1

**UNIVERSIDAD AUTÓNOMA DE MADRID**

**ESCUELA POLITÉCNICA SUPERIOR**

**Gráfico de barras

Descripción generada automáticamente con confianza media**

**Grado en Ingeniería Informática**

**TRABAJO FIN DE GRADO**

**Análisis y Estrategias para la Mitigación del Phishing**

**Un Enfoque en la Ciberseguridad**

**Autor: Adrián Crespo Musheghyán**

**Tutor: Óscar Delgado**

**Julio 2024**

### Agradecimientos

### Resumen

Este Trabajo presenta el diseño, desarrollo e implementación de una aplicación destinada a detectar y notificar URLs sospechosas. La aplicación permite introducir una URL y buscar coincidencias con URLs creadas recentemente, enviando un informe por correo electrónico con las URLs potencialmente maliciosas. Además, proporciona una visualización gráfica y una clasificación de las URLs según su tipo de similitud.

El documento se estructura en varios capítulos. En el primer capítulo, se contextualiza el problema del phishing y se presentan las principales amenazas actuales en el ámbito de la ciberseguridad. Se destacan las técnicas más utilizadas por los ciberdelincuentes y se subraya la importancia de herramientas efectivas para combatir estas amenazas.

El segundo capítulo se centra en el estado del arte, comparando nuestra aplicación con otros productos y proyectos existentes, tanto comerciales como académicos. Se describen brevemente las características principales de las soluciones más relevantes y se realiza una comparación detallada, destacando las similitudes y diferencias en cuanto a funcionalidad, precisión y usabilidad.

En el tercer capítulo, se describe el diseño de la aplicación. Se presentan los requisitos funcionales y no funcionales, así como los diagramas de secuencia y otros elementos de ingeniería de software necesarios para entender la arquitectura y el flujo de la aplicación. Este capítulo también aborda las tecnologías utilizadas y los desafíos enfrentados durante el desarrollo.

Los resultados obtenidos se discuten en el cuarto capítulo, donde se analizan los datos recopilados y se evalúa el desempeño de la aplicación en escenarios de prueba reales. Se presentan estadísticas y gráficos que ilustran la efectividad de la herramienta en la detección de URLs sospechosas y se comparan los resultados con las expectativas iniciales.

Finalmente, en el quinto capítulo, se presentan las conclusiones del trabajo. Se resumen los principales hallazgos, se discuten las implicaciones de los resultados y se sugieren posibles mejoras y líneas futuras de investigación. Se concluye que la aplicación desarrollada es una herramienta útil y efectiva para la detección de phishing, contribuyendo a la mejora de la seguridad en la navegación web.

**Palabras clave**

 **Phishing**:

Técnica de fraude online en la que los ciberdelincuentes intentan obtener información sensible, como contraseñas, datos bancarios, etc, haciéndose pasar por entidades confiables en comunicaciones digitales.

 **Ingeniería Social**:

Método utilizado por los atacantes para manipular a las personas a fin de obtener información confidencial, generalmente a través de interacciones humanas y engaños.

 **Correo Electrónico de Phishing**:

Mensajes fraudulentos que parecen provenir de fuentes legítimas, como bancos o empresas, con el objetivo de engañar a los destinatarios para que revelen información personal o descarguen malware.

 **Malware**:

Software malicioso diseñado para dañar, interrumpir o tomar el control de sistemas informáticos. Se puede distribuir a través de correos electrónicos de phishing y otros métodos.

 **Ransomware**:

Tipo de malware que cifra los archivos de una víctima y exige un rescate para restaurar el acceso. A menudo se entrega a través de campañas de phishing.

 **Spyware**:

Software que recopila información sobre una persona o una organización sin su conocimiento y lo envía a otra entidad sin el consentimiento del usuario.

 **Scareware**:

Táctica de ingeniería social que utiliza alertas falsas para asustar a los usuarios y hacerles creer que su computadora está infectada, incitándolos a descargar software malicioso.

 **Macros Maliciosas**:

Scripts embebidos en documentos, generalmente de Microsoft Office, que pueden ejecutar código malicioso cuando se abren, comúnmente distribuidos a través de correos electrónicos de phishing.

 **Suplantación de Identidad (Spoofing)**:

Acto de engañar a un sistema o a una persona haciéndoles creer que están interactuando con una entidad confiable. En phishing, esto se logra mediante la falsificación de direcciones de correo electrónico y URLs.

 **Ciberdelincuencia**:

* Actividades delictivas que involucran computadoras y redes informáticas, incluyendo el uso de phishing para cometer fraudes y otros delitos

### Índice

[Agradecimientos III](#_Toc169211831)

[Resumen 5](#_Toc169211832)

[Índice IX](#_Toc169211833)

[Índice de figuras XI](#_Toc169211834)

[Índice de tablas XI](#_Toc169211835)

[Índice de cuadros XI](#_Toc169211836)

[1 Introducción 1](#_Toc169211837)

[Tipos de Phishing 2](#_Toc169211838)

[Estrategias más comunes phishing 6](#_Toc169211839)

[Daños causados por el phishing 7](#_Toc169211840)

[1.1. Motivación 10](#_Toc169211841)

[1.2. Objetivos 11](#_Toc169211842)

[2 AEstado del arte 13](#_Toc169211843)

[Comparativa Detallada 14](#_Toc169211844)

[3 DDiseño 15](#_Toc169211845)

[4 AAlgoritmos de comparación de cadenas 17](#_Toc169211846)

[2.1. Distancia de Hamming 17](#_Toc169211847)

[2.2. Distancia de Levenshtein 18](#_Toc169211848)

[2.3. Distancia de Damerau Levenshtein 19](#_Toc169211849)

[2.4. Índice de Jaccard 20](#_Toc169211850)

[2.5. Distancia de Cosine 21](#_Toc169211851)

[2.6. Distancia de Jaro Winkler 22](#_Toc169211852)

[2.7. Conclusión 23](#_Toc169211853)

[Bibliografía 25](#_Toc169211854)

[Apéndices 27](#_Toc169211855)

[Apéndice A 29](#_Toc169211856)

### Índice de figuras

X.X Figura 1 XX

### Índice de tablas

X.X Tabla 1 XX

### Índice de cuadros

X.X Cuadro 1 XX

# Introducción

**El phishing** es uno de los métodos de fraude más antiguos y persistentes en la era de Internet. Los ciberdelincuentes buscan interceptar información sensible, como contraseñas, datos bancarios y de pago, utilizando técnicas de ingeniería social, correos electrónicos de phishing y malware. Originalmente, el phishing clásico se basaba en enlaces y archivos adjuntos que contenían redirecciones maliciosas o descargas de malware. Sin embargo, los métodos modernos han evolucionado y no siempre dependen de la entrega involuntaria de datos sensibles.

El término "phishing" se deriva de la palabra inglesa “fishing” (pescar), ya que las víctimas son atraídas como peces hacia un anzuelo. Los correos electrónicos o mensajes falsos que parecen provenir de bancos, servicios de suscripción, amigos o compañeros de trabajo actúan como señuelos. Lamentablemente, las víctimas suelen darse cuenta demasiado tarde de que han sido engañadas.

Incluso antes de la llegada de Internet, el robo de datos era una práctica común entre los delincuentes. Métodos como el “shoulder surfing” (espiar por encima del hombro) se utilizaban para obtener información valiosa, como números PIN, direcciones y datos bancarios. Con el auge de Internet, el phishing ha surgido como la evolución natural del robo de datos. Ejemplos comunes incluyen correos electrónicos que simulan ser de un banco solicitando acciones urgentes o falsas notificaciones de entregas de Amazon.

