Escuela Politécnica Superior de Jaén
Máster Universitario en Seguridad Informática

UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior (Jaén)

Trabajo Fin de Máster

DETECCIÓN DE AUTORÍA EN LA WEB OSCURA

Alumno: Collado Montañez, Sebastián

Tutor: Prof. D. Arturo Montejo Ráez
Dpto.: Departamento de Informática

Noviembre, 2022

Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don Arturo Montejo Ráez, tutor del Trabajo Fin de Máster titulado: **Detección de autoría en la web oscura**, que presenta Sebastián Collado Montañez, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, noviembre de 2022

El alumno:

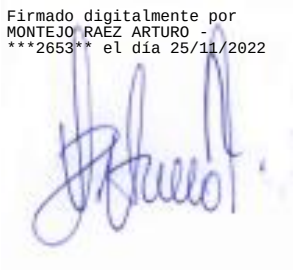
COLLADO MONTAÑEZ SEBASTIAN - 77368809Y

Firmado digitalmente por COLLADO MONTAÑEZ SEBASTIAN - 77368809Y
Fecha: 2022.11.25 00:05:49 +01'00'

Sebastián Collado Montañez

El tutor:

Firmado digitalmente por MONTEJO RAEZ ARTURO - ***2653** el día 25/11/2022



Arturo Montejo Ráez

Índice

Índice de ilustraciones

Índice de tablas

1. INTRODUCCIÓN

El presente documento refleja el resultado del trabajo de fin de máster enmarcado dentro de la titulación de Máster en Seguridad Informática de la Universidad de Jaén.

Con este proyecto se pretende abordar, con una nueva perspectiva adquirida durante el desarrollo del máster, un problema poco común desde el ámbito de la ciberseguridad, como es la autoría de textos en foros de Internet con mecanismo orientados a la protección del anonimato, y, sobre todo, qué información es posible desvelar acerca de un usuario a partir de cómo nos expresamos de forma escrita.

1.1. Contexto

A diario enviamos gran cantidad de datos e información a través de la red. De toda esta información, una parte se envía de manera consciente, como nuestras publicaciones, acciones y contenido multimedia; y otra parte no tanto, como, por ejemplo, la IP origen de la conexión, la ubicación geográfica, el navegador utilizado, su versión y extensiones instaladas, el sistema operativo, el idioma configurado del sistema, entre otros datos.

La *huella digital* o *fingerprint* [1] se construye en base al rastro que un individuo va dejando en la red. Este rastro está formado por datos que el individuo ha ido compartiendo, de forma consciente o inconsciente. Todos estos datos, tratados de forma aislada, pueden no suponer un problema mayor para la privacidad y seguridad del individuo, pero cuando se agrupan pueden dar lugar a un mayor riesgo de exposición de la identidad del individuo.

El *fingerprinting* [1] es el proceso mediante el cual un observador obtiene la suficiente cantidad de datos como para identificar únicamente y con una alta probabilidad a un individuo a partir de los mismos. Existen proyectos como Panopticlick [34] que nos permiten comprobar la cantidad de información que puede ser extraída de nuestro dispositivo sin que realmente prestemos atención a ello.

La *identidad digital* [2] se define como la representación del individuo en la red y está formada por el conjunto de acciones, omisiones, interacciones y habilidades que

el individuo muestra en la red. No siempre es deseado por parte de un individuo que su identidad digital refleje su identidad real, para lo que frecuentemente se recurre al anonimato.

El anonimato en la red es una navaja de doble filo: protege al individuo de posibles ataques, dota a las personas de libertad de expresión sin temer represalias, pero también facilita a otros a actuar al margen de las normas y leyes, en muchos casos, vulnerando los derechos fundamentales de los demás. Esta fina línea es motivo frecuente para la discusión.

Mantener una identidad anónima en la red, desde el punto de vista de la seguridad, es una tarea compleja y costosa, y que se suele pasar por alto en pos de la usabilidad. Esto se convierte en un problema cuando no se tienen en cuenta los riesgos derivados de publicar una información en la red que quede asociada a la identidad de ese individuo. Al igual que en el mundo real pueden tratar de hacerse pasar por nosotros o cometer actos ilegales en nuestro nombre, en el mundo digital no es diferente, siendo este sólo uno de los riesgos a los que se expone un usuario de Internet.

La web y red oscuras surgen con el objetivo de dotar de anonimato a las comunicaciones a través de la red. No obstante, de entre todos los datos enviados de forma inconsciente por el usuario, existen algunos que no son desvelados por la propia tecnología, sino que es la propia forma de ser y actuar del individuo la que desvela dichos datos, por ejemplo, sus hábitos horarios, las temáticas de interés y, por supuesto, el estilo de escritura. Es este rastro digital el que tomaremos como pieza clave en este proyecto.

1.2. Justificación

Los datos generados por los diferentes sistemas y tecnologías utilizados en la red no son lo único que puede ser usado para “des-anonimizar” a la persona que se sitúa detrás de un computador. El comportamiento del individuo en las diferentes plataformas sociales, sus interacciones y relaciones también son elementos que forman parte de su identidad, y que juegan en contra de su privacidad.

Los foros son una plataforma universalmente utilizada para el intercambio de opiniones, ideas, mercancía y cualquier otro tipo de relación interpersonal entre múltiples usuarios. Cada usuario tiene una manera de escribir y expresarse que puede ser utilizada como un indicador más que le identifique.

La *detección o atribución de autoría* [3] es una tarea objeto del Procesamiento del Lenguaje Natural (una de las ramas principales de la Inteligencia Artificial) que consiste en obtener información del autor en base a las características encontradas en los documentos escritos por dicho autor. Es un problema ampliamente tratado y con gran variedad de aplicaciones, la más frecuente y comercial es la detección de plagio.

