Universidad Cenfotec

Maestría en Tecnología de Bases de Datos

Documento Final

Proyecto de Investigación Aplicada 2.

Elaborado por:

Palacios Rojas Deivid Manuel

Fecha: Julio, 2024

Universidad Cenfotec
Carrera de Postgrado
Maestría en Tecnología de Bases de Datos

TRIBUNAL EXAMINADOR

Este proyecto fue aprobado por el Tribunal Examinador de la carrera: Maestría en Tecnología de Bases de Datos, requisito para optar por el título de grado de Maestría, para el estudiante: Deivid Manuel Palacios Rojas.

_____ _____

M.Sc Jason Ulloa Hernández M.Sc Jorge Arturo Garnier Rovira

_____

M.Sc. Miguel Pérez Montero

Universidad Cenfotec
Carrera de Postgrado
Maestría en Tecnología de Bases de Datos

Detección de enlaces fraudulentos, en bases de datos transaccionales, mediante técnicas de aprendizaje automático.

Detection of fraudulent links in transactional databases using machine learning techniques.

Deivid Manuel Palacios Rojas
dpalaciosr@ucenfotec.ac.cr

Miguel Pérez Montero, M.Sc
mperez@ucenfotec.ac.cr

Universidad Cenfotec, Maestría, Tecnología de Bases de Datos

Resumen

En la era digital, la seguridad informática se ha convertido en un pilar fundamental para proteger los activos y datos valiosos que se van generando todos los días en diferentes sitios. Uno de los principales desafíos que se enfrenta en este tiempo es el crecimiento constante de las amenazas cibernéticas, especialmente en lo que respecta al fraude informático. El fraude cibernético abarca una amplia gama de actividades maliciosas, desde el robo de identidad hasta el *phishing*, pasando por el *ransomware* y ataques de fuerza bruta, entre otros. Sus consecuencias pueden ser devastadoras, tanto para individuos como para organizaciones, en términos de reputación y en el campo económico. La importancia de la seguridad informática radica en la capacidad para prevenir y mitigar estos riesgos. Las soluciones de seguridad, como *firewalls* avanzados, sistemas de detección de intrusiones y autenticación de dos factores, desempeñan un papel crucial en la protección de la integridad y confidencialidad de la información. Además, la concientización y la educación de los usuarios son esenciales para prevenir fraudes en línea. En un mundo donde la mayor parte de la información está al alcance de unos cuantos clics, la seguridad informática

se convierte en un escudo protector que preserva la confianza en la tecnología y en la comunicación en línea, en las distintas plataformas con las que se cuenta actualmente, como lo son celulares, tabletas, laptops, entre otros. Las empresas, los gobiernos y los individuos deben tomar medidas proactivas para salvaguardar los sistemas y las bases de datos, y así contribuir a un entorno digital más seguro y confiable. Este estudio destaca la relevancia de la seguridad informática en la era actual y cómo se maneja la información en las bases de datos, enfocándose en las técnicas de identificación de enlaces fraudulentos que se pueden almacenar en las distintas bases de datos y subrayando la necesidad de tomar medidas preventivas para proteger los activos digitales.

Palabras Clave: Fraude cibernético, Plataforma tecnológica, Bases de datos, Seguridad informática, Inteligencia Artificial, Aprendizaje automático, Enlace web.

Abstract

In the current digital era, cybersecurity has become a fundamental pillar for protecting our assets and valuable data generated every day across different platforms. One of the main challenges we face today is the constant growth of cyber threats, especially concerning computer fraud. Cyber fraud encompasses a wide range of malicious activities, from identity theft to phishing, ransomware, and brute force attacks, among others. Its consequences can be devastating, both for individuals and organizations, in terms of reputation and economic impact. The importance of cybersecurity lies in its ability to prevent and mitigate these risks. Security solutions such as advanced firewalls, intrusion detection systems, and two-factor authentication play a crucial role in safeguarding the integrity and confidentiality of information. Furthermore, user awareness and education are essential to prevent online fraud. In a world where most information is just a few clicks away, cybersecurity becomes a protective shield that preserves trust in technology and online communication across various platforms such as mobile phones, tablets, laptops, etc. Companies, governments, and individuals must take proactive measures to safeguard systems and databases, contributing to a safer and more reliable digital environment. This study highlights the relevance of cybersecurity in the current era and how information is managed in databases, focusing on techniques for identifying fraudulent links stored in various databases and emphasizing the need for preventive measures to protect digital assets.

Keywords: Cyber fraud, Technological platform, Databases, Cybersecurity, Artificial Intelligence, Machine Learning, Web link.

1. Introducción

Los enlaces fraudulentos, en bases de datos, son un problema creciente en la era digital, y continua en crecimiento. Este fenómeno se ha convertido en una preocupación seria para la integridad de la información en línea y la que se almacena en los repositorios de datos cada día. Los antecedentes de este problema se remontan a la proliferación de sitios web y plataformas en línea que permiten la inclusión de enlaces a contenido externo, ya sea de manera intencionada o no.