El objetivo del phishing siempre es el mismo: obtener datos personales y sensibles, como datos bancarios, números de tarjetas de crédito, contraseñas para la banca en línea, cuentas de correo electrónico o accesos a sitios web. Cuanta más información personal y sensible logren recolectar, más valiosa se vuelve. Los estafadores persuaden a sus víctimas para que ingresen su información en páginas fraudulentas, lo que puede llevar a pérdidas financieras, suplantación de identidad y otros ataques a contactos personales o corporativos.

El phishing también suele ser la etapa preliminar para otros ataques cibernéticos, como malware, ransomware, spyware y scareware. Los archivos adjuntos en correos electrónicos de phishing pueden contener macros o códigos maliciosos que instalan software dañino en las computadoras de las víctimas.

##### Tipos de Phishing

El phishing, al igual que las tecnologías y la competencia digital, está sujeto a cambios constantes en sus procedimientos y métodos por parte de los defraudadores. En sus inicios, el phishing clásico dependía de la colaboración involuntaria de las víctimas, quienes solían divulgar datos personales o hacer clic en enlaces y archivos adjuntos maliciosos. Sin embargo, los nuevos métodos de phishing han evolucionado significativamente y ya no se limitan a estas interacciones directas.

Hoy en día, se han identificado más de 10,000 formas diferentes de phishing, cada una diseñada para engañar a las personas de maneras cada vez más sofisticadas. Entre los tipos de ataques de phishing más frecuentes se encuentran:

**Phishing general o tradicional**

Este método implica el envío masivo de correos electrónicos a usuarios, donde los mensajes falsifican entidades de confianza como bancos, con el objetivo de engañar al usuario y obtener información sensible. Por ejemplo, estos correos pueden incluir enlaces que aparentan llevar a URLs legítimas, pero que en realidad redirigen a sitios maliciosos. Es común que el texto del enlace muestre la URL correcta para camuflar su verdadero destino malicioso.

**Vishing**

Es similar al phishing tradicional pero el engaño se produce a través de una llamada telefónica. El término deriva de la unión de dos palabras en inglés: ‘'voice'’ y ‘'phishing’'. Un ejemplo típico de uso de esta técnica es cuando un ciberdelincuente ha robado ya información confidencial a través de un ataque de phising, pero necesita la clave SMS o token digital para realizar y validar una operación. Es en ese momento el ciberdelincuente llama por teléfono al cliente identificándose como personal del banco y, con mensajes particularmente alarmistas, intenta que el cliente revele el número de su clave SMS o token digital, que son los necesarios para autorizar la transacción.

**Qrishing**

phishing a través de códigos QR el cual consiste en la manipulación de códigos QR y el posterior engaño a las víctimas mediante la suplantanción de la página web o aplicación a la que se accede al escanear el código el cual dirige al usuario a un link fraudulento utilizado con la finalidad de obtener información privada de las víctimas.

**Smishing**

Es similar al phishing tradicional pero el engaño se produce a través mensajes de texto ya sean por SMS o [mensajería instantánea](https://es.wikipedia.org/wiki/Mensajer%C3%ADa_instant%C3%A1nea) (como [WhatsApp](https://es.wikipedia.org/wiki/WhatsApp)). Un ejemplo típico de esta técnica es cuando el cliente recibe un mensaje de texto, donde el emisor se hace pasar por el banco, y le informan que se ha realizado una compra sospechosa con su tarjeta de crédito. A su vez, el texto solicita que se comunique con la banca por teléfono de la entidad financiera y le brinda un número falso. El cliente devuelve la llamada y es ahí cuando el ciberdelincuente, haciéndose pasar por el banco, solicita información confidencial para supuestamente cancelar la compra. En una variante de esta modalidad el mensaje también podría incluir un enlace a una ‘web’ fraudulenta para solicitar información sensible.

**URL Phishing**

Se trata de engañar al usuario haciendo que una [URL](https://es.wikipedia.org/wiki/Localizador_de_recursos_uniforme) de un sitio malicioso parezca la de un sitio confiable. A ellas a veces se accede de forma inadvertida al escribir nombres de dominio mal escritos que están muy cerca del dominio legítimo, o siguiendo un enlace malicioso que parece correcto, o por [engaños al usar caracteres unicode parecidos difícilmente detectables](https://es.wikipedia.org/wiki/Nombre_de_dominio_internacionalizado#Consideraciones_por_enga%C3%B1os), especialmente en dispositivos móviles, con pantallas más pequeñas y generalmente una resolución de pantalla inferior. Por ejemplo caracteres latinos con un punto bajo o la letra griega ómicron, "ο".[19](https://es.wikipedia.org/wiki/Phishing#cite_note-19)​[20](https://es.wikipedia.org/wiki/Phishing#cite_note-20)​[21](https://es.wikipedia.org/wiki/Phishing#cite_note-21)​. Otra forma de disfrazar enlaces es utilizar direcciones que contengan el carácter [arroba](https://es.wikipedia.org/wiki/Arroba_(s%C3%ADmbolo)): @, para posteriormente preguntar el nombre de usuario y contraseña (contrario a los estándares[22](https://es.wikipedia.org/wiki/Phishing#cite_note-22)​). Por ejemplo, el enlace  http://www.google.com@members.tripod.com/ puede engañar a un observador casual y hacerlo creer que el enlace va a abrir en la página de www.google.com, cuando realmente el enlace envía al navegador a la página de members.tripod.com (y al intentar entrar con el nombre de usuario de www.google.com, si no existe tal usuario, la página abrirá normalmente). Este método ha sido erradicado desde entonces en los [navegadores](https://es.wikipedia.org/wiki/Navegador_web) de [Mozilla](https://es.wikipedia.org/wiki/Fundaci%C3%B3n_Mozilla)​ e [Internet Explorer](https://es.wikipedia.org/wiki/Internet_Explorer)

**Whaling**

Se diferencia de los otros tipos de intentos de phishing en que el objetivo son personas importantes como por ejemplo ejecutivos de alto rango (las ballenas, peces gordos de la empresa). Las solicitudes de información contenidas en el ataque están más adaptadas a la persona concreta. Por ejemplo, la información presentada puede incluir solicitudes de citaciones, quejas de clientes, solicitudes de transferencia bancaria u otras solicitudes relacionadas con transacciones financieras concretas. La persona objetivo desprevenida puede verse atraído a revelar información confidencial del sistema u otros datos valiosos a los que solo unos pocos individuos tienen acceso.

**Spear Phishing**

El objetivo es engañar es una persona o empleado específico de una compañía en concreto. Para ello los cibercriminales recopilan meticulosamente información sobre la víctima para conseguir su confianza. Un correo malicioso de spear phishing bien elaborado (típicamente con enlace a sitio malicioso o con documento adjunto malicioso) es muy complicado de distinguir de uno legítimo, por lo que acaba siendo más fácil cazar a la presa. El spear phishing es una herramienta típica usada en ataques a empresas, bancos o personas influyentes y es el método de infección más utilizado en campañas de [APT](https://es.wikipedia.org/wiki/Amenaza_persistente_avanzada). Los objetivos de este tipo de ataque son tanto los altos cargos con acceso a información potencial, como los departamentos cuyo trabajo consiste en abrir numerosos documentos provenientes de otras fuentes.

