de datos equivocados	12
Tasa de Predicción correcta	99.8%

Nota. Se agregaron 12497 datos del dataset de pruebas siendo Predecidos 5401 como Benignos y 7096 como Ransomware de forma correcta. Se tuvo una tasa de equivocación de 12 muestras de error del algoritmo Decision Tree lo que equivale al 0.20% y una tasa de predicción correcta del 99.80%.

Predicción de Datos por Random Forest

Figura 15:Resultados eficaces de predicciones del Algoritmo Random Forest.

RESULTADOS FINALES DE LA PREDICCIÓN	
Random Forest	
Total de datos Ingresados	12497
Total de datos Predecidos como Benignos	5423
Total de datos Predecidos como Ransomware	7074
Total de datos equivocados	56
Tasa de Predicción correcta	99.09%

Nota. Se agregaron 12497 datos del dataset de pruebas siendo predecidos 5423 como Benignos y 7074 como Ransomware de forma correcta. Se tuvo una tasa de error de 56 muestras con el algoritmo Random Forest que equivale al 0.91% y una tasa de predicción correcta del 99.09%.

Predicción de Datos por Super Vector Machine

Figura 16:Resultados eficaces de predicciones del Algoritmo Super Vector Machine.

RESULTADOS FINALES DE LA PREDICCIÓN	
Super Vector Machine	
Total de datos Ingresados	12497
Total de datos Predecidos como Benignos	5750

Total de datos Predecidos como Ransomware	6749
Total de datos equivocados	710
Tasa de Predicción correcta	92.16%

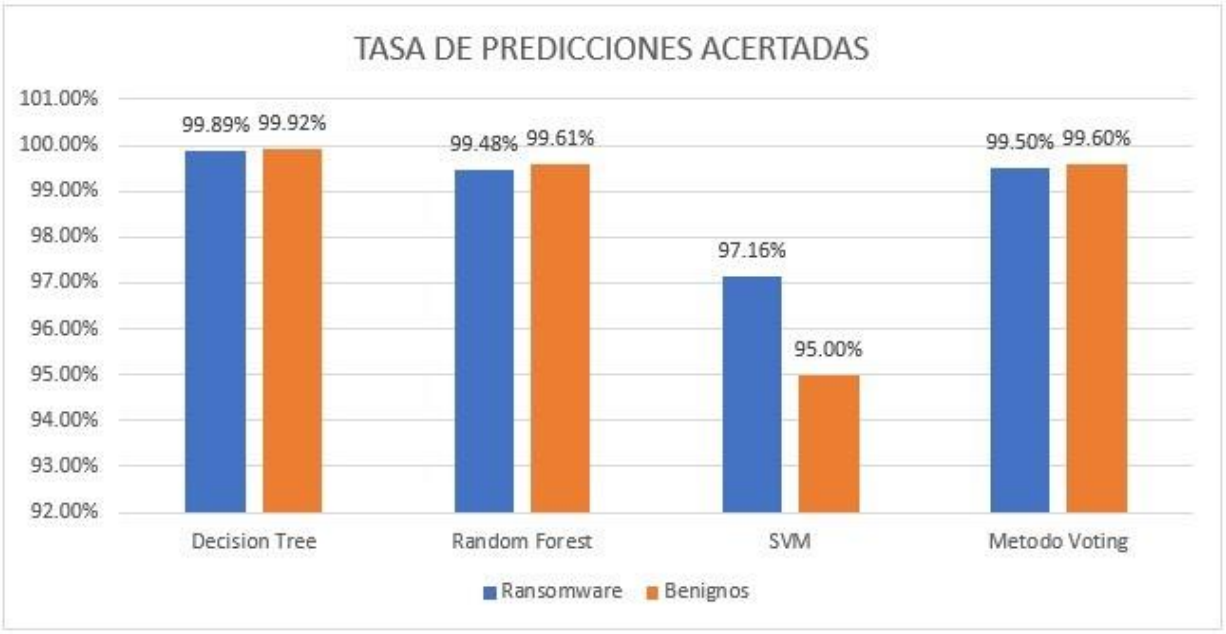
Nota. Se agregaron 12497 datos del dataset de pruebas siendo predecidos 5750 como Benignos y 6749 como Ransomware de forma correcta. Se tuvo una tasa de error de 710 muestras con el algoritmo Random Forest que equivale al 7.84% y una tasa de predicción correcta del 92.16%.

Predicción de Datos por Método Voting

RESULTADOS FINALES DE LA PREDICCIÓN	
Método Voting	
Total de datos Ingresados	12497
Total de datos Predecidos como Benignos	5424
Total de datos Predecidos como Ransomware	7073
Total de datos equivocados	58
Tasa de Predicción correcta	99.05%

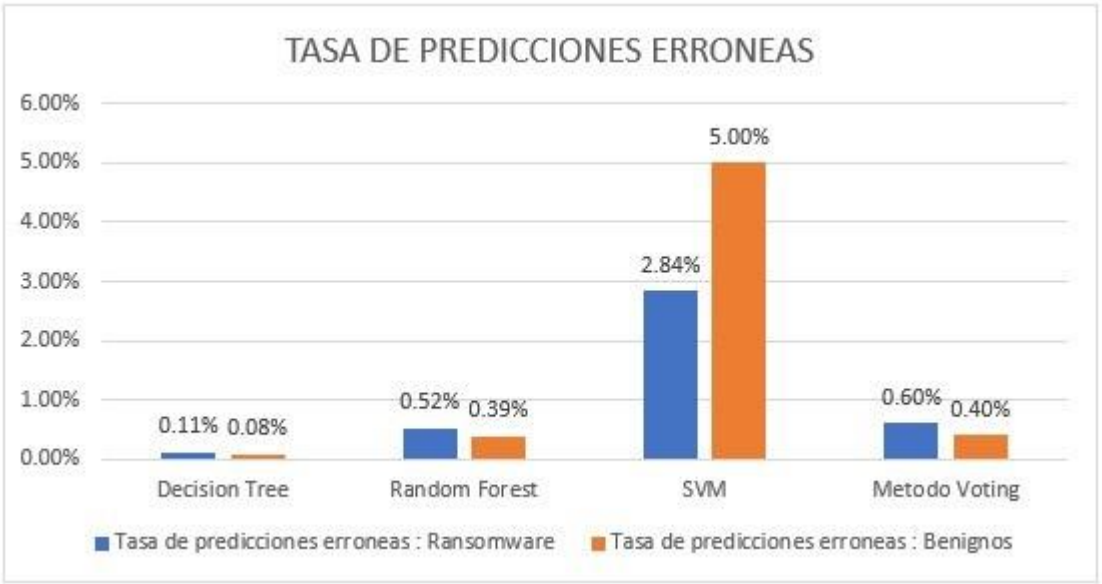
Nota. Se agregaron 12497 datos del dataset de pruebas siendo Predecidos 5424 como Benignos y 7073 como Ransomware de forma correcta. Se tuvo una tasa de error de 58 muestras utilizando el Método Voting que equivale al 0.95% y una tasa de predicción correcta del 99.05%.