Los avances técnicos y en investigación en áreas como el aprendizaje automático, la recuperación de la información y el procesamiento del lenguaje natural han dado un gran impulso a la detección de autoría [4], encontrando diferentes enfoques para abordar la tarea en base a las características que pueden ser analizadas de un texto: las palabras que se utilizan, cómo se ordenan entre ellas, el significado que tienen según el contexto o incluso la fecha de publicación de dicho texto en la red.

1.3. Objetivos

En los foros de la web oscura, que describiremos con detalle más adelante, encontramos diferentes nombres de usuario que representan a personas. En ocasiones, incluso pueden existir múltiples nombres de usuario que tengan detrás a un mismo individuo. El principal objetivo del proyecto es **estudiar la posibilidad de relacionar los textos que han sido escritos en la web oscura por un mismo individuo**, independientemente de su origen y el nombre de usuario utilizado.

Entre los objetivos secundarios se definen:

Explorar la web oscura en busca de foros de diferentes temáticas y evaluar la facilidad en el acceso y uso de los mismos.

Ampliar conocimientos sobre las diferentes tecnologías que entran en juego para obtener anonimato en la red.

Estudio de diferentes técnicas aplicadas para la detección de autoría en la web oscura.

1.4. Metodología

El proyecto planteado tiene un alto grado de incertidumbre a la hora de definir las tareas y el esfuerzo que requerirá acometerlas ya que inicialmente los dos temas principales a abordar son desconocidos para el autor: la web oscura y la detección de autoría. Otro factor a tener en cuenta es la no existencia de un cliente real que deba realizar revisiones durante las diferentes fases del proyecto, lo que aleja este proyecto de un desarrollo de ingeniería aplicada para enmarcarse en un proyecto de tipo experimental. Las diferentes etapas se han ido detallando conforme avanzaban las etapas previas y el proyecto, así, ha ido cogiendo forma de acuerdo a los nuevos descubrimientos y resultados de los diferentes experimentos llevados a cabo.

Los factores anteriormente mencionados nos inclinan a seleccionar un enfoque incremental para el desarrollo del proyecto, de modo que la toma de decisiones pueda hacerse conforme a los nuevos conocimientos que se iban obteniendo. Para ello se planificaron unas tareas generales que fueron progresando en conocimiento y funcionalidad a lo largo del desarrollo del proyecto.

2. PLANIFICACIÓN

2.1. Planificación temporal

El cómputo de horas a planificar se calcula teniendo en cuenta que el presente proyecto fin de máster consta de 12 créditos ECTS. Cada crédito ECTS equivale a un total de 25 horas de trabajo del estudiante [5], por lo tanto, tendríamos un total de 300 horas de dedicación para el proyecto.

El total de horas (300) se repartirá en un desempeño de 30 horas semanales, lo que resultaría en un total de 10 semanas para su finalización.

El proyecto se divide en las siguientes fases, las cuales se repartirán a lo largo de las 10 semanas indicadas anteriormente.

Planificación: Fase dedicada a la estimación de esfuerzo requerido para la realización del proyecto y distribución del mismo entre el tiempo disponible para su realización. Es importante tener en cuenta la base de conocimientos sobre las temáticas desarrolladas en el mismo. La planificación se ha ido revisando durante el seguimiento del proyecto, mostrando en esta memoria la planificación en su estado más maduro.

Documentación: Durante esta fase, desarrollada a lo largo de toda la duración del proyecto, se documentaron los nuevos conocimientos, hallazgos, hipótesis realizadas y resultados obtenidos, dando forma a lo que será la memoria final del proyecto de fin de máster.

Análisis: Estudio del sistema y tecnologías relacionado con la detección de autoría y la web oscura. Fase fundamental del proyecto donde se pretende extraer la mayor cantidad de información posible sobre lo que se está estudiando.

Diseño: Durante la fase de diseño se plantea el “cómo”, o el modo de afrontar los objetivos del proyecto. Se diseñó la manera para extraer textos desde la web oscura de forma automática, la forma de integrar tecnologías más adecuadas y los distintos experimentos que permitieron validar el enfoque o solución elegido.

Implementación: Llegados a esta fase, se tiene claro cuáles son los objetivos, con qué tecnologías y conceptos se está trabajando y se habrán planteado diferentes propuestas para llevar a cabo los experimentos. Ahora será el momento de hacer uso del conocimiento adquirido y ponerlo en práctica para el desarrollo de los experimentos (obtención de datos, implementación de algoritmos, entrenamiento y evaluación)

Conclusiones: En base a los resultados obtenidos en las fases anteriores, se extraen diferentes conclusiones que nos permiten conocer lo lejos o cerca que la solución elegida está de la detección de autoría en la web oscura mediante la obtención de información de un usuario en base a su escritura.

Planificación repartida en semanas desde el 29/8/2022 hasta 7/11/2022.