El surgimiento de enlaces fraudulentos se atribuye a diversas razones, entre las cuales se encuentra el deseo de aumentar el tráfico hacia un sitio web, mejorar su clasificación en los motores de búsqueda o, en casos más oscuros, difundir malware y realizar estafas en línea para apoderarse de datos de manera ilegal. Estos enlaces pueden camuflarse como recursos legítimos, lo que hace que sean difíciles de detectar para los usuarios y los administradores de bases de datos. El impacto de los enlaces fraudulentos es perjudicial, ya que destruye la confiabilidad de la información y la seguridad en línea para todos los usuarios. Además, puede afectar negativamente la reputación de los sitios web y las empresas afectadas, provocando pérdidas millonarias y, lastimosamente, de imagen ante el mundo.

Para abordar este problema, es esencial contar con mecanismos efectivos de detección y prevención de enlaces fraudulentos en bases de datos, así como educar a los usuarios sobre cómo identificar y evitar estos enlaces. La ciberseguridad y la lucha contra el fraude en línea son desafíos continuos en el mundo de la tecnología, y la evolución de las estrategias de enlaces fraudulentos plantea una necesidad constante de adaptación y vigilancia en la comunidad en línea. Es por lo que, mediante esta investigación, se intenta construir un mecanismo de identificación de enlaces mediante técnicas de inteligencia artificial (IA) que ayuden a los administradores de bases de datos a aportar más seguridad a sus repositorios de datos.

Es así como contar con mecanismos de defensa es crucial para la detección de enlaces fraudulentos en bases de datos. Estos mecanismos garantizan la integridad y la confiabilidad de los datos, protegiendo contra ataques maliciosos y preservando la precisión de los resultados.

Los enlaces fraudulentos pueden conducir a decisiones erróneas o a la manipulación de sistemas, lo que puede tener consecuencias graves en diversas áreas, como la seguridad financiera, la salud pública o la seguridad nacional. Los mecanismos de defensa pueden incluir técnicas de detección de anomalías, modelos de aprendizaje supervisado y no supervisado, así como la implementación de algoritmos de seguridad más avanzados. Estos sistemas están diseñados para identificar patrones sospechosos o comportamientos anómalos en los datos, lo que permite a los analistas y responsables de la toma de decisiones intervenir antes de que se produzcan daños.

Además de detectar enlaces fraudulentos, los mecanismos de defensa en *machine learning* también pueden mejorar la resiliencia del sistema frente a otros ataques, como la manipulación de datos. Esto ayuda a mantener la integridad de las bases de datos y a proteger la confianza en los sistemas de modelos de aprendizaje, lo que es fundamental en un mundo cada vez más dependiente de la automatización y el análisis de datos. Es claro, los mecanismos de defensa son fundamentales para garantizar la seguridad y la fiabilidad en el procesamiento de la información.

2. Métodos

Esta investigación es de carácter evaluativa y tiene como objetivo la revisión de los resultados obtenidos al final del proceso de identificación de enlaces fraudulentos en una base de datos, aportando recomendaciones desde el punto de vista profesional del grado de un Máster en Tecnología de Bases de Datos, con el apoyo de profesionales en Ciberseguridad, así como la correcta interpretación para una adecuada gestión cuando se revisan los resultados. Se debe tener claro que, como profesionales en las áreas de Tecnología de Bases de Datos y Ciberseguridad, se tiene que asegurar que los resultados que se obtienen, al final de un determinado proceso, cumplan con los requisitos establecidos para su debido análisis, dando como resultado un aumento en la seguridad en el ambiente destinado para las bases de datos.

Los administradores de bases de datos se enfrentan a la necesidad de proteger la información y privacidad datos de la empresa, así como a la responsabilidad de verificar la autenticidad de los recursos en línea, a los cuales están accediendo los usuarios en sus bases de datos. Esto plantea desafíos para las empresas que almacenan y gestionan información diariamente, ya que deben implementar medidas de seguridad para prevenir la inclusión de enlaces fraudulentos en sus repositorios. Además, se requiere una mayor conciencia y educación digital para que los usuarios

puedan identificar y evitar estos enlaces. En última instancia, la lucha contra los enlaces web fraudulentos es esencial para mantener un entorno en línea libre de potenciales escenarios de pérdida de información. Mantenerse actualizados en el desarrollo de algoritmos de *machine learning* para la seguridad de los datos es esencial, primariamente porque la evolución constante de las amenazas cibernéticas requiere de soluciones cada vez más sofisticadas para proteger la integridad y confidencialidad de la información. Al estar al día con los avances en algoritmos de *machine learning*, podemos desarrollar sistemas de detección y prevención de ataques más efectivos. Por lo tanto, el desarrollo de algoritmos de este tipo, para la seguridad de los datos, no solo ayuda a estar un paso adelante de los ciberatacantes, sino que también permite proteger de manera más eficaz la información sensible[1] y preservar la confianza de los usuarios y clientes.