Figura 17: Rendimiento de predicción de Algoritmos.



Nota. Esta figura nos muestra el nivel de predicción Correcta de archivos ransomware y benignos de cada Algoritmo.

Figura 18: Tasa de error en las Predicciones.



Nota. Esta figura muestra la tasa de predicciones erróneas de todos los algoritmos tanto como benignas y malignas

3.2 DISCUSIÓN

Los hallazgos presentados en la investigación actual, se enfocan en la clasificación de ransomware mediante la implementación de algoritmos de ML, abren nuevas vías de perspectiva en el diseño de herramientas avanzadas para fortalecer la ciberseguridad en el futuro. A través de la aplicación ingeniosa y meticulosa de varios algoritmos de aprendizaje automático, se han descubierto resultados alentadores que prometen un avance significativo en la lucha contra el ransomware. Estos resultados, medidos a través de su precisión y también recall en la clasificación de archivos de ransomware, destacan no solo la eficacia de los algoritmos utilizados, sino también su potencial para ser optimizados y adaptados en función de las crecientes amenazas cibernéticas.

Nuestra meticulosa evaluación ha revelado que los algoritmos de ML fundamentados en la clasificación exhibieron resultados sobresalientes, particularmente en términos de precisión y 'recall' con un impresionante 0.996, destacando entre otros enfoques utilizados. Este logro respalda la eficacia de los modelos de ML más frecuentes en la tarea de clasificación de ransomware, fortaleciendo su posición como herramientas esenciales en este ámbito. El significado de este hallazgo se amplifica aún más dado el creciente enfoque de la literatura científica que subraya el potencial ilimitado del aprendizaje automático en el dominio de la ciberseguridad. Este descubrimiento no sólo valida la relevancia de los algoritmos de aprendizaje automático existentes, sino que también establece un precedente para la exploración de enfoques más innovadores y eficientes para mejorar la ciberseguridad.

Es fundamental resaltar las limitaciones de este estudio. La restricción en el poder computacional podría haber influido en la eficiencia de los algoritmos de ML empleados para la clasificación de ransomware. A ello se suma la limitación de los conjuntos de datos, especialmente en términos de las características disponibles, lo que podría haber afectado la precisión de nuestras clasificaciones. Estas limitaciones, que representan retos sustanciales, no sólo destacan la importancia de contar con una mayor capacidad de procesamiento y conjuntos de datos más detallados, sino que también orientan las futuras

investigaciones hacia la mejora de la eficacia de los algoritmos de ML en el ámbito de la ciberseguridad y la clasificación de ransomware.

Al contrastar nuestros hallazgos con los de investigaciones anteriores, observamos que los resultados de nuestro entrenamiento muestran métricas tanto como su precisión, el recall también está el F1-Score y finalmente 'accuracy' con un notable 0.996. En comparación con [8], encontramos que, aunque sus resultados son notables, estas investigaciones presentan métricas ligeramente inferiores. Por ejemplo, observamos una precisión del 99.08%, un F1-score del 99.5% y, sin embargo, una mejora en el 'recall' del 99.92%, este último superando marginalmente nuestros resultados. Un segundo trabajo previo [14] exhibe resultados muy similares a los nuestros, con la única variación significativa en el 'recall', que alcanza el 99.8%. Al examinar otras métricas, notamos que sus resultados son generalmente inferiores a los nuestros, manifestando una precisión del 99.3% y un F1-score del 99.0%. Esta diferencia en el recall puede justificarse por las variaciones en los conjuntos de datos, las metodologías de entrenamiento o las configuraciones de los algoritmos utilizados en esa investigación. Aunque existen ciertas variaciones en las métricas, nuestro enfoque demuestra una eficacia comparable, sino superior, a las estrategias implementadas en investigaciones anteriores, reforzando así el valor y la relevancia de nuestros algoritmos de ML en la clasificación del ransomware.

Las implicaciones de nuestros hallazgos sugieren que la aplicación de algoritmos de ML podría mejorar de manera notable la clasificación precisa del ransomware. Sin embargo, la transición de la investigación a su aplicación real en el ámbito de la ciberseguridad conlleva varios desafíos. Estos incluyen la integración exitosa de estos algoritmos con sistemas de información existentes, adaptándose a diversas infraestructuras de software y hardware, garantizando simultáneamente la seguridad de los datos y manteniendo su privacidad. Además, es crucial validar la eficacia de estos algoritmos en entornos reales, que podría implicar su prueba en una variedad de sistemas y redes informáticas con diferentes características y vulnerabilidades. Este proceso de validación asegura que los

algoritmos son capaces de clasificar eficientemente una amplia gama de ransomware en diversos contextos, reforzando su aplicabilidad y utilidad en el fortalecimiento de la ciberseguridad.

En este estudio se aporta una valiosa evidencia acerca de la eficacia de los algoritmos de ML mencionados previamente para la clasificación de ransomware, teniendo en cuenta las necesidades que se requieren en futuras investigaciones para abordar desafíos a la hora de implementar este proyecto en la ciberseguridad.

IV. CONCLUSIONES Y RECOMENDACIONES

4.1 CONCLUSIÓN

La selección de un dataset de ataques ransomware se realizó de manera efectiva. Esta elección de datos permitió un análisis profundo y detallado de las características comunes de dichos archivos ransomware tanto benignos como malignos, estableciendo así una base sólida para la clasificación de ransomware. La calidad y la amplitud del dataset seleccionado fueron factores decisivos para el éxito de esta investigación, permitiendo un entrenamiento más preciso y completo de la naturaleza de los ataques ransomware. Esta selección de datos ha demostrado ser un paso imprescindible en la continua lucha contra las amenazas cibernéticas que cada día evoluciona más.