**Search Engine phishing**

En este tipo de ataque los estafadores crean su propio sitio malicioso y lo indexan los motores de búsqueda legítimos. Es habitual que estos sitios maliciosos ofrezcan productos barato, oportunidades de empleo o incluso lancen alertas de virus para los que es necesario adquirir su [antivirus](https://es.wikipedia.org/wiki/Antivirus). Los compradores en línea encontrarán estos sitios apareciendo en una página típica de resultados de Google, y puede ser muy difícil notar la diferencia con un sitio legítimo. El sitio malicioso alienta a los usuarios a entregar su información personal, como el número del documento de identificación o su número de cuenta bancaria para poder realizar la compra. Estos datos se pueden usar para robarle, secuestrar su identidad o destruir su reputación.[28](https://es.wikipedia.org/wiki/Phishing#cite_note-28)​

**Phising con evasión de filtros**

Los mecanismos 'Anti-Phishing' y 'Antimalware' disponen de mecanismos para detectar ataques. Por ejemplo:

* + Conocen las palabras clave que deben buscar en correos electrónicos para detectar ataques de phishing. Para evadir esos filtros los atacantes usan técnicas de para evadir esos filtros. Para evitarlo se pueden usar imágenes que contienen texto de phishing incrustado para evitar filtros anti-phishing basados en el análisis de cadenas de texto.
  + Analizan los ficheros adjuntos para detectar ataques. Para evitarlo se pueden añadir ficheros adjuntos protegidos con contraseña. Esta técnica a la vez que evita el análisis de detección crea una falsa sensación de seguridad.[29](https://es.wikipedia.org/wiki/Phishing#cite_note-29)​

[**Pharming**](https://es.wikipedia.org/wiki/Pharming)**o DNS-Based Phishing**

El engaño consiste en redirigir al usuario a una sitio falso aprovechando para ello vulnerabilidades en el proceso de conversión de la secuencia de letras que componen una URL en una [dirección IP](https://es.wikipedia.org/wiki/Direcci%C3%B3n_IP). El ataque puede ser dirigido contra el ordenador del usuario o aprovechar vulnerabilidad del servidor [DNS](https://es.wikipedia.org/wiki/Domain_Name_System). El término "pharming" es una palabra compuesta por los términos "phishing" y "farming".

[**Malware**](https://es.wikipedia.org/wiki/Malware)**-based phishing**

Se refiere a aquellos ataques de phishing que implican la ejecución de un software malicioso en los ordenadores de la víctima. Por ejemplo, en un correo electrónico que suplanta la identidad de una marca se incluye como adjunto, o es accesible a través de un enlace, un documento [PDF](https://es.wikipedia.org/wiki/PDF) que al abrirse infecta el dispositivo de la víctima

[**Watering Hole Phishing, watering hole attack o ataque de abrevadero**](https://es.wikipedia.org/wiki/Ataque_de_abrevadero)

El atacante infecta con malware sitios web de terceros muy utilizados por los usuarios de la organización. De esta forma cuando los usuarios de la organización acceden a ese sitio web quedan infectados. El ataque es altamente efectivo ya que, con la infección de un solo sitio, se puede lograr que miles de víctimas descarguen la amenaza. El éxito se incrementa si se usa vulnerabilidades 0-Day, no conocidas aun públicamente y que no han sido solucionadas por el fabricante. Su nombre proviene de la forma en que algunos depredadores del mundo animal esperan su oportunidad para atacar a su presa cerca de los pozos de agua que sirven de abrevadero.

**Evil Twin**

Se trata de crear un Punto de Acceso malicioso a una red Wireless, con apariencia de legítimo, para que las víctimas puedan conectarse y así capturar información confidencial. Por ejemplo, redirigiendo a sitios maliciosos que capturan nuestras credenciales.

**Social Network Phishing**

Son ataques de phishing en los que están involucradas las redes sociales. Por ejemplo:

* + Phishing de inyección de contenido en redes sociales consiste en insertar contenido malicioso en las redes sociales. Por ejemplo, publicación de post falsos publicados por usuarios cuyas cuentas se vieron afectadas con aplicaciones no autorizadas.
  + Man-in-the-middle social network attack también conocido como social network session hijacking attack. Es una forma de phishing en la que el atacante se posiciona entre el usuario y el sitio web de una red social legítima. La información que se manda a la red social pasa a través del atacante el cual la lee, la procesa e incluso puede añadir contenido. La forma en que el atacante se sitúa entre el usuario y la red social pueden ser variadas, por ejemplo, se puede aprovechar de una vulnerabilidad de la red social, o atraer a la víctima a un sitio de phishing (por ejemplo, una página de inicio de sesión falsa de Facebook) donde la víctima ingresa su nombre de usuario y contraseña que el servidor de phisher utiliza para ingresar al sitio web legítimo de la red social y actualizar y leer en la red social legítima.
  + Basado en malware. En este tipo de ataque se realiza la propagación de mensajes de phishing mediante el uso de malware. Por ejemplo, la cuenta de Facebook de una víctima que instaló una aplicación de Facebook no autorizada envía automáticamente mensajes a todos los amigos de la víctima. Dichos mensajes a menudo contienen enlaces que permiten a los receptores de los mensajes instalar la aplicación maliciosa de Facebook en sus computadoras o dispositivos móviles.
  + Deceptive phishing. En un escenario típico, un *phisher* crea una cuenta que finge ser la cuenta de la víctima. A continuación, el phisher envía solicitudes de amistad a los amigos de la víctima, así como un mensaje como «He abandonado mi cuenta de Facebook anterior. De ahora en adelante, comuníquese conmigo solo a través de esta cuenta». A continuación, el phisher comienza a enviar mensajes a los amigos de la víctima que exigen que el destinatario haga clic en un enlace. Por ejemplo, una factura ficticia que se puede cancelar haciendo clic en un enlace que solicita al usuario que proporcione su información personal.

**Phishing móvil**

Son ataques de phishing especialmente orientados a los dispositivos móviles. Algunos de estos son:

* + Aprovechar mensajes de SMS o de aplicaciones de [mensajería instantánea](https://es.wikipedia.org/wiki/Mensajer%C3%ADa_instant%C3%A1nea), por ejemplo, WhatsApp, para mandar enlaces falsos.
  + Aprovechar [aplicaciones móviles](https://es.wikipedia.org/wiki/Aplicaci%C3%B3n_m%C3%B3vil) maliciosas para recopilar información personal. Los datos pueden ser introducidos por el usuario o obtenidos directamente por la app, por ejemplo, accediendo a los ficheros del dispositivo o usando la información de geolocalización.
  + Obtener información personal como dónde vivimos o dónde estamos en un preciso momento, a partir de aplicaciones de venta de objetos de segunda mano como Vibbo o Wallapop
  + Aprovechar información de valoraciones de restaurantes y otros sitios de interés turístico para averiguar dónde está la víctima o para tener información sobre ella. Esta información puede ser usada, por ejemplo, para un primer contacto de tal manera que parezcan personas de confianza

##### Estrategias más comunes phishing

Existen dos enfoques comunes en el phishing: el spear phishing, dirigido a objetivos específicos con un uso avanzado de ingeniería social, y los ataques masivos de phishing, que se enfocan en una amplia base de posibles víctimas.