3. Escenario

Cuando se generan grandes cantidades de datos es vital empezar a pensar en que pueden existir usuarios no autorizados que quieran hacerse con estos repositorios de información, los fines pueden ser altamente dañitos para las organizaciones y sus clientes, por lo que cada empresa tendrá un panorama diferente y confidencial en cuanto a lo que quieran proteger, sin embargo, el objetivo es el mismo, poder cuidar sus datos para mantenerse a flote en este mundo cada vez más competitivo y amenazado por criminales informáticos.

Afortunadamente, el uso de técnicas de aprendizaje automático puede aportar valor dentro de las empresas como una estrategia clave en la defensa contra amenazas cibernéticas. Estas técnicas y sus algoritmos ofrecen analizar el comportamiento e identificar patrones que sugieran actividad maliciosa. Uno de los escenarios a los cuales hoy en día los usuarios y las empresas se ven expuestos, es la información que viaja a través de la web y a su vez, se queda almacenada en las distintas bases de datos que soportan estos procesos, cuando se trata de seguridad cibernética, los enlaces fraudulentos pueden representar una puerta de entrada para ataques como phishing, malware o robo de datos confidenciales. Estos enlaces suelen disfrazarse hábilmente para parecer legítimos, lo que dificulta su detección mediante métodos convencionales. Actualmente, las distintas técnicas de aprendizaje automático ofrecen soluciones dinámicas y adaptables los

[1] Los datos sensibles son información confidencial o privada que, si es divulgada, podría causar daño, pérdida o perjuicio a una persona, organización o entidad.

desafíos. Al entrenar modelos con grandes conjuntos de datos que contienen tanto enlaces legítimos como fraudulentos, los sistemas pueden aprender a distinguir entre ellos mediante la identificación de características comunes asociadas con el comportamiento malicioso. Estas características pueden incluir la estructura del enlace, el contenido de la página de destino, el contexto en el que se comparte el enlace y el comportamiento del usuario al interactuar con él. Una vez que un modelo está entrenado, puede aplicarse en tiempo real para analizar y clasificar enlaces entrantes, alertando a los usuarios o bloqueando los enlaces identificados como fraudulentos antes de que causen daño, que es exactamente el objetivo de esta investigación. Además, el aprendizaje automático permite la adaptación continua a medida que evolucionan las tácticas de los atacantes, lo que garantiza una defensa fuerte y actualizada contra las amenazas que no detienen su crecimiento. Mediante esta capa de defensa adicional en este tipo de escenarios es que se puede controlar el acceso a la información almacenada, cuidar al mismo tiempo la reputación de las organizaciones y sus clientes.

3.1 Interrogantes

La detección de enlaces fraudulentos en bases de datos ha emergido como un desafío crítico en el contexto de la seguridad de la información desde hace algún tiempo. Con el crecimiento exponencial y acelerado de la interconexión de datos, es más que necesario abordar la amenaza de enlaces falsos que podrían comprometer la integridad y confiabilidad de la información almacenada, algunas preguntas que surgen al intentar determinar con exactitud el problema son: ¿Cuáles son los desafíos específicos en la detección de enlaces fraudulentos? La identificación precisa de enlaces falsos enfrenta obstáculos como la variabilidad en los patrones de fraude y la evasión de métodos convencionales. ¿Cómo abordan las técnicas de aprendizaje automático estos desafíos? Los algoritmos de aprendizaje automático, como redes neuronales y árboles de decisión, ofrecen la capacidad de aprender patrones complejos y adaptarse a cambios en el comportamiento del fraude. ¿Cuál es el estado actual de la investigación en la detección de enlaces fraudulentos? Investigaciones recientes han destacado el uso de modelos de aprendizaje profundo y técnicas de procesamiento de lenguaje natural para mejorar la precisión en la identificación de enlaces fraudulentos. ¿Cómo se manejan las limitaciones en la disponibilidad de datos etiquetados? El etiquetado de datos puede ser costoso y laborioso; por lo tanto, se están explorando enfoques de aprendizaje semi-supervisado y no supervisado para superar estas limitaciones.

3.2 Implementación de algoritmos de ML en bases de datos

En este apartado, se detallan los pasos llevados a cabo para el desarrollo de un sistema de detección de enlaces fraudulentos en una base de datos, el *dataset* utilizado cuenta con más de 600.000 registros que fueron precisamente seleccionados para el desarrollo de esta investigación. La implementación abarca desde la preparación de los datos, limpieza, detección de patrones anómalos, hasta el entrenamiento de una porción de la información y su debida clasificación mediante el uso de cuatro diferentes algoritmos que se mencionan a continuación:

- Decision Tree Classifier
- Random Forest Classifier
- Extra Trees Classifier
- GaussianN

Es importante resaltar que, para efectos de esta investigación, se utilizan 4 distintos algoritmos previamente seleccionados según recomendaciones obtenidas de fuentes especializadas, ya que también se busca obtener respuesta sobre cuál sería el más eficiente, o si todos pueden servir como herramienta, al momento de trabajar con grandes volúmenes de datos, en términos de rendimiento.