Tras un riguroso proceso de análisis y evaluación, se determinó con éxito la selección de los algoritmos de ML para la clasificación de ransomware. Este estudio se basó en la eficiencia y precisión probada de dichos algoritmos en tareas de clasificación similares, lo que permitió seleccionar las opciones más adecuadas. El rendimiento demostrado de estos algoritmos para su clasificación a través de diferentes datos es más complejo también dependiendo de la capacidad para poder realizar unas grandes cantidades de datos de ataques de ransomware fueron aspectos clave en esta elección. La selección acertada de estos algoritmos ha sido fundamental, pues ha permitido desarrollar un modelo de clasificación eficaz y potente, ofreciendo una herramienta precisa para la identificación y clasificación de ataques de ransomware. Esta conclusión valida la relevancia

y el impacto de los algoritmos seleccionados en la lucha contra las amenazas de ransomware y subraya su importancia en la mejora de la ciberseguridad.

El diseño de un método innovador para la clasificación de ataques de ransomware. Este diseño, meticulosamente elaborado para optimizar el rendimiento de los diferentes algoritmos de ML seleccionados, ha demostrado su potencial en la identificación precisa y eficiente de los ataques ransomware, basándose en el dataset previamente seleccionado. Este logro representa un avance notable en el campo de la ciberseguridad, proporcionando una herramienta sólida y eficaz en la lucha contra las amenazas de ransomware. Este diseño mejora significativamente nuestras capacidades de prevención, clasificación y respuesta frente a dichas amenazas, y se perfila como un recurso valioso para fortalecer la seguridad de nuestras redes y sistemas.

La exitosa implementación del método previamente diseñado para la clasificación de ataques ransomware se tradujo en la aplicación práctica del método en entornos reales, demostrando su eficacia en la clasificación de ataques de ransomware. Esta prueba práctica ha validado la robustez del diseño y su relevancia como una herramienta valiosa en la lucha contra los ataques de ransomware. La exitosa implementación de este método no solo ha cumplido con el objetivo establecido, sino que también ha establecido un precedente para futuras investigaciones y mejoras en el ámbito de la ciberseguridad, subrayando su potencial para fortalecer aún más nuestras defensas contra las amenazas cibernéticas.

Para evaluar el método de clasificación se utilizaron algoritmos de Aprendizaje Automático, empleando diversas métricas, destacando entre ellas la Precisión. En este contexto, el algoritmo Árbol de Decisión (Decision Tree) presentó una precisión de un 99.40%, mientras que Random Forest logró una precisión ligeramente superior, alcanzando el 99.60%. En último lugar, el algoritmo SVM mostró una precisión de 87.50%, siendo el más bajo en términos de precisión. Cabe señalar que se empleó un método denominado "Voting" para la clasificación. Según los resultados obtenidos, se puede afirmar que el

algoritmo Bosque Aleatorio es el más preciso en la tarea de clasificar archivos de ransomware de manera individual

4.2 RECOMENDACIONES

Se recomienda la creación de un conjunto de datos robusto y accesible que incluya diversas familias de ransomware. Este recurso será invaluable para facilitar futuras investigaciones y permitir la clasificación de nuevas variantes que puedan surgir. Al tener un mayor conocimiento y comprensión sobre las diferentes familias de ransomware, se podrá mejorar la eficacia de los métodos de clasificación actualmente utilizados. Estos métodos perfeccionados serán de gran ayuda para las organizaciones al prevenir la infiltración de ransomware en sus archivos. Al clasificar con precisión las diferentes variantes de ransomware, las organizaciones podrán tomar medidas preventivas más eficientes y evitar la amenaza antes de que pueda causar daño.

Se recomienda llevar a cabo el proceso de entrenamiento del algoritmo Máquina de Vectores de Soporte utilizando mayor capacidad computacional. El incremento de esta capacidad permitirá manipular y procesar una cantidad más grande de datos a una velocidad superior, lo cual puede conducir a una clasificación más precisa y eficiente. Además, optimizar la computación permitirá una exploración más exhaustiva y efectiva de los hiperparámetros que son inherentes al algoritmo SVM. Cada uno de estos hiperparámetros puede ser ajustado a la hora de realizar mejor precisión a la hora de obtener el modelo. En este sentido, el uso de un poder computacional superior facilitará la identificación y aplicación de los hiperparámetros más adecuados para el algoritmo.

V. REFERENCIAS

[1] S. Nivens, A. Stock, D. M. Gardiner, and J. Lazzarotti, “RANSOMWARE: To Pay or Not to Pay?,” 2022.

[2] Latamsales, “EL ESTADO DEL RANSOMWARE 2020 1,” May 2020.

[3] D. F. Carnero Garay, M. A. Carbajal Ramos, J. Armas Aguirre, and J. M. Madrid Molina, *Modelo de gestión de riesgos de seguridad de información para mitigar el impacto en las PYMEs en Perú*. 2020.

[4] I. H. Sarker, “Machine Learning: Algorithms, Real-World Applications and Research Directions,” *SN Computer Science*, vol. 2, no. 3. Springer, May 01, 2021. doi: 10.1007/s42979-021-00592-x.

[5] Coursera, “10 Machine Learning Algorithms to Know in 2023,” Coursera. Accessed: Dec. 04, 2023. [Online]. Available: https://www.coursera.org/articles/machine-learning-algorithms

[6] C. Beaman, A. Barkworth, T. D. Akande, S. Hakak, and M. K. Khan, “Ransomware: Recent advances, analysis, challenges and future research directions,” *Comput Secur*, vol. 111, Dec. 2021, doi: 10.1016/j.cose.2021.102490.

[7] M. Techlabs, “9 Real-World Problems that can be Solved by Machine Learning,” marutitech. Accessed: Dec. 04, 2023. [Online]. Available: https://marutitech.com/problems-solved-machine-learning/

[8] S. A. Alsaif, “Machine Learning-Based Ransomware Classification of Bitcoin Transactions,” *Applied Computational Intelligence and Soft Computing*, vol. 2023, 2023, doi: 10.1155/2023/6274260.

[9] J. Zhu, J. Jang-Jaccard, A. Singh, I. Welch, H. AL-Sahaf, and S. Camtepe, “A few-shot meta-learning based siamese neural network using entropy features for ransomware classification,” *Comput Secur*, vol. 117, Jun. 2022, doi: 10.1016/j.cose.2022.102691.