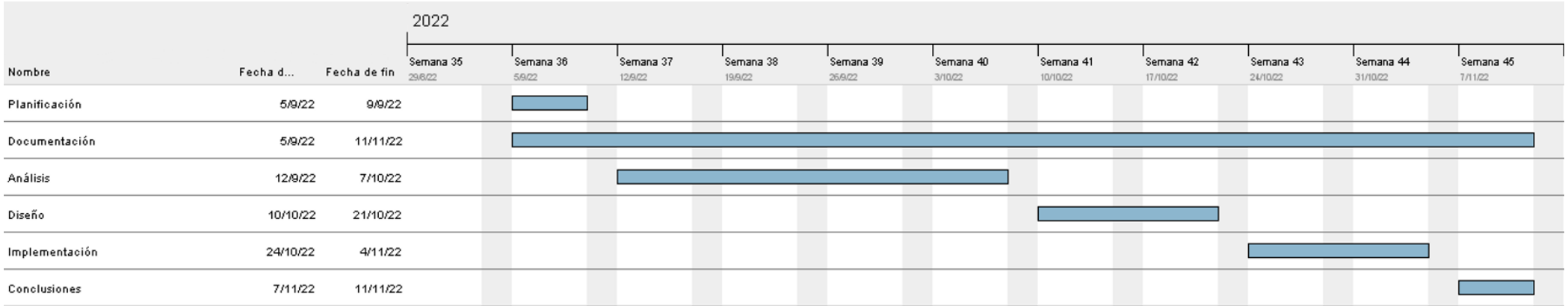


Ilustración 1. Planificación temporal

2.2. Planificación económica

Los recursos que se requieren para el desarrollo del proyecto se agrupan en hardware, software y recursos humanos.

2.2.1. Costes

En la siguiente tabla se muestran los diferentes bienes **hardware** utilizados para llevar a cabo el proyecto.

Concepto	Coste
Equipo sobremesa *Intel Core i7, 16GB RAM, 1 TB SSD, NVIDIA GTX 1660 SUPER, PSU 700W*	1000€
Periféricos *2x monitor 24", ratón, teclado*	300€

Tabla 1. Coste hardware

En la siguiente tabla se reflejan los costes de los bienes **software** utilizados para el desarrollo del proyecto. El único coste asociado es la herramienta para la redacción de la documentación, siendo este un servicio de suscripción anual.

Concepto	Coste
Sistema operativo Ubuntu 22.04 LTS	0€
Tor Browser	0€
Python y librerías asociadas	0€
Microsoft Office 365	69€/año

Tabla 2. Coste software

Para calcular el **coste humano** se tendrá como referencia un esfuerzo de 300 horas de trabajo de una única persona. El salario referencia se escoge de acuerdo al conjunto de competencias necesarias para llevar a cabo las correspondientes tareas. Para encontrar dichas referencias se hace uso del portal Glassdoor para cada uno de los perfiles.

Para calcular el coste por hora se tendrá como referencia un total de 1.696 horas anuales, según lo marcado en (BOE-A-2022-3092, Capítulo II, Art. 5).

Perfil	Coste año	Coste hora	Horas estimadas	Coste total estimado
Gestor de proyectos	40.000€	23€	50	1.150€
Desarrollador Python	31.000€	18€	50	900€
Ingeniero especialista en seguridad de la información	44.000€	25€	200	5.000€
Total				**7.050€**

Tabla 3. Coste recursos humanos

2.2.2. Amortización

La amortización refleja la pérdida de valor de un bien debida a su uso. Esta cantidad puede calcularse mediante diferentes métodos, en este caso se utilizará la amortización lineal. Mediante este tipo de amortización el valor del bien se reduce en la misma cuantía año tras año.

Para hacer el cálculo de la amortización de los bienes asociados al proyecto, se definen los siguientes conceptos:

Valor adquisición: Precio de los bienes en el momento de su adquisición.

Vida útil: Número de años tras los cuales el bien se considera amortizado.

Amortización anual: Pérdida de valor de un bien a lo largo de su vida útil.

Valor residual: Precio estimado de los bienes una vez finalizada su vida útil.

Se utilizarán las tablas oficiales [6] de la Agencia Tributaria que reflejan el coeficiente máximo de amortización y el máximo número de años de vida útil, basándose en el tipo de bien.

Para el caso del presente proyecto nos basaremos en los valores del grupo de bienes “Equipos para tratamiento de la información y sistemas y programas informáticos”. Estos valores definen un coeficiente lineal máximo del 26%, con un máximo de 10 años.

Para los bienes de tipo hardware aplicaremos un valor de vida útil de 5 años para calcular la amortización, lo que nos daría un coeficiente de amortización anual de un 10%.

Los bienes de tipo software son una suscripción anual, la cual una vez finalizada deja de ser utilizable. Por lo tanto, se considerará una vida útil de un año (la duración de la suscripción) y un valor residual nulo.

Concepto	Valor adquisición	Valor residual	Vida útil	Amortización anual	Amortización 5 meses
Hardware	1300€	200€	5 años	220€	91,66€
Software	69€	0	1 año	69€	28,75€
Total				**169€**	**120,41€**

Tabla 4. Presupuesto amortizaciones

2.2.3. Coste final

Para el cálculo final de los costes asociados al proyecto se tendrán en cuenta todos los costes derivados de la amortización de bienes hardware y software, junto al coste asociado a los recursos humanos. Además, se incluirá una variable de costes indirectos por un valor del 5% del total destinada a cubrir otros gastos no contemplados como son: el coste de la energía, la conexión a Internet y otros servicios similares.

Concepto	Importe
Hardware	91,66€
Software	28,75€
Recursos humanos	7.050€
Costes indirectos (5%)	(7.170,41 * 0.05) = 358,52€
Total	**7.528,93€**

Tabla 5. Presupuesto global

3. ANÁLISIS DEL SISTEMA

A continuación, se realiza un estudio del sistema, tecnologías implicadas y estudios existentes relacionados con el proyecto.

3.1. Anonimato en la red

Con frecuencia, cuando navegamos por Internet, tratamos de protegernos ante grupos de ciberdelincuentes que puedan hacerse con nuestras pertenencias, incluyendo nuestra información. Durante este estudio, se observan otros factores que normalmente pasamos por alto, pero que son igualmente importantes: ¿Todos los empleados que forman parte de la cadena de suministro que me da conexión a Internet miran por mi bien? ¿En cuántos puntos intermedios puede ser leída mi información hasta que llega al destinatario? ¿Vivo o estoy de viaje en un país en el cual la censura está afectando a la información que soy capaz de alcanzar desde mi conexión?