3.3 Preparación del *dataset*

Para una mejor obtención de resultados, se inicia "limpiando" cada registro dentro del *dataset* mediante técnicas de programación en Python, algunos valores dentro del conjunto de datos pueden impactar de manera negativa al momento de que los algoritmos se encuentren en ejecución, es por ello que se decide evitar este comportamiento no deseado mediante:

- Eliminación de palabras repetidas que no aporten valor al resultado
- Eliminación de datos de carácter nulo
- Reorganización de columnas de una manera lógica
- Eliminación de espacios dentro de los registros que puedan alterar el resultado
- Modificación de valores de tipo de dato texto a entero para optimización.

3.4 Análisis e identificación de características (*features*)

Encontrar características o patrones en un conjunto de datos es crucial para aplicar *machine learning* ya que permite comprender la estructura subyacente de los datos y extraer información importante. Estas características sirven como entradas para los algoritmos de aprendizaje

automático, permitiéndoles generalizar y hacer predicciones precisas sobre nuevos datos. Identificar patrones facilita la clasificación, la predicción y la toma de decisiones informada. Además, al entender las características significativas, se pueden optimizar los modelos, mejorar su rendimiento y evitar el sobreajuste. En resumen, la identificación de características es fundamental para el éxito y la eficacia del aprendizaje automático. Para la presente investigación se ejecutan las siguientes técnicas de análisis de los datos:

- Limpieza del campo URL: Mediante el uso de una expresión regular en Python una de las columnas llamada "url" dentro del *dataset* es la que aporta los registros principales para esta investigación, debido a la existencia de muchos campos con el mismo dato repetido "www." Esa porción de la información no aporta valor relevante así que se remplaza por una cadena vacía.
- Asignación de etiquetas para el modelo supervisado: Siguiendo las técnicas recomendadas para limpieza y preparación de los datos de entrenamiento en los algoritmos de *machine learning*, los datos (etiquetas) que serán pasados al modelo son modificados por valores numéricos debido a las siguientes ventajas:

  1. Compatibilidad con algoritmos: Muchos algoritmos de *machine learning* están diseñados para trabajar con datos numéricos. Utilizar etiquetas numéricas facilita la implementación de estos algoritmos.
  2. Eficiencia computacional: Los cálculos numéricos suelen ser más eficientes que el procesamiento de texto, lo que puede mejorar el rendimiento de los algoritmos de *machine learning*, especialmente en conjuntos de datos grandes.
  3. Generalización: Los modelos de *machine learning* pueden generalizar mejor los patrones en los datos numéricos. Los datos numéricos proporcionan una representación más compacta y generalizable de la información, lo que puede resultar en modelos más robustos y con mejor capacidad de generalización, especialmente cuanto se trata de clasificación.
  4. Facilidad de interpretación: Los resultados de los modelos de *machine learning* suelen ser más fáciles de interpretar cuando se utilizan datos numéricos. Las relaciones entre las características son más claras cuando se expresan en términos numéricos, lo que facilita la comprensión de cómo el modelo está tomando decisiones.

- Extracción y conteo de caracteres especiales del *dataset*: ['@','?','-','=','.','#','%','+','$','!','*',',','//']: Con el fin de aumentar la efectividad también se toma en cuenta la cantidad de caracteres especiales que puede contener un registro, y se procede con su debido conteo en este paso del análisis.
- Extracción del dominio primario del campo: Mediante una función en Python se cumple con el objetivo de solamente extraer el dominio principal del enlace web, este valor permite asociar el dato a un enlace que efectivamente existe.
- Detección de la cantidad de números dentro de un registro: Se considera de importancia identificar toda la información relevante dentro del set de datos, es por ello por lo que también se hace análisis y conteo de la cantidad de números que se encuentran presentes en cada registro.
- Detección de la cantidad de letras dentro de un registro: Se considera de importancia identificar toda la información relevante dentro del set de datos, es por ello por lo que también se hace análisis y conteo de la cantidad de letras que se encuentran presentes en cada registro.

3.5 Datos de entrenamiento

A continuación, se describe cómo se inicia a dar forma al proceso de entrenar y seleccionar los datos que serán procesados por los algoritmos de *machine learning* partiendo de la siguiente imagen:

```
#Entrenamiento y division de los datos a evaluar en los modelos de ML
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=2)
```

Imagen 1 Instrucción para seleccionar datos para entrenamiento

Como se puede apreciar, el código realiza una división de un conjunto de datos en *conjuntos de entrenamiento y prueba* utilizando la función *train_test_split* de la biblioteca scikit-learn (sklearn). *X*, es el conjunto de características (o variables independientes) del conjunto de datos.
y, es el conjunto de etiquetas (o variable dependiente) asociadas a las características en *X*.
La función *train_test_split* toma como entrada *X* y *y*, y divide los datos en cuatro partes:

- *X_train*: Conjunto de características para entrenamiento.
- *X_test*: Conjunto de características para pruebas.