Spear Phishing:

El spear phishing implica que los ciberdelincuentes investigan a un pequeño grupo o a una única víctima. Utilizan la ingeniería social para recopilar información pública, como direcciones de correo electrónico, conexiones en redes sociales, historial profesional y cargos laborales. Luego, los delincuentes crean correos electrónicos que parecen genuinos al suplantar la identidad de contactos, empresas, bancos o conocidos. Estos correos a menudo contienen enlaces a páginas web falsas diseñadas profesionalmente, donde se solicitan contraseñas, datos bancarios u otra información confidencial. También pueden incitar a la víctima a abrir archivos adjuntos maliciosos. Es importante destacar que el spear phishing puede ser utilizado para preparar ataques a gran escala contra empresas o para el robo de activos corporativos mediante la recopilación de información de altos directivos.

Phishing masivo:

En contraste con el sofisticado spear phishing, las campañas de phishing de base amplia se enfocan en la cantidad de víctimas. Estos ataques suelen ser reconocibles por direcciones de correo electrónico evidentemente falsas, redirecciones a páginas web con URLs sospechosas o sin cifrar, y errores gramaticales notables. Los correos electrónicos pueden hacerse pasar por servicios de paquetería o de pedidos, incluso si el destinatario no ha realizado ninguna compra, o pueden simular ser de plataformas como Amazon o PayPal, aunque la víctima no tenga cuenta en dichos servicios. El objetivo principal de este tipo de phishing es obtener la mayor cantidad posible de datos sensibles a través de un gran número de posibles víctimas.

Otras tácticas de phishing:

Las nuevas técnicas de phishing ya no requieren la interacción directa de la víctima. Por ejemplo, no es necesario que la víctima haga clic en enlaces peligrosos o introduzca datos. En su lugar, los ciberdelincuentes pueden iniciar un ataque de intermediario o man-in-the-middle (MITM) mediante una página web o un correo electrónico infectado con código malicioso. En un ataque MITM, los delincuentes se interponen entre el dispositivo de la víctima y los servidores de Internet, interceptando comunicaciones y datos sensibles sin ser detectados fácilmente.

##### Daños causados por el phishing

Los daños causados por el phishing varían desde la pérdida del acceso al correo electrónico hasta importantes pérdidas económicas. Este método de robo de identidad está ganando popularidad debido a lo fácil que resulta para personas confiadas revelar información personal a los phishers, como números de tarjetas de crédito y de seguridad social. Una vez que obtienen esta información, los phishers pueden crear cuentas falsas a nombre de la víctima, realizar compras con su crédito e incluso bloquear el acceso de las víctimas a sus propias cuentas. Además, es crucial considerar el impacto psicológico y emocional que el robo de identidad puede causar al comprometer esta información personal.

Con el incremento de la dependencia de las empresas en los canales digitales de comunicación, los ciberdelincuentes aprovechan las vulnerabilidades en el correo electrónico, SMS y comunicaciones de voz para llevar a cabo ataques de phishing cada vez más sofisticados. El aumento del trabajo remoto, exacerbado por la pandemia de COVID-19 en los últimos años, ha elevado aún más el riesgo de estos ataques.

Según un informe del Centro de Denuncias de Delitos en Internet (IC3) del FBI, se recibieron 800,944 denuncias de phishing en 2022, con pérdidas que superaron los 10,300 millones de dólares. El informe también reveló que las personas de entre 30 y 39 años fueron las que más denunciaron ser víctimas de estafas de phishing, mientras que los ciudadanos mayores de 60 años sufrieron las mayores pérdidas económicas.

Además, el Informe 2023 de IBM sobre la violación de datos indicó que el costo promedio mundial de una violación de datos fue de 4.45 millones de dólares, con un promedio de 9.48 millones de dólares en los Estados Unidos. A pesar de que las reclamaciones por estafas en Internet disminuyeron de 2021 a 2022 según el Informe IC3 2022, las pérdidas totales aumentaron significativamente, pasando de 6,900 millones de dólares en 2021 a 10,300 millones de dólares en 2022.A graph of a number of companies

Description automatically generated with medium confidence

Estadísticas de phishing por sector

El Informe IBM 2023 sobre el Coste de una Filtración de Datos revela los sectores más afectados económicamente por estas violaciones. Durante los últimos 13 años, el sector de la salud ha liderado como el más costoso en cuanto a violaciones de datos.

En 2022, la tecnología ocupaba la cuarta posición en pérdidas económicas, pero en 2023 fue desplazada por los sectores de energía e industria. Por otro lado, tanto la industria farmacéutica como la financiera experimentaron una ligera disminución en los costes asociados a estas filtraciones.

A screenshot of a phone

Description automatically generated

Los ataques de phishing continúan representando una de las amenazas más significativas para las organizaciones en la actualidad. A medida que las empresas dependen cada vez más de los canales de comunicación digital, los ciberdelincuentes explotan vulnerabilidades en el correo electrónico, SMS y comunicaciones de voz para perpetrar ataques sofisticados de phishing. El incremento del trabajo remoto impulsado por la pandemia de COVID-19 en los últimos años ha exacerbado este riesgo, aumentando la frecuencia y la complejidad de los ataques de phishing.

A graph of a number of people

Description automatically generated with medium confidence

El informe más reciente de phishing de Zscaler ThreatLabz revela un preocupante aumento del 47.2% en los ataques de phishing durante 2022 en comparación con el año anterior. Este incremento se debe a la creciente sofisticación de las técnicas utilizadas por los ciberdelincuentes para llevar a cabo ataques a gran escala. En 2022, el sector de la educación fue el más afectado, con un alarmante aumento del 576% en el número de ataques. En contraste, el sector minorista y mayorista experimentó una notable disminución del 67% en comparación con 2021.

A colorful pie chart with text

Description automatically generated

Las marcas de Microsoft, como OneDrive y Sharepoint, junto con el intercambio de criptomonedas Binance y los servicios de streaming ilegales, fueron los principales objetivos de los ataques según un informe reciente. Los cinco países más afectados fueron Estados Unidos, el Reino Unido, los Países Bajos, Rusia y Canadá. El informe, basado en el análisis de 280,000 millones de transacciones diarias y el bloqueo de 8,000 millones de ataques diarios, subraya el uso cada vez mayor de kits de phishing y herramientas de inteligencia artificial para llevar a cabo campañas altamente efectivas que pueden evadir los modelos de seguridad tradicionales, incluido el MFA.

A pie chart with text on it

Description automatically generated

#### Motivación

El rápido crecimiento y la sofisticación de las tecnologías han transformado nuestra vida diaria, ofreciendo una gran cantidad de beneficios y oportunidades. Sin embargo, esta evolución también ha traído consigo un incremento y creación de nuevas amenazas cibernéticas, entre las cuales el phishing destaca como una de las más prevalentes y peligrosas.

El phishing, una técnica de fraude que busca obtener información confidencial de los usuarios a través de la suplantación de identidad, ha evolucionado constantemente, adoptando nuevas estrategias. Las consecuencias de un ataque de phishing pueden ser muy graves, afectando tanto a individuos como a organizaciones, y resultando en pérdidas económicas, compromisos de seguridad y daños a la reputación.

En este contexto, surge la necesidad de desarrollar herramientas eficaces que puedan identificar y mitigar estas amenazas. La creación de una aplicación capaz de detectar URLs de phishing no solo ofrece una solución práctica y accesible para los usuarios, sino que también contribuye al fortalecimiento de la ciberseguridad en general.