[10] B. Yamany, M. S. Elsayed, A. D. Jurcut, N. Abdelbaki, and M. A. Azer, “A New Scheme for Ransomware Classification and Clustering Using Static Features,” *Electronics (Switzerland)*, vol. 11, no. 20, Oct. 2022, doi: 10.3390/electronics11203307.

[11] P. Wang, Z. Tang, and J. Wang, “A novel few-shot malware classification approach for unknown family recognition with multi-prototype modeling,” *Comput Secur*, vol. 106, Jul. 2021, doi: 10.1016/j.cose.2021.102273.

[12] M. S. Abbasi, H. Al-Sahaf, M. Mansoori, and I. Welch, “Behavior-based ransomware classification: A particle swarm optimization wrapper-based approach for feature selection,” *Appl Soft Comput*, vol. 121, May 2022, doi: 10.1016/j.asoc.2022.108744.

[13] Y. Tang, X. Qi, J. Jing, C. Liu, and W. Dong, “BHMDC: A byte and hex n-gram based malware detection and classification method,” *Comput Secur*, vol. 128, May 2023, doi: 10.1016/j.cose.2023.103118.

[14] H. Zhang, X. Xiao, F. Mercaldo, S. Ni, F. Martinelli, and A. K. Sangaiah, “Classification of ransomware families with machine learning based on N-gram of opcodes,” *Future Generation Computer Systems*, vol. 90, pp. 211–221, Jan. 2019, doi: 10.1016/j.future.2018.07.052.

[15] M. A. Abdullah, Y. Yu, K. Adu, Y. Imrana, X. Wang, and J. Cai, “HCL-Classifier: CNN and LSTM based hybrid malware classifier for Internet of Things (IoT),” *Future Generation Computer Systems*, vol. 142, pp. 41–58, May 2023, doi: 10.1016/j.future.2022.12.034.

[16] Q. A. Al-Haija and A. A. Alsulami, “High performance classification model to identify ransomware payments for heterogeneous bitcoin networks,” *Electronics (Switzerland)*, vol. 10, no. 17, Sep. 2021, doi: 10.3390/electronics10172113.

[17] X. Gao, C. Hu, C. Shan, and W. Han, “MaliCage: A packed malware family classification framework based on DNN and GAN,” *Journal of Information Security and Applications*, vol. 68, Aug. 2022, doi: 10.1016/j.jisa.2022.103267.

[18] A. Amdj3dax, “XGBOost_ransomware_Detection_and_classification,” Kaggle. Accessed: Dec. 04, 2023. [Online]. Available: https://www.kaggle.com/code/amdj3dax/xgboost-ransomware-detection-and-classification

[19] M. Mohamedcherifbsr, “RandomFore_ransomware_detection_and_classification,” Kaggle. Accessed: Dec. 04, 2023. [Online]. Available: https://www.kaggle.com/code/mohamedcherifbsr/randomfore-ransomware-detection-and-classification

[20] P. Ppsec, “GitHub - 4ppsec/virustotal-Api-v2: Python scripts to interact with the Virustotal.com public API,” GitHub. Accessed: Dec. 04, 2023. [Online]. Available: https://github.com/4ppsec/virustotal-api-v2/tree/master

[21] E. D. External Data, “IMPACT - VirusShare Dataset,” 2019, Accessed: Dec. 04, 2023. [Online]. Available: https://www.impactcybertrust.org/dataset_view?idDataset=1271

[22] M. Manabu-Hirano, “GitHub - Manabu-hirano/RANSAP: RANSAP: An open dataset of ransomware storage access patterns for training machine learning models,” GitHub. Accessed: Dec. 04, 2023. [Online]. Available: https://github.com/manabu-hirano/RanSAP

[23] Y. Ytisf, “GitHub - YTISF/TheZoo: a repository of LIVE malwares for your own joy and pleasure. TheZoo is a project created to make the possibility of malware analysis open and available to the public.,” GitHub. Accessed: Dec. 04, 2023. [Online]. Available: https://github.com/ytisf/theZoo

[24] A. Aparisot, “GitHub - Aparisot84/Sandbox-Ransomware-Analysis-Dataset: Montagem de dataset para detecção de ataques de ransomware com Cuckoo Sandbox e Python,” GitHub. Accessed: Dec. 04, 2023. [Online]. Available: https://github.com/aparisot84/Sandbox-Ransomware-Analysis-Dataset

[25] S. H. Kok, A. Abdullah, and N. Z. Jhanjhi, “Early detection of crypto-ransomware using pre-encryption detection algorithm,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 5, pp. 1984–1999, May 2022, doi: 10.1016/j.jksuci.2020.06.012.

[26] S. H. Kok, A. Azween, and N. Z. Jhanjhi, “Evaluation metric for crypto-ransomware detection using machine learning,” *Journal of Information Security and Applications*, vol. 55, Dec. 2020, doi: 10.1016/j.jisa.2020.102646.

[27] J. A. Herrera-Silva and M. Hernández-Álvarez, “Dynamic Feature Dataset for Ransomware Detection Using Machine Learning Algorithms,” *Sensors*, vol. 23, no. 3, Feb. 2023, doi: 10.3390/s23031053.

[28] H. Zuhair, A. Selamat, and O. Krejcar, “A multi-tier streaming analytics model of 0-day ransomware detection using machine learning,” *Applied Sciences (Switzerland)*, vol. 10, no. 9, May 2020, doi: 10.3390/app10093210.

[29] C. Li *et al.*, “DMalNet: Dynamic malware analysis based on API feature engineering and graph learning,” *Comput Secur*, vol. 122, Nov. 2022, doi: 10.1016/j.cose.2022.102872.

[30] 王平, “LeNet-5 卷積神經網路應用於勒索病毒分類,” 2020.

[31] S. A. Alsaif, “Machine Learning-Based Ransomware Classification of Bitcoin Transactions,” *Applied Computational Intelligence and Soft Computing*, vol. 2023, 2023, doi: 10.1155/2023/6274260.

[32] IEEE Communications Society and Institute of Electrical and Electronics Engineers, *Evaluating Shallow and Deep Networks forRansomware Detection and Classification*. 2017.