- *y_train*: Etiquetas correspondientes a X_train.
- *y_test*: Etiquetas correspondientes a X_test.

Los parámetros que se transfieren a *train_test_split* son:

test_size=0.2: Indica el tamaño que se desea para el conjunto de prueba. En este caso, se está especificando que el 20% de los datos se reservarán para pruebas, mientras que el 80% restante se usará para entrenamiento.

random_state=2: Esto asigna una semilla para el generador de números aleatorios, lo que asegura que la división de los datos sea reproducible. Es decir, si se ejecuta el mismo código con la misma semilla varias veces, se obtendrá la misma división de datos en cada ejecución. Esto es útil para propósitos de reproducibilidad.

```
#Se inicializan los modelos y se inicia el entrenamiento
models_list = [DecisionTreeClassifier,RandomForestClassifier,ExtraTreesClassifier,GaussianNB]
```

Imagen 2 Inicio del entrenamiento

Seguidamente, se determinan los algoritmos de *machine learning* que se utilizaron durante el desarrollo de esta investigación. Resaltando un punto importante: se están utilizando cuatro algoritmos porque también se quiere medir la eficiencia de los resultados, es decir, si todos son suficientemente capaces de producir resultados o si solamente uno podrá terminar todas las tareas.

Existen más algoritmos que se pueden implementar al momento de resolver una situación de este tipo, sin embargo, para efectos de esta investigación se utilizaron los siguientes:

- Decision Tree Classifier: Este recibirá datos preetiquetados después clasificarlos y posteriormente con estos mismos datos llevar a cabo una predicción. Este algoritmo forma parte de los modelos de aprendizaje supervisados.
- Random Forest Classifier: Este algoritmo cuenta con buena reputación al momento de predecir, es conocido por obtener buenas coincidencias y es posible utilizarlo en tareas de regresión y clasificación.
- Extra Trees Classifier: Como su nombre lo dice Árboles "extra" crea numerosos árboles de decisión, pero el muestreo de cada árbol es aleatorio, sin sustitución. Con ello se crea un dataset para cada árbol con muestras única.

- Gaussian NB: Este algoritmo se caracteriza ya que se basa en la aplicación una distribución normal gaussiana admitiendo datos continuos, de igual manera como se menciona para los algoritmos anteriores será de gran ayuda en temas de clasificación.

3.6 Distribución de los datos de entrenamiento

Una vez entendido el proceso inicial, se continua el trabajo que requiere la manipulación y puesta en marcha de los algoritmos. En principio se cuenta con la siguiente distribución de los datos:

Type	Records
Benign	428103
Defacement	96457
Phishing	94111
Malware	32520
Total	**651191**

Es importante aclarar que este *dataset* puede ser modificado en caso de ser necesario, para efectos de esta investigación se estuvieron agregando, modificando y eliminando datos para prueba y validación de los modelos, sin embargo, no hay limitante ya que puede contener tantos datos como sea necesario. La idea de trabajar con esta cantidad de información es simular la extracción desde un motor de bases de datos en una organización y seguidamente iniciar las tareas de limpieza y procesamiento hasta tener un set de datos limpio y preparado.

Como se puede apreciar en la tabla anterior la muestra cuenta con 4 diferentes grupos de datos los cuales no poseen ningún trabajo previo en aspectos de Inteligencia Artificial, lo que se busca llevar a cabo es mediante *machine learning* como se explicó anteriormente, entrenar una serie de modelos de aprendizaje reconociendo parámetros para que estos clasifiquen los enlaces dependiendo de sus características. Un aspecto para detallar es que para poder iniciar con todo el proceso de entrenamiento los datos deben ser previamente etiquetados debido a que el tipo de aprendizaje aplicado es supervisado en todo el alcance de esta investigación, una vez que se completa el correcto etiquetado el conjunto de datos se visualiza de la siguiente forma:

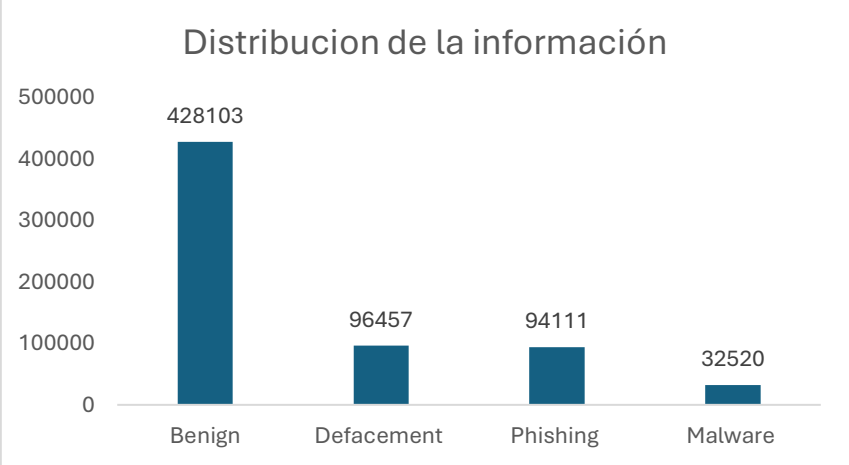
Type	Records	Category
Benign	428103	0
Defacement	96457	1
Phishing	94111	2
Malware	32520	3
Total	**651191**	

```
add = {"Category": {"benign": 0, "defacement": 1, "phishing":2, "malware":3}}
```