Además, el análisis histórico de la evolución del phishing es fundamental para comprender las tácticas y técnicas empleadas por los atacantes a lo largo del tiempo. Este conocimiento permitirá anticipar futuras tendencias y mejorar continuamente las estrategias de detección y defensa.

La motivación principal detrás de este Trabajo es contribuir a la lucha contra el phishing mediante la combinación de investigación académica y desarrollo tecnológico. Al diseñar e implementar una aplicación que utilice algoritmos avanzados para detectar URLs maliciosas.

A través de este proyecto, se pretende:

1. **Mejorar la seguridad de los usuarios**: Proteger a los usuarios de internet de posibles fraudes y ataques, ayudándolos a navegar de manera segura.
2. **Aportar al campo de la ciberseguridad**: Contribuir al conocimiento y las prácticas en ciberseguridad.
3. **Desarrollar habilidades técnicas y analíticas**: Aplicar conocimientos adquiridos durante la carrera en un proyecto práctico y relevante, desarrollando competencias en programación, análisis de datos, y ciberseguridad.
4. **Fomentar la conciencia sobre el phishing**: Informar a los usuarios sobre las amenazas de phishing y la importancia de la ciberseguridad, promoviendo buenas prácticas y una cultura de seguridad en línea.

#### Objetivos

**Objetivos Generales**

1. **Desarrollar una aplicación eficiente y efectiva para detectar URLs de phishing.**

Crear una solución que pueda ser utilizada por usuarios finales para protegerse contra ataques de phishing.

1. **Estudiar y analizar la evolución del phishing a lo largo de los años.**

Investigar cómo han evolucionado las técnicas de phishing y las contramedidas desarrolladas para combatirlas.

**Objetivos Específicos**

1. **Identificar características clave de las URLs de phishing.**

* Investigar y documentar los patrones comunes y técnicas utilizadas en URLs de phishing.

1. **Analizar la evolución histórica del phishing.**
   * Revisar la literatura y estudios existentes sobre phishing para entender su evolución y tendencias.
   * Documentar las principales técnicas de phishing utilizadas en diferentes periodos.
2. **Implementar algoritmos de detección de phishing.**
   * Desarrollar e integrar algoritmos de aprendizaje automático y técnicas de análisis de datos para identificar URLs sospechosas.
3. **Evaluar la eficacia de los métodos de detección.**
   * Realizar pruebas exhaustivas para validar la precisión y el rendimiento de la aplicación en la identificación de URLs de phishing.
4. **Desarrollar una interfaz de usuario intuitiva.**
   * Diseñar y crear una interfaz de usuario que permita a los usuarios introducir y verificar URLs fácilmente.

**Objetivos Técnicos**

1. **Optimización del rendimiento de la aplicación.**
   * Asegurar que la aplicación funcione de manera eficiente en términos de tiempo de respuesta y uso de recursos.
2. **Implementación de medidas de seguridad.**
   * Garantizar que la aplicación sea segura y proteja los datos de los usuarios contra accesos no autorizados.

**Objetivos de Evaluación y Validación**

1. **Pruebas de usabilidad y experiencia del usuario.**
   * Realizar pruebas con usuarios para asegurar que la aplicación sea fácil de usar y cumpla con sus expectativas.
2. **Validación con casos reales.**
   * Probar la aplicación con un conjunto de URLs reales para evaluar su desempeño en condiciones del mundo real.
3. **Documentación exhaustiva del desarrollo.**
   * Crear documentación detallada del proceso de desarrollo, incluyendo metodologías, resultados de pruebas y posibles mejoras futuras.

**Objetivos de Implementación y Despliegue**

1. **Despliegue en un entorno de producción.**
   * Implementar la aplicación en un servidor o plataforma accesible para los usuarios finales.
2. **Mantenimiento y soporte post-despliegue.**
   * Establecer un plan de mantenimiento para asegurar que la aplicación se mantenga actualizada y funcional después de su despliegue inicial.

# AEstado del arte

Existen número productos en el mercado orientados a la detección de phishing en sitios web, cada uno de ellos ofrecen herramientas distintas para la detección de phishing, a continuación, una breve decisión de algunas de las soluciones más populares en el mercado:

1. **VirusTotal**

VirusTotal es una plataforma líder en la industria para el análisis de archivos y URLs. Permite a los usuarios cargar archivos o introducir URLs para verificar si son maliciosos utilizando múltiples motores antivirus y herramientas de análisis de comportamiento.

* + **Características Relevantes:** Proporciona informes detallados que incluyen resultados de detección de diversos motores antivirus, análisis de comportamiento y reputación de URLs y archivos.
  + **Comparación:** A diferencia de nuestra aplicación, VirusTotal se centra en el análisis profundo de URLs y archivos desde la perspectiva de múltiples motores antivirus y no se limita específicamente a la búsqueda de coincidencias con URLs recientes.

1. **PhishTank**

PhishTank es una comunidad en línea que recopila y comparte información sobre URLs de phishing reportadas por usuarios y verificadas por voluntarios.

* + **Características Relevantes:** Permite a los usuarios verificar si una URL es maliciosa o no mediante la búsqueda en su base de datos pública y el reporte comunitario.
  + **Comparación:** Aunque PhishTank es excelente para identificar URLs de phishing reportadas, se basa en la participación de la comunidad y no realiza una búsqueda proactiva de URLs recientemente creadas como lo hace nuestra aplicación.

1. **Google Safe Browsing**

Google Safe Browsing es un servicio proporcionado por Google que ayuda a proteger a los usuarios al identificar y advertir sobre sitios web inseguros a través de una lista de sitios web peligrosos conocidos.

* + **Características Relevantes:** Ofrece una API pública que permite a los desarrolladores integrar la verificación de URLs en sus aplicaciones para proteger a los usuarios contra sitios web maliciosos.
  + **Comparación:** A diferencia de nuestra aplicación, que busca coincidencias con URLs recientes, Google Safe Browsing se basa en una lista estática de sitios web peligrosos y no realiza una búsqueda activa de nuevas amenazas en tiempo real.

##### Comparativa Detallada

A continuación, se presenta una comparativa más detallada de las características clave entre los productos y proyectos mencionados y nuestra aplicación:

| **Característica** | **VirusTotal** | **PhishTank** | **Google Safe Browsing** | **Nuestra Aplicación** |
| --- | --- | --- | --- | --- |
| **Tipo de Análisis** | Análisis exhaustivo de archivos y URLs | Verificación de URLs de phishing reportadas | Verificación de URLs conocidas peligrosas | Búsqueda de coincidencias con URLs recientes |
| **Fuente de Datos** | Múltiples motores antivirus, análisis de comportamiento | Base de datos comunitaria de URLs de phishing | Lista estática de sitios web peligrosos | URLs creadas en las últimas 24 horas |
| **Enfoque Principal** | Seguridad y análisis profundo | Identificación y reporte comunitario de phishing | Protección preventiva basada en listas | Identificación proactiva basada en coincidencias y semejanza de urls |
| **Interfaz de Usuario** | Informes detallados, API para integración | Búsqueda y reporte de URLs de phishing | API para integración en aplicaciones | Funcionalidad de búsqueda y generación de informes vía mail |
| **Limitaciones** | Requiere carga de archivos o URLs específicas | Dependencia de reportes comunitarios | Basado en lista estática, puede no cubrir nuevos ataques | Depende de la frecuencia de actualización de URLs |

# DDiseño

En esta sección se detalla el diseño de la aplicación desarrollada para detectar y reportar URLs sospechosas creadas recientemente. El objetivo principal del diseño es cumplir con los requisitos funcionales y no funcionales necesarios para proporcionar una herramienta eficiente y segura que ayude a los usuarios a identificar posibles amenazas de phishing.