[33] IEEE Staff, *Malware Classification of Portable Executables using Tree-Based Ensemble Machine Learning*. IEEE, 2019.

[34] S. Usharani, P. M. Bala, and M. M. J. Mary, "Dynamic analysis on crypto-ransomware by using machine learning: Gandcrab ransomware," in *Journal of Physics: Conference Series*, IOP Publishing Ltd, Jan. 2021. doi: 10.1088/1742-6596/1717/1/012024.

[35] S. Agarkar and S. Ghosh, "Malware detection & classification using machine learning," in *Proceedings - 2020 IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security, iSSSC 2020*, Institute of Electrical and Electronics Engineers Inc., Dec. 2020. doi: 10.1109/iSSSC50941.2020.9358835.

[36] IEEE Computer Society. Technical Council on Test Technology, IEEE Solid-State Circuits Society, International Federation for Information Processing, and EDA Association, *2SMaRT: A Two-Stage Machine Learning-Based Approach forRun-Time Specialized Hardware-Assisted Malware Detection*. 2019.

[37] IEEE Staff, *Overview and Case Study for RansomwareClassification Using Deep Neural Network*. IEEE, 2019.

[38] M. Chemmakha, O. Habibi, and M. Lazaar, "Improving Machine Learning Models for Malware Detection Using Embedded Feature Selection Method," in *IFAC-PapersOnLine*, Elsevier B.V., 2022, pp. 771–776. doi: 10.1016/j.ifacol.2022.07.406.

[39] A. Kamboj, P. Kumar, A. K. Bairwa, and S. Joshi, "Detection of malware in downloaded files using various machine learning models," *Egyptian Informatics Journal*, vol. 24, no. 1, pp. 81–94, Mar. 2023, doi: 10.1016/j.eij.2022.12.002.

[40] F. Biondi, M. A. Enescu, T. Given-Wilson, A. Legay, L. Noureddine, and V. Verma, "Effective, efficient, and robust packing detection and classification," *Comput Secur*, vol. 85, pp. 436–451, Aug. 2019, doi: 10.1016/j.cose.2019.05.007.

[41] M. Masum, M. Jobair Hossain Faruk, H. Shahriar, K. Qian, D. Lo, and M. Islam Adnan, "Ransomware Classification and Detection With Machine Learning Algorithms," 2020.

[42] M. Al-Janabi and A. M. Altamimi, "A comparative analysis of machine learning techniques for classification and detection of malware," in *Proceedings - 2020 21st International Arab Conference on Information Technology, ACIT 2020*, Institute of Electrical and Electronics Engineers Inc., Nov. 2020. doi: 10.1109/ACIT50332.2020.9300081.

[43] V. Verma, S. K. Muttoo, and V. B. Singh, "Multiclass malware classification via first- and second-order texture statistics," *Comput Secur*, vol. 97, Oct. 2020, doi: 10.1016/j.cose.2020.101895.

[44] T. Landman and N. Nissim, "Deep-Hook: A trusted deep learning-based framework for unknown malware detection and classification in Linux cloud environments," *Neural Networks*, vol. 144, pp. 648–685, Dec. 2021, doi: 10.1016/j.neunet.2021.09.019.

[45] X. Gao, C. Hu, C. Shan, B. Liu, Z. Niu, and H. Xie, "Malware classification for the cloud via semi-supervised transfer learning," *Journal of Information Security and Applications*, vol. 55, Dec. 2020, doi: 10.1016/j.jisa.2020.102661.

[46] K. C. Roy and Q. Chen, “DeepRan: Attention-based BiLSTM and CRF for Ransomware Early Detection and Classification,” *Information Systems Frontiers*, vol. 23, no. 2, pp. 299–315, Apr. 2021, doi: 10.1007/s10796-020-10017-4.

[47] a S, “MEDIDAS DE ALMACENAMIENTO EN INFORMÁTICA,” 2016. [Online]. Available: http://unidadesdealmacenamientodeinformacion.blogspot.com.co

[48] S.a, “Introducción a la Operación de Computadoras Personales,” 2017.

VI. ANEXOS

Anexo 01: Acta de revisión de similitud de la Investigación

Reporte de similitud

NOMBRE DEL TRABAJO

Método De Clasificación De Ataques Ransomware Utilizando Algoritmos A Través De Machine Learning.doc

AUTOR

Ángel Junior / Robert Frank Bazán Carhuatanta / Perez Arica

RECUENTO DE PALABRAS

8274 Words

RECUENTO DE CARACTERES

47013 Characters

RECUENTO DE PÁGINAS

36 Pages

TAMAÑO DEL ARCHIVO

2.9MB

FECHA DE ENTREGA

Apr 29, 2024 2:23 PM GMT-5

FECHA DEL INFORME

Apr 29, 2024 2:23 PM GMT-5

2% de similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para cada base de datos.

- 1% Base de datos de Internet
- 0% Base de datos de publicaciones
- Base de datos de Crossref
- Base de datos de contenido publicado de Crossref
- 1% Base de datos de trabajos entregados

Excluir del Reporte de Similitud

- Material bibliográfico
- Material citado
- Coincidencia baja (menos de 8 palabras)