Imagen 3 Etiquetado del conjunto de datos para procesamiento

3.7 Ejecución de los algoritmos de *machine learning* y resultados

En este paso luego de preparar la información de manera correcta y asignar la porción de los datos para entrenamiento y prueba los algoritmos se encuentran listos para ser ejecutados, lo primero que el programa en Python realiza es una visualización de los datos un poco más detallada:



En el siguiente paso, el programa empieza a aplicar los algoritmos seleccionados, se puede notar como la primera vez que se pone en marcha, toda la ejecución tarda en procesar toda la información un tiempo considerable (el proceso no se está llevando en máquina virtual), a continuación, los resultados de cada uno, luego de iterar varias veces, para encontrar el mayor porcentaje de acierto:

DecisionTreeClassifier: Test Accuracy : 90.88%

	Precisión	Recall	f1-score	support
0	0.92	0.97	0.94	85565
1	0.93	0.96	0.94	19319
2	0.80	0.56	0.66	18805
3	0.94	0.91	0.92	6550

Accuracy			0.91	130239
macro avg	0.90	0.85	0.87	130239
weighted avg	0.90	0.91	0.90	130239

Para este primer modelo, se logran obtener datos muy interesantes, aquí se explican algunos de ellos, empezando por revisar la variable *Test Accuracy* (Precisión del Test) la cual indica que la proporción de predicciones correctas, sobre el total de predicciones realizadas por el modelo, en el conjunto de datos de prueba, tiene una precisión del 90.88%, lo que indica que clasifica correctamente el 90.88% de los ejemplos en el conjunto de datos de prueba. Con respecto a los demás datos, *Precisión* (Precisión) se nota que, por ejemplo, para la clase 0, el modelo clasifica un 92% de los datos de manera correcta, 93% para la clase 1 y así hasta completar las clases. Por otra parte, para *Recall* (Recuperación o Sensibilidad), para la clase 0 el recall es del 97%, lo que significa que el 97% de todos los ejemplos de la clase 0 fueron clasificados correctamente por el modelo. Por último, *Accuracy* (Precisión Global), el modelo clasifica correctamente el 91% de todos los ejemplos en el conjunto de datos de prueba. En resumen, para este primer modelo, validando las métricas proporcionadas, el modelo parece ser bastante efectivo al final del procesamiento.

RandomForestClassifier: Test Accuracy : 91.43%

	precision	recall	f1-score	support
0	0.92	0.98	0.95	85565
1	0.94	0.96	0.95	19319
2	0.83	0.57	0.68	18805
3	0.96	0.91	0.93	6550
accuracy			0.91	130239
macro avg		0.91	0.86	0.88	130239
weighted avg		0.91	0.91	0.91	130239

Para este segundo modelo se obtienen los resultados, aquí se explican algunos de ellos, como en el caso anterior, empezando por revisar la variable *Test Accuracy* (Precisión del Test), la cual indica que la proporción de predicciones correctas sobre el total de predicciones realizadas por el modelo, en el conjunto de datos de prueba, tiene una precisión del 91.43%, lo que indica que clasifica correctamente el 91.43% de los ejemplos en el conjunto de datos de prueba. Con respecto a los demás datos, *Precisión* (Precisión), se nota que, por ejemplo, para la clase 1, el modelo clasifica un 94% de los datos de manera correcta, 83% para la clase 2 y así hasta completar las clases. Por otra parte, para *Recall* (Recuperación o Sensibilidad), para la clase 0 el

recall es del 98%, lo que significa que el 98% de todos los ejemplos de la clase 0 fueron clasificados correctamente por el modelo. Por último, *Accuracy* (Precisión Global), el modelo clasifica correctamente el 91% de todos los ejemplos en el conjunto de datos de prueba. En resumen, para este segundo modelo, validando las métricas proporcionadas, el modelo también parece ser muy efectivo al final del procesamiento.