Cuando las respuestas a las preguntas anteriores se tornan negativas, la tecnología nos aporta alternativas que tratarán de cubrir, tanto estas, como otras tantas cuestiones relacionadas con la privacidad y anonimato.

La primera barrera que normalmente se aplica para la obtención de cierto grado de anonimato es la utilización de servidores *proxy*[1]. Un proxy no es más que un equipo informático que atiende la petición del usuario (1) y la reenvía hacia el servidor destino (2), posteriormente, devuelve la respuesta al usuario (3, 4). De este modo, el servidor destino recibirá la solicitud desde el proxy, ocultando la dirección del remitente original.

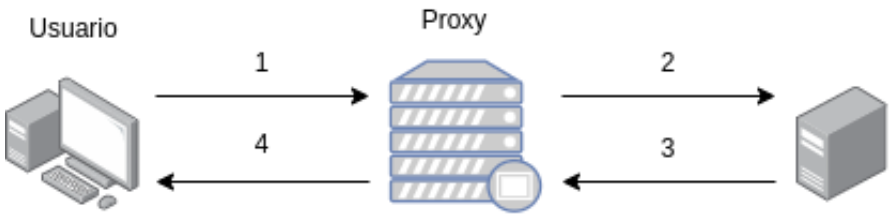


Ilustración 2. Esquema general de las comunicaciones a través de un proxy

[1] https://es.wikipedia.org/wiki/Servidor_proxy

El principal problema de un servidor proxy se encuentra en el intermediario, puesto que de ser un actor malicioso podría ver, modificar o incluso omitir parcial o completamente los datos enviados a través de él. Además, las comunicaciones que viajan a través del proxy no tienen porqué estar necesariamente cifradas si no se está utilizando un cifrado en la capa de aplicación, añadiendo un riesgo más en su utilización.

Las redes privadas virtuales (VPN) son otra opción que permite dotar de seguridad a las comunicaciones a través de redes públicas mediante túneles cifrados. En el caso de contratar un servicio VPN, el proveedor recibirá las peticiones y las enviará al servidor destino a través de uno de sus múltiples servidores, ocultando la dirección de origen del usuario emisor. La principal mejora respecto al proxy es que utilizan protocolos de comunicación cifrados entre el equipo del usuario y el servidor VPN, por lo tanto se reduce el riesgo de robo de información en el camino. Sigue presentando ciertos inconvenientes, dado que la entidad prestadora del servicio tendría control sobre la información que transporta, pudiendo leerla, manipularla o borrarla en caso de querer hacerlo. Además, este tipo de conexiones no son necesariamente anónimas puesto que la entidad prestadora podría siempre trazar e identificar al usuario en caso de necesitarlo o de requerimiento judicial.

Como último apunte sobre los servicios VPN, existen servicios, generalmente de pago, que ofrecen una política “no-log” en la cual aseguran no almacenar ningún registro que relacione al usuario con el uso que hace del servicio, disponiendo de auditorías que así lo verifican.

Más adelante introduciremos y estudiaremos más detalladamente la siguiente alternativa frecuentemente utilizada como es la red Tor [7].

3.2. La web oscura

En Internet, para que un pequeño paquete de datos llegue a su destino, cada punto de interconexión que se encuentre en su camino debe conocer desde dónde viene y hacia dónde va, lo que permitirá siempre identificar el origen y destinatario de la información [8]. Esta arquitectura es uno de los principales problemas a resolver para conseguir anonimato en Internet.

Todos los servicios ofrecidos sobre Internet sufren el problema mencionado anteriormente, siendo la World Wide Web uno de ellos [9]. Además, en el caso de la Web, se añaden también una serie de factores adicionales que juegan un papel importante en contra de la privacidad, como por ejemplo las extensiones, versión e idioma del navegador, Javascript, cookies, etc.

Las alternativas existentes para tratar de solucionar los problemas comentados anteriormente se basan en superponer diferentes capas lógicas sobre Internet, de modo que el camino que siguen los datos sea muy difícil de rastrear, aunque no es infalible, como veremos más adelante.

La red oscura es un conjunto de redes y tecnologías superpuestas a Internet, formada por una serie de nodos de interconexión, protocolos y algoritmos que tratan de dotar de anonimato a sus usuarios ocultando el origen y destino de la información que se envía a través de ella.

La web oscura es el conjunto de sitios web existentes dentro de las diferentes redes oscuras. Para su visualización se requieren navegadores y configuraciones específicas. Dadas las dificultades para el acceso a estos sitios web y su habitual corto tiempo de vida, los buscadores e indexadores de contenido no los ofrecen como resultado de sus búsquedas, haciendo que encontrar la información buscada dentro de la web oscura sea una tarea compleja.

3.3. Tor

La red Tor [10] es un proyecto *open source* formado por un grupo de nodos de interconexión desplegados por voluntarios de la propia comunidad con el objetivo de dotar de privacidad al usuario, protegiendo su identidad y evitando el rastreo de su tráfico.

La red Tor pone foco en dos puntos diferentes de la conexión del usuario, donde dota de privacidad:

En el origen de su conexión, es decir: frente a su proveedor de servicios de Internet (ISP) y frente a cualquier otro usuario que pudiera estar controlando su

conexión de forma local. Desde este punto no podrían trazar la actividad del usuario, incluyendo direcciones o nombres de servicios utilizados.