Resumen

Anexo 02: Acta de Aprobación de asesor

Universidad Señor de Sipán

ANEXO 03: ACTA DE APROBACIÓN DEL ASESOR

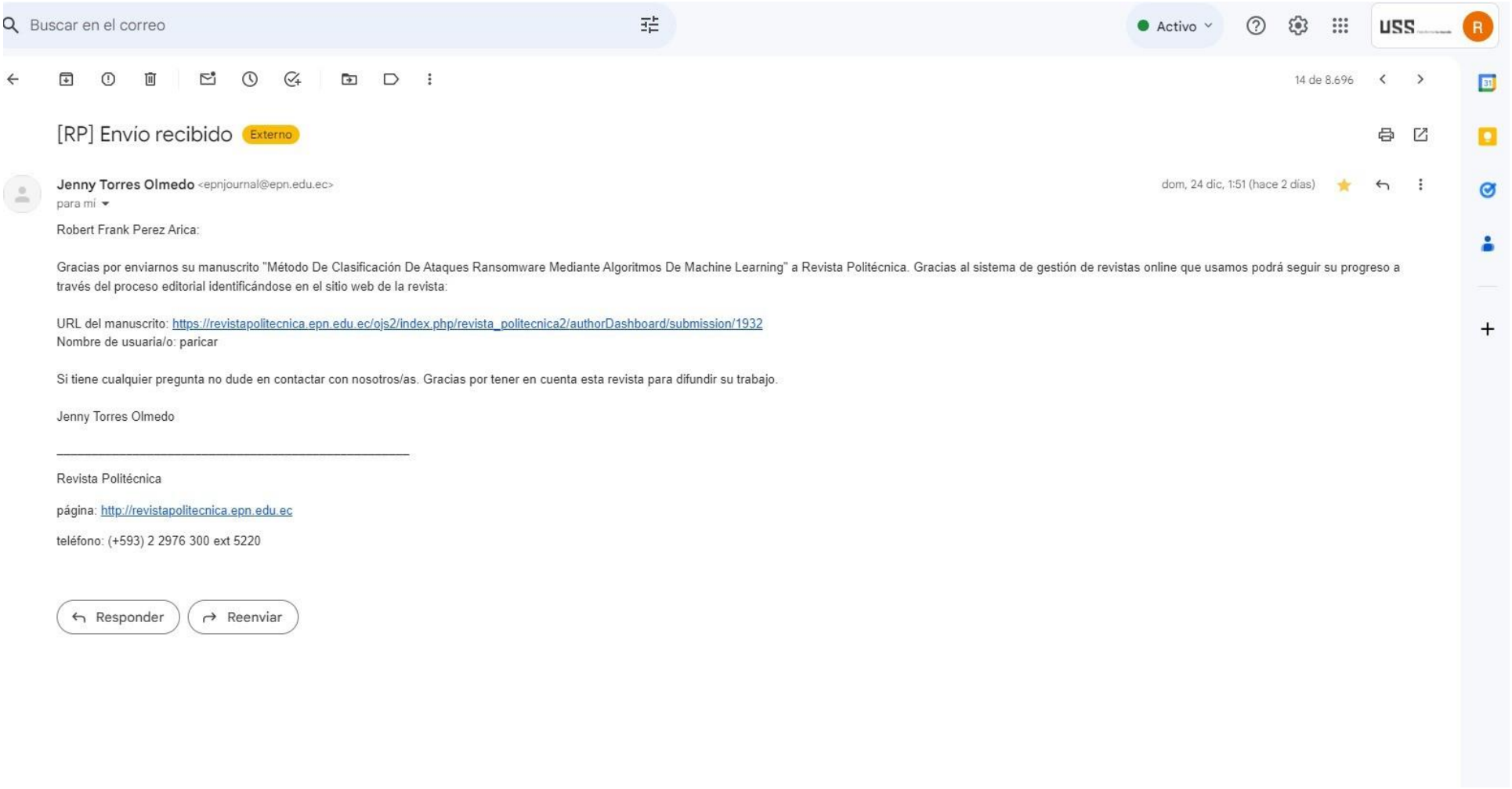
Yo **Manuel Guillermo Forero Vargas**. quien suscribe como asesor designado mediante Resolución de Facultad N° **0426-2023**, del proyecto de investigación titulado **Método de clasificación de ataques ransomware utilizando algoritmos a través de machine learning**., desarrollado por el(los) estudiante(s): **Bazán Carhuatanta Ángel Junior, Pérez Arica Robert Frank**., del programa de estudios de **Ingeniería de Sistemas**, acredito haber revisado, realizado observaciones y recomendaciones pertinentes, encontrándose expedito para su revisión por parte del docente del curso.

En virtud de lo antes mencionado, firman:

Manuel Guillermo Forero Vargas	DNI: AV702661	
Bazán Carhuatanta Ángel Junior	DNI: 74747774	
Pérez Arica Robert Frank	DNI: 72496295	

Pimentel, 20 de diciembre de 2023

Anexo 03: Carta o correo de recepción del manuscrito remitido por la revista



Buscar en el correo

Activo

14 de 8.696

[RP] Envio recibido Externo

Jenny Torres Olmedo <epnjournal@epn.edu.ec>
para mí

dom, 24 dic, 1:51 (hace 2 días)

Robert Frank Perez Arica:

Gracias por enviarnos su manuscrito "Método De Clasificación De Ataques Ransomware Mediante Algoritmos De Machine Learning" a Revista Politécnica. Gracias al sistema de gestión de revistas online que usamos podrá seguir su progreso a través del proceso editorial identificándose en el sitio web de la revista:

URL del manuscrito: https://revistapolitecnica.epn.edu.ec/ojs2/index.php/revista_politecnica2/authorDashboard/submission/1932
Nombre de usuaria/o: paricar

Si tiene cualquier pregunta no dude en contactar con nosotros/as. Gracias por tener en cuenta esta revista para difundir su trabajo.

Jenny Torres Olmedo

Revista Politécnica

página: http://revistapolitecnica.epn.edu.ec

teléfono: (+593) 2 2976 300 ext 5220

Responder Reenviar

Anexo 04: Actas de revisión por el Asesor

Universidad Señor de Sipán

ACTA DE REVISIÓN DE ASESORÍA

Yo **Forero Vargas, Manuel Guillermo** quien suscribe como asesor designado mediante Resolución de **Facultad de ingeniería, arquitectura y Urbanismo** N° **0426-2023/FIAU-USS**, del proyecto de investigación titulado **MÉTODO DE CLASIFICACIÓN DE ATAQUES RANSOMWARE UTILIZANDO ALGORITMOS A TRAVÉS DE MACHINE LEARNING.**, desarrollado por los estudiantes: **Bazan Carhuatanta, Angel Junior, Pérez Arica, Robert Frank**, del programa de estudios de **Ingeniería de Sistemas**, acreditó haber revisado, realizado observaciones y recomendaciones pertinentes tal como se detalla en el siguiente cuadro:

Fecha de revisión:	Modalidad de Asesoría:	Medio de Asesoría:	Veredicto de Asesoría:
08/09/2023	Virtual	Reunión en Google Meet	Aprobado

En virtud de lo antes mencionado, firman:

Forero Vargas, Manuel Guillermo	DNI: AV702661	
Bazan Carhuatanta, Angel Junior	DNI: 74747774	
Pérez Arica, Robert Frank	DNI: 72496295	

Pimentel, 08 de septiembre de 2023

FICHA DE REVISIÓN				
N°	Sección del Informe observado *(Seleccione una opción)*	Número de página observado	Comentario de la observación	Estado de la Observación (Seleccione una opción)
1.	DataSet		Buscar un dataset más adecuado por el poder computacional que tenemos	Aprobado
2.	Entrenamiento de los algoritmos		Buscar ejemplos de Entrenamientos en el repositorio Kaggle	Aprobado
3.				