ExtraTreesClassifier: Test Accuracy : 91.45%

	precision	recall	f1-score	support
0	0.92	0.98	0.95	85565
1	0.93	0.97	0.95	19319
2	0.84	0.57	0.68	18805
3	0.96	0.91	0.94	6550

Accuracy				0.91	130239
macro avg		0.91	0.86	0.88	130239
weighted avg		0.91	0.91	0.91	130239

Para este tercer modelo se obtienen los resultados después de varias iteraciones, aquí se explican algunos de ellos, como en el caso anterior, empezando por revisar la variable *Test Accuracy* (Precisión del Test), la cual indica que la proporción de predicciones correctas sobre el total de predicciones realizadas por el modelo en el conjunto de datos de prueba. tiene una precisión del 91.45%, lo que indica que clasifica correctamente el 91.45% de los ejemplos en el conjunto de datos de prueba. Con respecto a los demás datos, *Precisión* (Precisión). se nota que, por ejemplo. para la clase 3. el modelo clasifica un 96% de los datos de manera correcta, 84% para la clase 2 y así hasta completar las clases. Por otra parte. para *Recall* (Recuperación o Sensibilidad). para la clase 1 el recall es del 97%, lo que significa que el 97% de todos los ejemplos de la clase 1 fueron clasificados correctamente por el modelo. Por último. *Accuracy* (Precisión Global). el modelo clasifica correctamente el 91% de todos los ejemplos en el conjunto de datos de prueba. En resumen. para este tercer modelo validando las métricas proporcionadas, el modelo también parece ser muy efectivo al final del procesamiento.

Gaussian NB: Test Accuracy : 77.05%

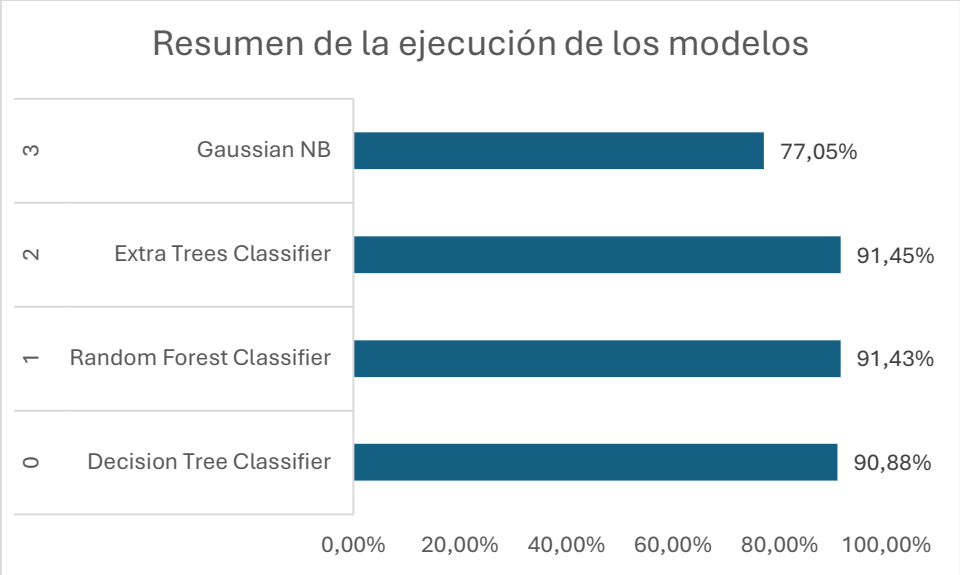
	precision	recall	f1-score	support
0	0.85	0.92	0.88	85565
1	0.61	1.00	0.76	19319
2	0.57	0.02	0.03	18805
3	0.42	0.34	0.38	6550

accuracy				0.77	130239
macro avg		0.61	0.57	0.51	130239
weighted avg		0.75	0.77	0.71	130239

Para este cuarto modelo se obtienen los resultados también, aquí se explican algunos de ellos. como en el caso anterior. empezando por revisar la variable *Test Accuracy* (Precisión del Test), la cual indica que la proporción de predicciones correctas sobre el total de predicciones realizadas por el modelo. en el conjunto de datos de prueba. tiene una precisión del 77.05%, lo que indica que clasifica correctamente el 77.05% de los ejemplos en el conjunto de datos de prueba. Con respecto a los demás datos, *Precisión* (Precisión). se nota que, por ejemplo. para la clase 3 el modelo clasifica apenas un 42% de los datos de manera correcta, 57% para la clase 2 y así hasta completar las clases. Por otra parte. para *Recall* (Recuperación o Sensibilidad). para la clase 3 el recall es del 34%, lo que significa que el 34% de todos los ejemplos de la clase 3 fueron clasificados correctamente por el modelo. Por último. *Accuracy* (Precisión Global). el modelo clasifica correctamente solo un 77% de todos los ejemplos en el conjunto de datos de prueba. En resumen. para este cuarto modelo, validando las métricas proporcionadas, el modelo no cuenta con tanta efectividad al momento de terminar el procesamiento.