Requisitos de Ingeniería del Software

Los requisitos funcionales se centran en las capacidades clave de la aplicación, como la búsqueda de coincidencias de URLs, la generación de informes personalizables y el envío de estos informes por correo electrónico. Por otro lado, los requisitos no funcionales aseguran que la aplicación sea rápida, segura y accesible desde diferentes plataformas.

**Requisitos Funcionales:**

1. **Buscar Coincidencias de URLs:**
   * La aplicación debe permitir al usuario introducir una URL y buscar coincidencias con URLs creadas en las últimas 24 horas.
   * Debe mostrar resultados de URLs que coincidan con la URL introducida por el usuario.
2. **Generar Informe por Correo Electrónico:**
   * Después de realizar la búsqueda de coincidencias, la aplicación debe permitir al usuario generar un informe con las URLs sospechosas encontradas.
   * Debe proporcionar opciones para personalizar el contenido del informe antes de enviarlo por correo electrónico.
3. **Interfaz de Usuario Intuitiva:**
   * La interfaz de usuario debe ser amigable y fácil de usar.
   * Debe incluir formularios de entrada para la URL a buscar y opciones claras para generar y enviar informes por correo electrónico.

**Requisitos No Funcionales:**

1. **Rendimiento:**
   * La búsqueda de coincidencias y la generación de informes deben ser rápidas y eficientes, incluso con grandes volúmenes de datos.
2. **Seguridad:**
   * Debe garantizarse la seguridad de las URLs introducidas y los informes generados, evitando accesos no autorizados y filtrando contenido malicioso.
3. **Portabilidad:**
   * La aplicación debe ser compatible con diferentes sistemas operativos y dispositivos, asegurando su accesibilidad desde múltiples plataformas.

Diagrama de Secuencia

Para visualizar la interacción entre los componentes principales de la aplicación, se incluye un diagrama de secuencia que muestra cómo los usuarios interactúan con la interfaz de usuario para iniciar búsquedas, generar informes y enviarlos por correo electrónico. Este diagrama proporciona una representación clara de los procesos internos que ocurren desde la solicitud del usuario hasta la entrega del informe final.

El diseño detallado en este capítulo garantiza que la aplicación sea robusta y escalable, cumpliendo con las expectativas de rendimiento y seguridad necesarias para enfrentar las crecientes amenazas cibernéticas en un entorno digital cada vez más complejo y dinámico.

# AAlgoritmos de comparación de cadenas

En el mundo de la informática, la capacidad de comprender y procesar el lenguaje humano es una tarea fundamental y compleja. Aunque para los humanos puede resultar intuitivo reconocer y corregir errores tipográficos o malas ortografías en el texto, para las máquinas esta tarea puede resultar considerablemente más desafiante. La concordancia difusa de cadenas, también conocida como "fuzzy string matching", emerge como una técnica esencial para abordar este desafío al permitir la identificación de cadenas de texto que coinciden parcialmente pero no exactamente.

La distancia entre cadenas se basa en el reconocimiento de patrones y la comparación inteligente de cadenas de texto para encontrar similitudes incluso en presencia de errores tipográficos, malas ortografías o variaciones leves en las palabras. Esto es especialmente prominente en los motores de búsqueda, donde los usuarios a menudo introducen consultas con errores ortográficos o variantes de palabras.

Por ejemplo, si un usuario escribiera "Londin" en lugar de "London" en Google, la concordancia difusa de cadenas identificaría que "London" era la palabra buscada, y Google devolvería resultados de búsqueda para ella.

Sin embargo, la distancia entre cadenas va mucho más allá de la mera corrección ortográfica en los motores de búsqueda. Esta técnica tiene aplicaciones en una variedad de campos, desde la corrección automática de texto en procesadores de texto hasta la identificación de nombres de productos en bases de datos comerciales. En el ámbito médico, por ejemplo, la distancia entre cadenas puede ayudar a identificar variantes ortográficas de nombres de medicamentos o enfermedades en registros electrónicos de salud, facilitando así la búsqueda y el análisis de datos clínicos.

Además, en el ámbito de la seguridad informática, la concordancia difusa de cadenas tiene una gran importancia en la detección de fraudes y el análisis forense de datos.

Los investigadores utilizan esta técnica para identificar patrones sospechosos en correos electrónicos, mensajes de texto o registros de actividad en línea, así como para detectar Phishing, lo que les permite detectar posibles amenazas cibernéticas y tomar medidas preventivas.

#### 4.1. Distancia de Hamming

La Distancia de Hamming es un concepto fundamental en el campo de la teoría de la información y la codificación de datos. Su historia se remonta a la década de 1950, cuando el matemático estadounidense Richard Hamming desarrolló este concepto como parte de su trabajo en el diseño de códigos de corrección de errores para sistemas de comunicación digital. Richard Hamming fue pionero en la aplicación de técnicas matemáticas en el campo emergente de la informática y las telecomunicaciones. Durante su tiempo en los Laboratorios Bell de Investigación en Informática (Bell Labs), Hamming trabajó en una variedad de proyectos relacionados con la codificación de datos y la detección y corrección de errores en sistemas de transmisión digital. En teoría de la información se denomina distancia de Hamming a la efectividad de los códigos de bloque y depende de la diferencia entre una palabra de código válida y otra. Cuanto mayor sea esta diferencia, menor es la posibilidad de que un código válido se transforme en otro código válido por una serie de errores. A esta diferencia se le llama distancia de Hamming, y se define como el número de bits que tienen que cambiarse para transformar una palabra de código válida en otra palabra de código válida.

Si dos palabras de código difieren en una distancia d, se necesitan d errores para convertir una en la otra.

Por ejemplo:

La distancia Hamming entre 1011101 y 1001001 es 2.

La distancia Hamming entre 2143896 y 2233796 es 3.

La distancia Hamming entre "tener" y "reses" es 3.

El funcionamiento del algoritmo de la Distancia de Hamming es directo. Para calcular la distancia entre dos cadenas, simplemente se comparan carácter por carácter y se cuentan las posiciones en las que difieren. Este enfoque es especialmente útil en aplicaciones donde las cadenas representan datos binarios, como en la transmisión de información digital o la detección de errores en sistemas de almacenamiento. La Distancia de Hamming ha encontrado aplicaciones en una variedad de campos, incluidos los códigos de corrección de errores, la bioinformática, la criptografía y la teoría de la complejidad computacional. También, se emplea para el uso de detección de Phishing, nos permite cuantificar la similitud entre dos urls, La distancia de Hamming es un algoritmo con una base simple pero muy potente ya que nos permite detectar urls potenciales de phishing hacia una entidad. El principal inconveniente es que llas dos cadenas deben tener una misma longitud para que el algoritmo sea efectivo por lo que si un atacan suplanta a gruposantander.com con la url grupsantander.com, la distancia de haming no devolvería un análisis lo suficientemente completo com para detectar Phising.