Universidad Señor de Sipán

ACTA DE REVISIÓN DE ASESORÍA

Yo **Forero Vargas, Manuel Guillermo** quien suscribe como asesor designado mediante Resolución de **Facultad de ingeniería, arquitectura y Urbanismo** N° **0426-2023/FIAU-USS**, del proyecto de investigación titulado **MÉTODO DE CLASIFICACIÓN DE ATAQUES RANSOMWARE UTILIZANDO ALGORITMOS A TRAVÉS DE MACHINE LEARNING**., desarrollado por los estudiantes: **Bazan Carhuatanta, Angel Junior, Pérez Arica, Robert Frank**, del programa de estudios de **Ingeniería de Sistemas**, acreditó haber revisado, realizado observaciones y recomendaciones pertinentes tal como se detalla en el siguiente cuadro:

Fecha de revisión:	Modalidad de Asesoría:	Medio de Asesoría:	Veredicto de Asesoría:
02/10/2023	Virtual	Reunión en Google Meet	Aprobado

En virtud de lo antes mencionado, firman:

Forero Vargas, Manuel Guillermo	DNI: AV702661	
Bazan Carhuatanta, Angel Junior	DNI: 74747774	
Pérez Arica, Robert Frank	DNI: 72496295	

Pimentel, 02 de octubre de 2023

FICHA DE REVISIÓN				
N°	Sección del Informe observado *(Seleccione una opción)*	Número de página observado	Comentario de la observación	Estado de la Observación (Seleccione una opción)
6.	Entrenamiento de los algoritmos		Agregar hiperparametros para mejorar el entrenamiento	Aprobado
7.	Entrenamiento de los algoritmos		Agregar el GridSearch para la búsqueda de hiperparametros	Aprobado
8.	Entrenamiento de los algoritmos		Mejorar la documentación del código de entrenamiento	Aprobado
9.				
10.				

Universidad Señor de Sipán

ACTA DE REVISIÓN DE ASESORÍA

Yo **Forero Vargas, Manuel Guillermo** quien suscribe como asesor designado mediante Resolución de **Facultad de ingeniería, arquitectura y Urbanismo** N° **0426-2023/FIAU-USS**, del proyecto de investigación titulado **MÉTODO DE CLASIFICACIÓN DE ATAQUES RANSOMWARE UTILIZANDO ALGORITMOS A TRAVÉS DE MACHINE LEARNING.**, desarrollado por los estudiantes: **Bazan Carhuatanta, Angel Junior, Pérez Arica, Robert Frank**, del programa de estudios de **Ingeniería de Sistemas**, acreditó haber revisado, realizado observaciones y recomendaciones pertinentes tal como se detalla en el siguiente cuadro:

Fecha de revisión:	Modalidad de Asesoría:	Medio de Asesoría:	Veredicto de Asesoría:
17/10/2023	Virtual	Reunión en Google Meet	Aprobado

En virtud de lo antes mencionado, firman:

Forero Vargas, Manuel Guillermo	DNI: AV702661	
Bazan Carhuatanta, Angel Junior	DNI: 74747774	
Pérez Arica, Robert Frank	DNI: 72496295	

Pimentel, 17 de octubre de 2023

FICHA DE REVISIÓN				
N°	Sección del Informe observado *(Seleccione una opción)*	Número de página observado	Comentario de la observación	Estado de la Observación (Seleccione una opción)
11.	**Modificar Dataset**		Eliminar columnas innecesarias del dataset para un mejor entrenamiento (FileName, md5Hash, BitcoinAddresess)	Aprobado
12.	**Dividir Dataset**		Dividir el dataset en pruebas y entrenamiento (20% de prueba y 80 de entrenamiento)	Aprobado
13.				

Universidad Señor de Sipán

ACTA DE REVISIÓN DE ASESORÍA

Yo **Forero Vargas, Manuel Guillermo** quien suscribe como asesor designado mediante Resolución de **Facultad de ingeniería, arquitectura y Urbanismo** N° **0426-2023/FIAU-USS**, del proyecto de investigación titulado **MÉTODO DE CLASIFICACIÓN DE ATAQUES RANSOMWARE UTILIZANDO ALGORITMOS A TRAVÉS DE MACHINE LEARNING**., desarrollado por los estudiantes: **Bazan Carhuatanta, Angel Junior, Pérez Arica, Robert Frank**, del programa de estudios de **Ingeniería de Sistemas**, acreditó haber revisado, realizado observaciones y recomendaciones pertinentes tal como se detalla en el siguiente cuadro:

Fecha de revisión:	Modalidad de Asesoría:	Medio de Asesoría:	Veredicto de Asesoría:
03/11/2023	Virtual	Reunión en Google Meet	Aprobado

En virtud de lo antes mencionado, firman:

Forero Vargas, Manuel Guillermo	DNI: AV702661	
Bazan Carhuatanta, Angel Junior	DNI: 74747774	
Pérez Arica, Robert Frank	DNI: 72496295	

Pimentel, 03 de noviembre de 2023

FICHA DE REVISIÓN				
N°	Sección del Informe observado *(Seleccione una opción)*	Número de página observado	Comentario de la observación	Estado de la Observación (Seleccione una opción)
16	Modelos		Mejorar el modelo entrenado y guardarlo en un archivo Joblib para posterior implementación en la api	Aprobado
17.	Agregar Matriz de confusión		Agregar una biblioteca para mostrar la matriz de confusión de acuerdo al dataset de pruebas	Aprobado

Universidad Señor de Sipán

ACTA DE REVISIÓN DE ASESORÍA

Yo **Forero Vargas, Manuel Guillermo** quien suscribe como asesor designado mediante Resolución de **Facultad de ingeniería, arquitectura y Urbanismo** N° **0426-2023/FIAU-USS**, del proyecto de investigación titulado **MÉTODO DE CLASIFICACIÓN DE ATAQUES RANSOMWARE UTILIZANDO ALGORITMOS A TRAVÉS DE MACHINE LEARNING**., desarrollado por los estudiantes: **Bazan Carhuatanta, Angel Junior, Pérez Arica, Robert Frank**, del programa de estudios de **Ingeniería de Sistemas**, acreditó haber revisado, realizado observaciones y recomendaciones pertinentes tal como se detalla en el siguiente cuadro:

Fecha de revisión:	Modalidad de Asesoría:	Medio de Asesoría:	Veredicto de Asesoría:
23/11/2023	Virtual	Reunión en Google Meet	Aprobado

En virtud de lo antes mencionado, firman:

Forero Vargas, Manuel Guillermo	DNI: AV702661	
Bazan Carhuatanta, Angel Junior	DNI: 74747774	
Pérez Arica, Robert Frank	DNI: 72496295	

Pimentel, 23 de noviembre de 2023

FICHA DE REVISIÓN				
N°	Sección del Informe observado *(Seleccione una opción)*	Número de página observado	Comentario de la observación	Estado de la Observación (Seleccione una opción)
21.	Arquitecturas de Algoritmos		Agregar bibliotecas para mostrar las arquitecturas de los algoritmos	Aprobado
22.	Revisar resultados previos a la implementación de la API		Resultados óptimos en base al entrenamiento y pruebas	Aprobado
23.	Agregar método voting		Agregar el método voting para realizar predicciones en base a votación de los 3 algoritmos	Aprobado
24.	Informe tesis		Mejorar la redacción del informe	Aprobado

Universidad Señor de Sipán

ACTA DE REVISIÓN DE ASESORÍA

Yo **Forero Vargas, Manuel Guillermo** quien suscribe como asesor designado mediante Resolución de **Facultad de ingeniería, arquitectura y Urbanismo** N° **0426-2023/FIAU-USS**, del proyecto de investigación titulado **MÉTODO DE CLASIFICACIÓN DE ATAQUES RANSOMWARE UTILIZANDO ALGORITMOS A TRAVÉS DE MACHINE LEARNING**., desarrollado por los estudiantes: **Bazan Carhuatanta, Angel Junior, Pérez Arica, Robert Frank**, del programa de estudios de **Ingeniería de Sistemas**, acreditó haber revisado, realizado observaciones y recomendaciones pertinentes tal como se detalla en el siguiente cuadro:

Fecha de revisión:	Modalidad de Asesoría:	Medio de Asesoría:	Veredicto de Asesoría:
11/12/2023	Virtual	Reunión en Google Meet	Aprobado

En virtud de lo antes mencionado, firman:

Forero Vargas, Manuel Guillermo	DNI: AV702661	
Bazan Carhuatanta, Angel Junior	DNI: 74747774	
Pérez Arica, Robert Frank	DNI: 72496295	

Pimentel, 11 de diciembre de 2023

FICHA DE REVISIÓN				
N°	Sección del Informe observado *(Seleccione una opción)*	Número de página observado	Comentario de la observación	Estado de la Observación (Seleccione una opción)
26.	**Revisión de resultados finales**		Los resultados y conclusiones están bien escritos. Los resultados son muy buenos y es un gusto ver que los estudiantes hayan logrado una exactitud superior a la encontrada en las publicaciones previas.	Aprobado
27.				
28.				
29.				
30.				

Anexo 05: Lista de Algoritmos de ML para la Clasificación de Ransomware

N°	Algoritmo
1	Naive Bayes
2	Árboles de decisión (Decision Trees)
3	Random Forest
4	Máquinas de Vectores de Soporte
5	K-Nearest Neighbors
6	Regresión Logística
7	AdaBoost
8	Isolation Forest
9	Mixture Models
10	Redes Neuronales Artificiales
11	Redes Neuronales Convolucionales
12	Redes Neuronales Recurrentes
13	Redes Neuronales Generativas Adversariales
14	Gradient Boosting
15	Máquinas de Aprendizaje Extremo
16	Algoritmos Genéticos
17	Análisis de Componentes Principales
18	Support Vector Data Description
19	Reducción de dimensionalidad
20	Support Vector Clustering
21	Conditional Random Fields
22	Ensemble Methods
23	Reinforcement Learning
24	Deep Belief Networks
25	Extreme Gradient Boosting
26	LightGBM
27	CatBoost
28	Deep Reinforcement Learning
29	Variational Autoencoders
30	Isolation-based Anomaly Detection
31	Local Outlier Factor
32	Kernel Density Estimation
33	Nearest Centroid Classifier
34	Linear Discriminant Analysis
35	Bayesian Networks
36	Stacked Generalization
37	Elastic Net
38	Locally Linear Embedding

39	Long-Short Term Memory Networks
40	Gated Recurrent Units
41	Hierarchical Temporal Memory
42	Hidden Semi-Markov Models
43	Adaptive Boosting
44	Regularized Least Squares
45	Support Vector Regression
46	Gaussian Processes
47	K-Means Clustering
48	Mean Shift
49	Hierarchical Clustering
50	Gaussian Mixture Models

Tabla 14: Lista de Algoritmos de ML para la Clasificación de Ransomware

Anexo 06: Operacionalización de Variables

Variable de estudio independiente	**Definición conceptual**	**Definición operacional**	**Indicadores**	**Instrumentos**	**Valores Finales**	**Tipo de variables**	**Escala de medición**
Algoritmos de Machine Learning	Se utilizan para crear sistemas capaces de aprender y tomar decisiones a partir de datos	Random Forest Decision Tree Support Vector Machine	Consumo de GPU $cp = \sum_{i}^{n} \frac{ce_i}{n}$ Consumo de memoria RAM $CM = \sum_{i}^{n} \frac{cm_i}{n}$ Tiempo de respuesta $t = \sum_{i}^{n} \frac{tf_i - tf_i}{n}$	Ficha electrónica automatizada de observación	Porcentajes	Categórica	Nominal
Variable de estudio dependiente	**Definición conceptual**	**Definición operacional**	**Indicadores**	**Instrumentos**	**Valores Finales**	**Tipo de variables**	**Escala de medición**
Clasificación de Ransomware	Clasificación de Archivos Ransomware a través de sus características definidas por "Ransomware o No Ransomware"	La clasificación del ransomware existen de muchas maneras, en este proyecto de investigación se basa a través de sus características	Precisión Exactitud Error Recall	Ficha electrónica automatizada de observación	Ransomware No Ransomware	Categórica	Nominal

Tabla 15: Operacionalización de Variables