Finalmente, como último paso, se obtiene la tabla general del resumen de los modelos, una vez completadas todas las iteraciones:

No	Model	Accuracy
0	Decision Tree Classifier	0.908845
1	Random Forest Classifier	0.914311
2	Extra Trees Classifier	0.914473
3	Gaussian NB	0.770545



4. Conclusiones

En primer lugar, luego de haber completado esta investigación de principio a fin, se puede entender que los algoritmos de *machine learning* son capaces de identificar anomalías y comportamientos sospechosos que podrían pasar desapercibidos para algunas aplicaciones convencionales, sin embargo, la preparación de los datos juega un papel importante en el éxito de las predicciones, por ello, la recomendación es invertir el adecuado tiempo en la correcta manipulación de los datos y su análisis antes de ser procesados por los algoritmos. Estos algoritmos, además, podrían (según sea el caso) aprender de datos históricos y adaptarse a nuevas amenazas, lo que los hace altamente efectivos en la detección de peligros en tiempo real. Además, el uso de algoritmos de *machine learning* puede reducir significativamente el tiempo y los recursos necesarios para analizar grandes cantidades de datos. Al automatizar el proceso de detección, las organizaciones pueden identificar y eliminar rápidamente las amenazas, reduciendo el impacto en la seguridad de la información y en la reputación de la empresa, que es tan vital para toda organización. Otra ventaja es la capacidad de los algoritmos de mejorar con el tiempo. A medida que se alimentan con más datos y se mejoran con retroalimentación, estos algoritmos pueden volverse más precisos y sofisticados en la detección de enlaces o alguna anomalía en específico, manteniendo así actualizada la seguridad de las bases de datos y adaptada a las últimas amenazas. Finalizando, en resumen, el uso de algoritmos de *machine learning* en bases de datos, para detectar enlaces fraudulentos, es adecuado y beneficioso debido a su capacidad para identificar patrones sutiles, su eficiencia en el análisis de grandes volúmenes de datos, así como su capacidad para

mejorar con el tiempo. Estos algoritmos ofrecen una herramienta considerable en la lucha contra el fraude digital a bases de datos, proporcionando a las empresas una mayor seguridad y tranquilidad en un entorno cada vez más digitalizado y cambiante.

5. Bibliografía

1. Rojas, L. G. (2023, February 28). Arboles de decisión (Decision Tree Classifier) - Desafío de programación: Python para ciencia de datos [Video]. LinkedIn. https://es.linkedin.com/learning/desafio-de-programacion-python-para-ciencia-de-datos/arboles-de-decision-decision-tree-classifier#:~:text=(M%C3%BAsica%20de%20videojuego)%20Los%20clasificadores,usar%20para%20hacer%20una%20predicci%C3%B3n.
2. Team, D. (2023, October 30). Random Forest: Bosque aleatorio. Definición y funcionamiento. Formation Data Science | DataScientest.com. https://datascientest.com/es/random-forest-bosque-aleatorio-definicion-y-funcionamiento
3. Cómo funciona el algoritmo de clasificación y regresión Árboles extra—ArcGIS Pro | Documentación. (n.d.). https://pro.arcgis.com/es/pro-app/latest/tool-reference/geoai/how-extra-tree-classification-and-regression-works.htm#:~:text=%C3%81rboles%20extra%20(%22extra%22%20proviene,la%20herramienta%20Entrenar%20con%20AutoML.
4. Majumder, P. (2020, February 23). Gaussian naive bayes. OpenGenus IQ: Computing Expertise & Legacy. https://iq.opengenus.org/gaussian-naive-bayes/
5. ¿Qué es una base de datos? (n.d.). https://www.oracle.com/mx/database/what-is-database/
6. Blanco, A. G. (2023, June 23). ¿Qué son los enlaces maliciosos y cómo protegerse ante esta amenaza? BBVA NOTICIAS. https://www.bbva.com/es/innovacion/que-son-los-enlaces-maliciosos-y-como-protegerse-ante-esta-amenaza/

7. Mimecast. (n.d.). What is Cyber Fraud? Mimecast. https://www.mimecast.com/content/cyber-fraud/
8. What is Cyberfraud | IGI Global. (n.d.). https://www.igi-global.com/dictionary/turning-westward-information-policies-post/6602
9. Craigen, D. (2014). Defining cybersecurity. TIM Review. https://www.timreview.ca/article/835
10. ¿Qué es la ciberseguridad? - Explicación de la ciberseguridad - AWS. (n.d.-b). Amazon Web Services, Inc. https://aws.amazon.com/es/what-is/cybersecurity/
11. González, O. (2023, October 4). Machine Learning: ¿Qué es y por qué es tan importante? | Tecon. Tecon. https://www.tecon.es/machine-learning-que-es-y-por-que-es-tan-importante/#:~:text=La%20importancia%20del%20Machine%20Learning%20y%20sus%20beneficios&text=Como%20hemos%20comentado%20anteriormente%2C%20gracias,en%20una%20escala%20muy%20grande.
12. IBM documentation. (n.d.). https://www.ibm.com/docs/en/db2/11.5?topic=content-in-database-machine-learning