En el destino de sus conexiones, como serían los diferentes operadores de los sitios web y otros servicios accedidos por el usuario. Estos verán que el tráfico procede de la red Tor en lugar de la conexión original.

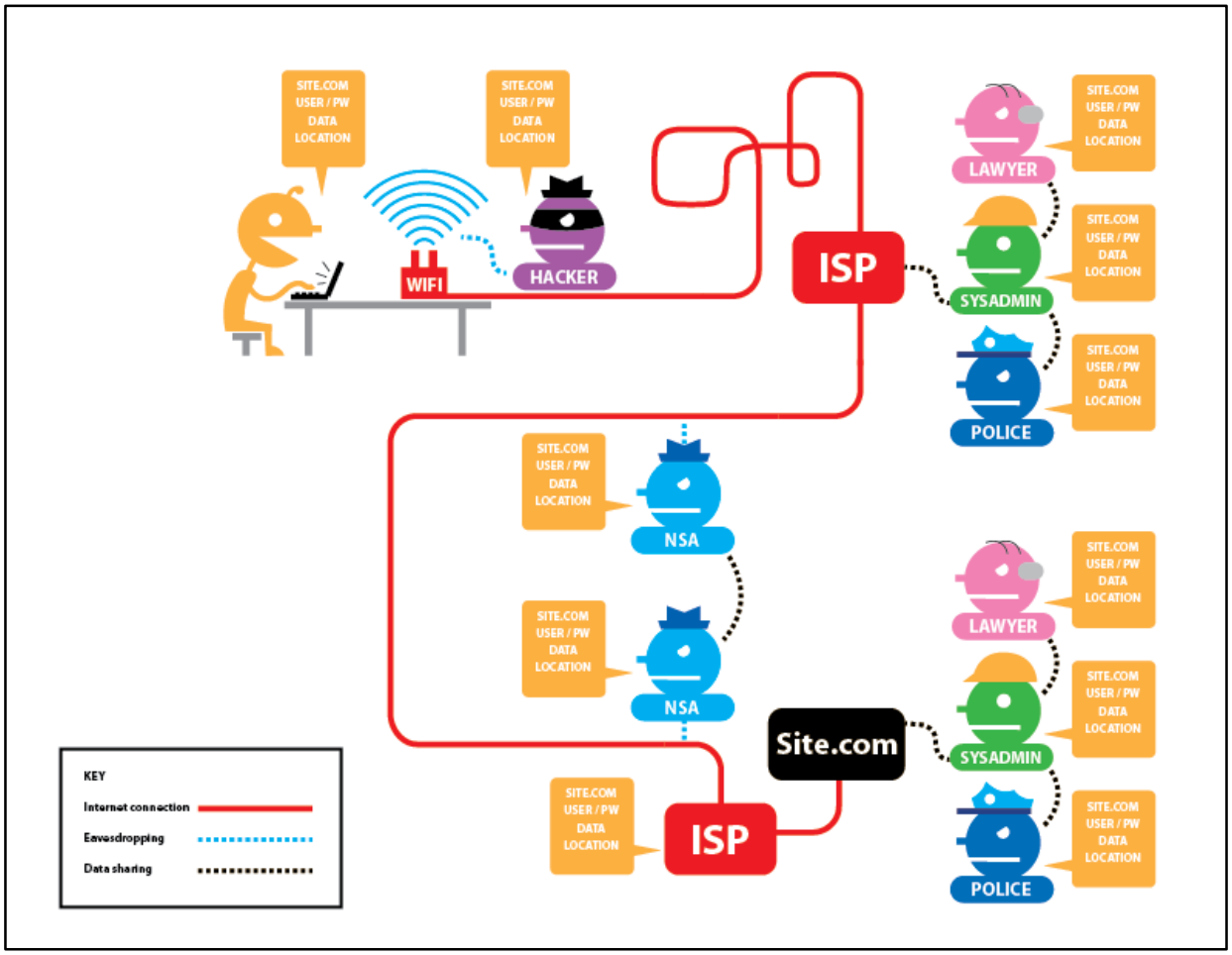


Ilustración 3. Actores que pueden intervenir las comunicaciones del usuario

En total, cada comunicación enviada pasará por un total de tres nodos aleatoriamente asignados de la red Tor. La comunicación entre los diferentes nodos se transmite de forma cifrada y no puede ser leída por los nodos intermedios. El último nodo del circuito es el encargado de enviar el tráfico a la red pública Internet (Ilustración 4).