A computer screen with text

Description automatically generated

#### 4.2. Distancia de Levenshtein

La distancia de Levenshtein, distancia de edición o distancia entre palabras es el número mínimo de operaciones requeridas para transformar una cadena de caracteres en otra, se usa ampliamente en teoría de la información y ciencias de la computación. Se entiende por operación, bien una inserción, eliminación o la sustitución de un carácter. Esta distancia recibe ese nombre en honor al científico ruso Vladimir Levenshtein, quien se ocupó de esta distancia en 1965. Es útil en programas que determinan cuán similares son dos cadenas de caracteres, como es el caso de los correctores ortográficos. Por ejemplo, la distancia de Levenshtein entre "casa" y "calle" es de 3 porque se necesitan al menos tres ediciones elementales para cambiar uno en el otro. casa → cala (sustitución de 's' por 'l') cala → calla (inserción de 'l' entre 'l' y 'a') calla → calle (sustitución de 'a' por 'e') Se le considera una generalización de la distancia de Hamming, que solo compara cadenas de la misma longitud y solo considera como operación la sustitución. Hay otras generalizaciones de la distancia de Levenshtein, como la distancia de Damerau-Levenshtein, que consideran el intercambio de dos caracteres como una operación. La distancia de Levenshtein tiene una amplia variedad de aplicaciones prácticas. Por ejemplo, en la corrección ortográfica, se utiliza para sugerir palabras correctas cuando una palabra está mal escrita. En la bioinformática, se emplea para comparar secuencias genéticas y determinar su similitud o parentesco evolutivo. En la detección de plagio, se utiliza para identificar similitudes entre textos y detectar posibles casos de copia no autorizada. Este algoritmo, tiene más alcance que la distancia de Hamming ya que nos permie comparar distancias entre cadenas de distinta longitud. El resultado del algoritmo, potemos cuantificarlo en un valor del 0 a 1 siendo 0 totalmente distinto y 1 una copia idéntica de la siguiente forma:

max\_len = max(len(linea), len(url))

levenshtein\_dist = levenshtein\_distance(linea, url)

similarity = 1 - (levenshtein\_dist / max\_len)

Siendo *similarity*, el valor entre 0 y 1 que nos cuantifica la similitud entre ambas cadenas.

A screen shot of a computer code

Description automatically generated

#### 4.3. Distancia de Damerau Levenshtein

La distancia de Damerau-Levenshtein es una variante de la distancia de Levenshtein que también tiene en cuenta las transposiciones de caracteres además de las operaciones de inserción, eliminación y sustitución. Esta métrica lleva el nombre de los científicos soviéticos Vladimir Damerau y Vladimir Levenshtein, quienes contribuyeron significativamente al desarrollo de la teoría de la información y la informática en la década de 1960. La distancia de Damerau-Levenshtein se utiliza para medir la similitud entre dos cadenas de caracteres al calcular la cantidad mínima de operaciones requeridas para convertir una cadena en la otra, incluyendo transposiciones adyacentes de caracteres. Esto la hace especialmente útil en aplicaciones donde las transposiciones son comunes, como la corrección ortográfica y la comparación de palabras. El algoritmo de Damerau-Levenshtein es una extensión del algoritmo de Levenshtein y utiliza una matriz similar para calcular la distancia entre las cadenas. Sin embargo, también considera las transposiciones de caracteres al determinar el camino óptimo a través de la matriz. Esto permite una mayor precisión en la medición de la similitud entre las cadenas y una mejor detección de errores ortográficos o de escritura. La distancia de Damerau-Levenshtein tiene una amplia variedad de aplicaciones en campos como la corrección ortográfica, la bioinformática, la detección de plagio y la comparación de textos. Su capacidad para tener en cuenta las transposiciones de caracteres la hace especialmente útil en situaciones donde las cadenas pueden contener errores tipográficos o variaciones leves en la ortografía. Con lo que esta versión mejorada del algoritmo anterior de Levenhstein es más precisa a la hora de detectar Phising en dos cadenas (urls). Aplicamos un razonomiento similar al anterior para cuentificar la similiritud entre dos cadenas y así, sacar un valor que nos permita detectar casos de phisings.

A computer screen shot of a program

Description automatically generated

#### 4.4. Índice de Jaccard

El ***índice de Jaccard ( IJ )*** o ***coeficiente de Jaccard ( IJ )*** mide el grado de similitud entre dos conjuntos, sea cual sea el tipo de elementos.

La formulación es la siguiente:

A mathematical equation with black text

Description automatically generated

Es decir, la cardinalidad de la intersección de ambos conjuntos dividida por la cardinalidad de su unión.

Siempre toma valores entre 0 y 1, correspondiente este último a la igualdad total entre ambos conjuntos.

En [ecología](https://es.wikipedia.org/wiki/Ecolog%C3%ADa) se usa para medir la similitud, disimilitud o distancias (estas dos últimas si se le resta el índice a 1) que existen entre dos estaciones de muestreo, con una formulación equivalente:[1](https://es.wikipedia.org/wiki/%C3%8Dndice_de_Jaccard#cite_note-1)​**IJ: c / (a+b-c)**

A mathematical equation with black text

Description automatically generated

Donde:

* **a:** es el número de especies presentes en la estación A.
* **b:** es el número de especies presentes en la estación B.
* **c:** es el número de especies presentes en ambas estaciones, A y B.

En [informática](https://es.wikipedia.org/wiki/Inform%C3%A1tica) se utiliza para medir la distancia entre vectores definidos sobre un espacio vectorial booleano (las componentes del vector sólo pueden ser 0 o 1).

**J(A,B) = |A ∧ B| / |A ∨ B|**

donde ∧ y ∨ son, respectivamente, las operaciones × (AND) y + (OR) de la lógica booleana, y |A|=∑ai.

En el caso de URLs, con el objetivo de detectar phishing, podemos considerar cada URL como un conjunto de tokens, como palabras o caracteres, dependiendo de la granularidad deseada.

Para utilizar el algoritmo de Jaccard en la detección de phishing de URLs, primero representamos cada URL como un conjunto de tokens. Luego, calculamos la similitud de Jaccard entre la URL sospechosa y una lista de URLs conocidas como legítimas. Si la similitud de Jaccard entre la URL sospechosa y alguna de las URLs legítimas no supera un umbral predefinido, podemos considerar que la URL sospechosa tiene una alta probabilidad de ser legítima. De lo contrario, podría tratarse de un intento de phishing.

A computer screen with white and orange text

Description automatically generated

#### 4.5. Distancia de Cosine

La similitud del coseno es una métrica que determina en qué se parecen dos vectores (palabras, oraciones, características) entre sí. Básicamente, es un ángulo entre dos vectores.



El rango de similitud está entre -1 y 1, donde -1 vectores absolutamente opuestos (python - seguridad del código) , 0 sin correlación (conocimiento universitario - trabajo) , 1 absolutamente similar (chatgpt - hype) . Se puede explicar nuevamente por los ángulos, recuerda el coseno de dos vectores que apuntan en direcciones diferentes. El ángulo entre ellos es 180, son opuestos, el coseno es igual a -1. También, se puede plantear la utilización de la distancia del coseno, que se define como 1 – similitud del coseno. Para detectar phishing y calcular similitud en las urls, cualquiera de las dos sería válida ya que el razonamiento para calcular el porcentaje de similitud sería el mismo, pero a la inversa.

A screen shot of a computer code

Description automatically generated

#### 4.6. Distancia de Jaro Winkler

La distancia de Levenshtein es una medida que cuantifica el número de operaciones requeridas para transformar una cadena de caracteres en otra. Sin embargo, otra métrica ampliamente utilizada para evaluar la similitud entre dos cadenas de texto es la distancia de Jaro-Winkler. Esta métrica, propuesta en 1990 por Winkler como una modificación de la distancia de Jaro desarrollada un año antes, otorga un peso adicional a las cadenas que comparten prefijos. A diferencia de la distancia de Levenshtein, el algoritmo de Jaro-Winkler proporciona directamente un valor de similitud entre 0 y 1, lo que elimina la necesidad de normalizar posteriormente los resultados. Esta característica hace que el algoritmo de Jaro-Winkler sea particularmente útil en aplicaciones donde se necesita una evaluación rápida y precisa de la similitud entre cadenas de texto. Al considerar las similitudes entre cadenas de texto, tanto la distancia de Levenshtein como la distancia de Jaro-Winkler ofrecen herramientas poderosas y complementarias. Mientras que la distancia de Levenshtein se centra en la cantidad de operaciones necesarias para igualar las cadenas, la distancia de Jaro-Winkler destaca la importancia de los prefijos compartidos y proporciona una medida más intuitiva de la similitud entre las cadenas.

A screen shot of a computer program

Description automatically generated

#### 4.7. Conclusión

Tras analizar los distintos algoritmos planteados para cuantificar la similitud entre dos cadenas como urls para detectar phishing se han encontrado los siguientes pros y contras para cada algoritmo expuesto:

**1. Distancia de Hamming:**

Pros:

Simplicidad: La distancia de Hamming es fácil de entender y calcular.

Eficiencia: Es eficiente en la comparación de cadenas de igual longitud, lo que la convierte en una opción rápida para detectar discrepancias en URLs de longitud fija.

Contras:

Longitud fija de las cadenas: Limitada a cadenas de igual longitud.

Sensibilidad a transposiciones: No detecta transposiciones de caracteres, lo que podría pasar por alto casos de phishing donde los atacantes modifican el orden de los caracteres en las URLs.

**2. Distancia de Levenshtein:**

Pros:

Flexibilidad: Puede manejar cadenas de diferentes longitudes.

Sensibilidad a errores: Es capaz de detectar una amplia gama de cambios entre cadenas.

Contras:

Costo computacional: El cálculo de la distancia de Levenshtein es computacionalmente más costoso.

Sensibilidad a transposiciones: No detecta transposiciones de caracteres, lo que podría pasar por alto casos de phishing donde los atacantes modifican el orden de los caracteres en las URLs.

**3. Distancia de Levenshtein Damerau:**

Pros:

Manejo de transposiciones: A diferencia de la distancia de Levenshtein estándar, tiene en cuenta las transposiciones de caracteres.

Sensibilidad a errores: Al igual que la distancia de Levenshtein, puede detectar inserciones, eliminaciones, sustituciones y transposiciones de caracteres.

Contras:

Costo computacional: Similar a la distancia de Levenshtein, puede ser computacionalmente costoso.

Sensibilidad a la longitud de las cadenas: A medida que aumenta la longitud de las cadenas, el costo computacional también aumenta, lo que puede limitar su aplicabilidad en la detección de phishing en URLs largas.

**4. Algoritmo de Jaccard:**

Pros:

Simplicidad: Calcula la similitud entre conjuntos de elementos.

Flexibilidad: No requiere que las cadenas tengan la misma longitud.

Contras:

Sensibilidad al contenido del conjunto: La similitud de Jaccard puede ser influenciada por el contenido específico del conjunto de elementos.

Limitación a conjuntos: Se basa en la representación de las cadenas como conjuntos de elementos, lo que puede limitar su aplicabilidad en la detección de phishing en URLs que requieren una comparación más precisa de los caracteres individuales.

**5. Algoritmo de Cosine:**

Pros:

Representación vectorial: Calcula la similitud entre dos vectores.

Flexibilidad: No requiere que las cadenas tengan la misma longitud.

Contras:

Sensibilidad a la representación vectorial: La calidad de la representación vectorial puede afectar la precisión del algoritmo, especialmente en la detección de phishing en URLs donde la representación de términos puede variar.

Sensibilidad a la longitud de las cadenas: A medida que aumenta la longitud de las cadenas, el espacio vectorial también puede aumentar, lo que puede afectar la eficiencia computacional en el cálculo de la similitud.

**6. Algoritmo de Jaro-Winkler:**

Pros:

Consideración de prefijos compartidos: Otorga un mayor peso a los prefijos compartidos entre las cadenas, lo que puede mejorar la precisión en la detección de similitudes en URLs con prefijos similares.

Valor de similitud directa: Proporciona un valor de similitud entre 0 y 1, lo que facilita la interpretación de los resultados.

Contras:

Sensibilidad a la longitud de las cadenas: Puede ser menos efectivo en la comparación de URLs de diferentes longitudes.

Limitación a cadenas de texto: Se basa en la comparación de caracteres individuales en las cadenas de texto, lo que puede limitar su aplicabilidad en la detección de phishing en URLs que requieren una comparación más precisa de los componentes de la URL.

Por lo que el algoritmo de los anteriormente expuestos con sus pros y cntras, el que mejor se ajusta a la búsqueda de urls potenciales de phishing a partir de la url original es el de **Jaro-Winkler.** Por ejemplo, en una muestra de las últimas urls creadas a día 2 de mayo de 2024, con la palabra **facebook.com** comourl legítima nos avisa de las siguientes urls con un umbral de similitud establecido al 80%:

Jaro-Winkler distance between facebookagi.com

and facebook.com is 0.98 (98.33%)

Jaro-Winkler distance between facerbook.in

and facebook.com is 0.88 (88.85%)

Observamos grandes parecidos entre ambos dominós, lo cual puede llevar a confusión a la víctima.

A graph with blue squares

Description automatically generated

En el gráfico anterior, se muestra 4 algoritmos evaluados de 0 a 1, la similitud entre 10000 urls, es decir, cuanto de parecido es un fichero con 10.000 urls distintas entre si. Apreciamos con el algoritmo de Hamming, no es valido ya que urls de distinta longitud no son aptas para este algoritmo. Entre los otros tres comparados, vemos como Jaccard da poca coincidencia entre estos y Cosine da +0.8, (80%) de similitud, siendo excesivo, ya que hay cadenas que no se parecen entre si pero da un alto porcentaje de similitud debido a la naturaleza de la comparación del algoritmo. Jaro-Winkler, ofrece un termino medio y es el algoritmo que devulve unos resultados mas hacertados

### Bibliografía

<https://www.bbva.es/finanzas-vistazo/ciberseguridad/ataques-informaticos/que-es-el-phishing-y-cuales-son-sus-consecuencias.html>

<https://es.wikipedia.org/wiki/Phishing>

<https://www.proofpoint.com/es/threat-reference/phishing>

https://www.whoisds.com/newly-registered-domains

<https://www.ocu.org/tecnologia/antivirus/consejos/evitar-ataque-phishing>

<https://phishtank.org/>

https://www.virustotal.com/gui/home/upload

https://www.techopedia.com/es/estadisticas-sobre-phishing

https://www.zscaler.es/blogs/security-research/2023-phishing-report-reveals-47-2-surge-phishing-attacks-last-year

https://easydmarc.com/blog/es/estadisticas-de-phishing-informe-easydmarc-enero-junio-de-2022/

https://www.stormshield.com/es/noticias/breve-historia-del-phishing/

https://www.keepersecurity.com/es\_ES/threats/what-is-phishing.html

https://blog.grupomicronet.com/cuales-son-los-origenes-del-phishing

https://www.stormshield.com/es/noticias/breve-historia-del-phishing/

### Apéndices

A

### Apéndice A

Apéndice A