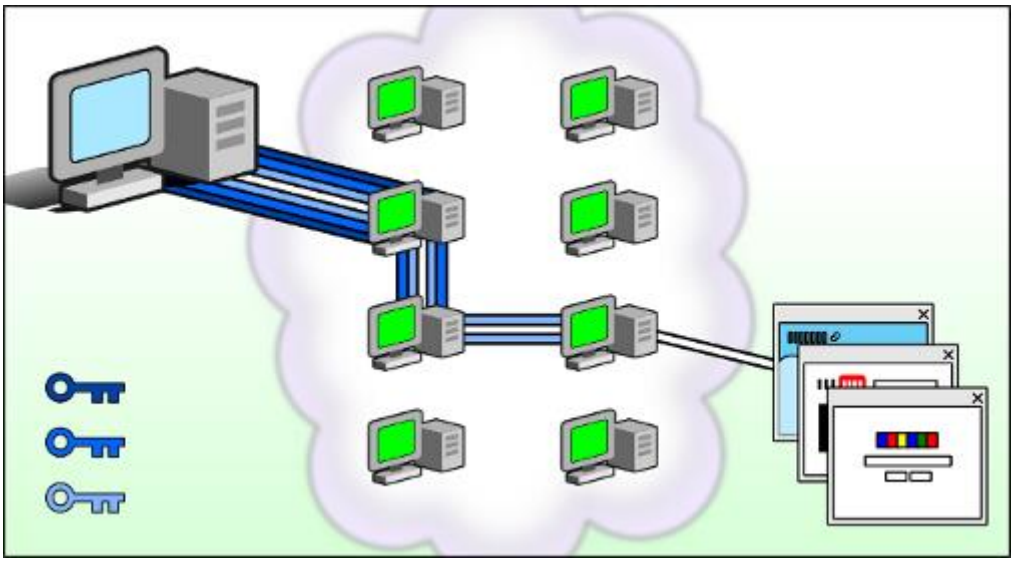


Ilustración 4. Funcionamiento de la red Tor

La arquitectura anteriormente descrita tiene un problema documentado [11] que ha sido explotado en varias ocasiones [12] [13] y ha sido motivo de compromiso de toda la red Tor. Los ataques se desarrollan en el nodo situado en el extremo final, encargado de devolver la información a la red pública. Por tanto, el cifrado de la propia red Tor es suprimido en este último punto. El problema se da cuando el propietario de este último nodo tiene intenciones maliciosas y realiza la lectura o manipulación sobre dicho tráfico antes de devolverlo a Internet. Este tipo de comportamiento es perseguido por los administradores de la red Tor y para minimizar el impacto de un posible nodo comprometido se recomienda siempre utilizar las versiones seguras de los protocolos en capa de aplicación, utilizando HTTPS en lugar de HTTP.

"Tor Browser" es la herramienta principal para navegar por la web utilizando Tor. Está diseñado específicamente para prevenir que los sitios web puedan identificar al usuario basándose en la configuración del propio navegador o mediante técnicas de perfilado en base a la huella digital o "fingerprinting". También se puede navegar en la red Tor a través del uso de extensiones sobre navegadores convencionales como Chrome, Firefox o Brave, no obstante, esto es algo poco recomendable puesto que dichos navegadores no están diseñados de forma expresa para proteger la privacidad del usuario y perdería las características de privacidad implementadas y recomendadas por defecto en "Tor Browser".

3.4. Otras redes oscuras

Aunque Tor es la red oscura más conocida y con mayor base de usuarios, existen múltiples alternativas, aunque con distintas finalidades que no siempre son la privacidad del usuario [14].

I2P[2]: Es una red basada en el enrutamiento a través de túneles entrantes y salientes, similar a Tor. Mejora la compatibilidad sobre múltiples protocolos y herramientas que trabajan sobre Internet. El envío y recepción de datos se realiza mediante caminos diferentes. Los mensajes enviados son divididos en múltiples paquetes y caminos. Mientras mayor sea el número de usuarios, mayor es la privacidad y la seguridad de la propia red puesto que se necesita interceptar dos nodos completos para tomar el control de una comunicación. Es más lenta que Tor y tiene menor número de usuarios.

ZeroNet[3]: Red descentralizada que principalmente se preocupa por la censura en la red. Es una red entre pares (P2P), cada usuario conectado se convierte a su vez en un servidor de las páginas que visita. Soporta contenido dinámico y todo aquello que no es frecuentemente utilizado va expirando, de modo que deja de ser accesible con el paso del tiempo. Por defecto no es anónima, sino que depende de Tor para anonimizar la IP del usuario.

Freenet[4]: Al igual que ZeroNet, el objetivo que establece es eliminar la posibilidad de censura por parte de gobiernos, grupos o individuos. Funciona mediante una red de pares (P2P) sobre la que se distribuyen los diferentes archivos publicados de manera cifrada. El usuario no conoce el contenido que almacena en su equipo y dicho contenido no puede ser borrado de la red. Suele ser utilizado como almacenamiento en la nube para contenido estático. Tampoco garantiza la privacidad del usuario.

[2] Red anónima I2P. (2022). Recuperado de https://geti2p.net/es/
[3] ZeroNet: Decentralizez websites. (2022). Recuperado de https://zeronet.io/es
[4] Freenet. (2022). Recuperado de https://freenetproject.org/

La red oscura que mejor se adapta a las necesidades del proyecto, tanto por volumen de datos como por finalidad (privacidad) es Tor, por lo que será esta sobre la que trabajaremos.

3.5. Búsqueda de foros de interés

Encontrar sitios de interés en Tor es un proceso tedioso y lento, puesto que al no existir una indexación de sitios ni de su contenido, suele ser un proceso manual que se basa en conocer grupos de individuos que tengan dicha información o buscar referencias dentro de Internet o “Clearnet”.

Una vez empezamos a buscar, observamos que las URL dentro de Tor tienen un formato de entre 16 a 56 caracteres aleatorios difícilmente memorizables seguidos de la extensión “.onion”. Estos enlaces únicamente pueden ser accedidos desde un navegador Tor. Estos dominios no están regulados por ninguna entidad y su propietario es quien posea la clave privada asociada al mismo.

La generación de estas direcciones reguladas recae en un mecanismo de cifrado asimétrico llamado RSA [15]. Mediante RSA se generan claves compuestas de dos partes, una privada (no compartida) que poseería el propietario del dominio y otra parte pública que sería a partir de la cual se generaría el dominio mediante una serie de operaciones.

Las direcciones “.onion” son generadas aplicando (en minúsculas) una codificación *base32* [16] a los primeros 10 bytes del resumen *SHA1* [17] aplicado a sobre un fichero de clave pública en formato *DER* [18].

Existen técnicas[5] mediante las cuales los operadores de sitios web en Tor, mediante fuerza bruta, generan dominios secuencialmente hasta encontrar uno cuyos primeros caracteres coinciden con una palabra que les represente. No obstante, este mecanismo es computacionalmente costoso y no suele superar los 6 caracteres reconocibles. Un ejemplo sería:

```
tortaxi7axhn2fv4j475a6blv7vwjtpieokolfnojwvkhsnj7sgctkqd.onion
```

[5] mkp224o. (2022). Recuperado de https://github.com/cathugger/mkp224o

Existen páginas que contienen múltiples enlaces a los diferentes sitios más conocidos y visitados dentro de la web oscura.

The Hidden Wiki: Uno de los sitios más conocidos para encontrar enlaces ".onion". Realmente es un conjunto de wikis independientes, cada una con una URL, que muestran enlaces y el estado de los mismos.

tor.taxi: Muestra múltiples enlaces agrupados por tipo de sitio, desde sitios de noticias hasta mercados ilegales. Muestra el estado de los enlaces y tiene una interfaz simple.

dark.fail: Lista de sitios frecuentemente visitados, tanto de contenido legal como ilegal. Muestra el estado de los mismos.

3.6. Foros candidatos

Los sitios que se muestran a continuación son una selección basada en el tipo de web buscada, descartando todos los que no se ajustan a la tipología de foro como, por ejemplo, sitios para la realización de ventas o *markets*, y en base al número de referencias de los mismos en los diferentes agregadores de enlaces analizados anteriormente.

Dread es un popular foro de estructura similar a Reddit pero en la web oscura y con unas reglas de censura más laxas.

The Majestic Garden es un foro de la web oscura que se centra en la reducción del dolor mediante el uso de sustancias psicodélicas de forma segura. Tiene unas reglas estrictas sobre temáticas moralmente censurables como el fraude, asesinato o pornografía infantil.

CryptBB es un foro cuya temática principal gira en torno a la ciberseguridad y hacking.

The Hub es un foro de discusión centrado en el mercado de la web oscura, las criptomonedas y ciberseguridad.

Suprbay es un foro de la web oscura basado principalmente en la piratería de contenido digital. Es el foro oficial del conocido sitio The Pirate Bay.

3.7. Detección de autoría

La detección de autoría, tal y como se describió anteriormente, permite inferir características del autor de un documento en base al estudio de las diferentes características existentes en el propio documento.

Según el estudio del estado del arte sobre la detección de autoría, se han encontrado una serie de características de interés que frecuentemente son utilizadas para resolver este tipo de tarea:

Basado en el léxico [3] [19]: Mediante el análisis léxico se estudian las palabras que aparecen en los textos y su frecuencia de aparición.

Basado en el estilo [3] [20]: Este tipo de análisis se centra en el modo de escribir del autor, centrándose en errores gramaticales, tipográficos, emojis, el número de frases o palabras, la longitud de las palabras utilizadas, uso de la puntuación, etc.

Basado en el tiempo [3] [21]: Análisis mediante el cual se incorpora al texto su fecha de redacción o publicación, lo que permite analizar aspectos como la posible ubicación geográfica, su región o incluso sus horarios de actividad.

Para, a partir de las características anteriormente enumeradas, realizar un proceso de clasificación de mensajes por autor, se aborda desde un enfoque de *aprendizaje supervisado* [22]. Esta es una técnica que nos permite deducir una función a partir de un conjunto de datos de entrenamiento. Su objetivo es predecir el valor de salida que se obtendrá dado cualquier objeto de entrada válido. Para conseguirlo debe ser entrenado viendo una serie de ejemplos. Esta técnica nos permitirá clasificar con cierto grado de confianza la relación entre un texto y el autor al que puede pertenecer.

Los árboles de decisión [23] son tipos de algoritmos de aprendizaje supervisado principalmente utilizados en clasificación (en caso de variables categóricas) [24] y regresión (en caso de variables numéricas). Tiene una estructura de árbol jerárquico,

que consta de un nodo raíz y nodos hoja. Los nodos hoja representan todos los resultados posibles dentro del conjunto de datos dado.

Random Forest [23] es una combinación de árboles de decisión en la que cada árbol vota por asociar o clasificar un objeto con cierta clase y posteriormente se promedian los votos de todos los árboles (bosque) para dictar el resultado final. Este método permite construir un modelo robusto en base a un grupo de modelos más débiles [25].

XGBoost [26] es otro método basado en árboles de decisión que obtiene mejores resultados [27] que *Random Forest*. Hace uso de la técnica de aumento de gradiente, mediante la cual se ajustan diferentes pesos dentro de los árboles en busca de una mejora en la precisión y en la velocidad.

4. DISEÑO

En esta sección se muestra el modo de abordar el problema a resolver, partiendo de un esquema general y detallando de manera progresiva los diferentes posibles planteamientos y tecnologías escogidos.

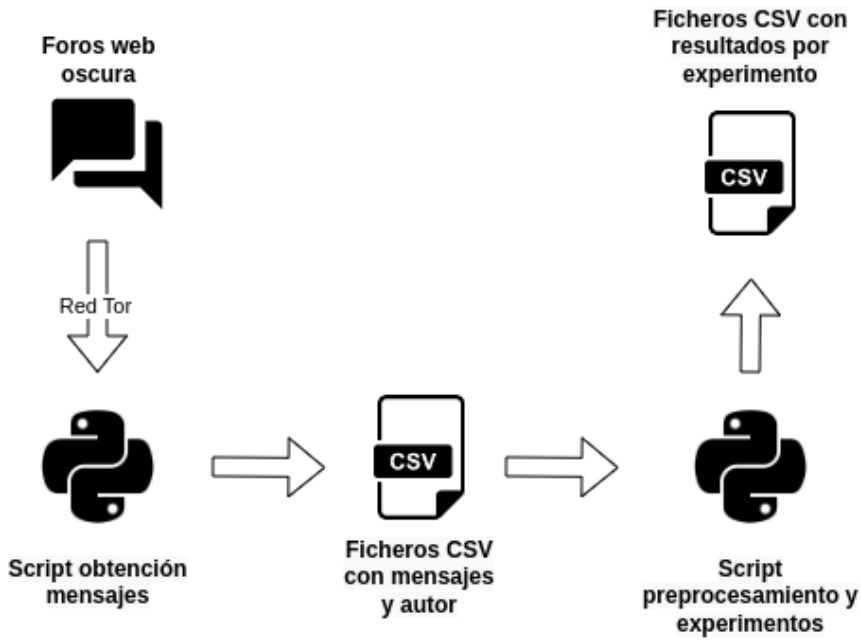


Ilustración 5. Arquitectura general del sistema

En la ilustración anterior se muestra la arquitectura general del sistema planteado, en la cual se identifican los diferentes elementos que intervienen en en el proceso:

Los foros de la web oscura son el origen de la información que permitirá realizar experimentos posteriormente sobre ellos. Para acceder a los mismos, se necesitará un mecanismo para poder acceder a los mismos desde un programa automático a través de Tor.

Una vez obtenido el acceso, se necesitará un programa que recorra los diferentes foros obteniendo, descargando y almacenando los mensajes de manera estructurada.

Los ficheros de datos obtenidos se almacenarán en formato adecuado para su manipulación posterior.

Cuando los datos se encuentren almacenados en el equipo, se les deberá dar un tratamiento adaptado a los objetivos y experimentos que se desean cubrir. Este proceso se llama preprocesamiento.

El programa encargado de realizar los experimentos obtendrá los datos desde el disco, aplicará el preprocesamiento y posteriormente ejecutará los diferentes experimentos, almacenando de vuelta los resultados en el disco en formato legible para su interpretación.

En la siguiente ilustración se muestra a modo resumen la pila tecnológica con la que se pretende realizar el proyecto, siendo cada una de estas tecnologías descritas en siguientes secciones.

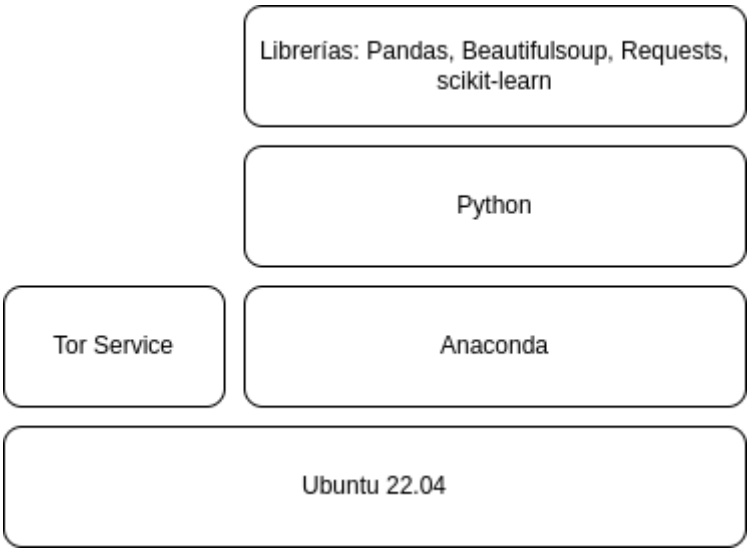


Ilustración 6. Pila de tecnologías

4.1. Acceso a los foros

Durante la fase de análisis se han encontrado una serie de foros candidatos desde los cuales se tratará de obtener el mayor número de mensajes posibles para su posterior estudio.

Dread: Durante el tiempo que se ha desarrollado este proyecto no se ha podido acceder debido a problemas técnicos del foro. Según se puede leer en otros foros de la web oscura, se debe a un ataque de denegación de servicio con origen desconocido.

The Majestic Garden: Para acceder a este foro se requiere un correo de la red oscura y un par de claves PGP de 4096 bits. Tras el proceso de registro, existe una validación manual por parte de los administradores.

CryptBB: El registro se divide en tres fases: un registro estándar con usuario contraseña, un desafío a completar para validar unos conocimientos mínimos de la temática y posteriormente una validación manual por parte de los administradores.

The Hub: Dispone de un método de registro estándar con usuario y contraseña seguido de una validación manual por parte de los administradores.

Suprbay: Dispone de una gran cantidad de mensajes sin necesidad de registro.

4.2. Automatizar el acceso a los foros

Una vez realizado y validado el proceso de registro, se dispone de un acceso a través del navegador Tor. No obstante, para automatizar la obtención de mensajes, se requiere que un sistema automatizado sea capaz de acceder a dichos foros con las credenciales correspondientes y descargue todos los mensajes disponibles. Este sistema se construirá mediante una serie de scripts programados en lenguaje *Python*[6] dadas las facilidades que ofrece el lenguaje a la hora de leer, manipular y guardar datos.

Python se desplegará sobre el sistema operativo mediante *Anaconda*[7], una plataforma de distribución de Python especializada en la ciencia de datos que permite el aislamiento del entorno y su replicación de forma rápida.

Para poder lanzar peticiones a través de la red Tor desde un script de Python se requiere de la intervención de una herramienta adicional que lo permita. La solución más adecuada para ello es el uso de un proxy que reciba las peticiones del script y se encargue de enviarlas a través del servicio Tor. Existen varias alternativas disponibles para Python y Tor:

torify[8]: Utilidad que permite lanzar comandos por consola a través de la red Tor.

torrequest[9]: Biblioteca simple que permite hacer peticiones HTTP y HTTPS mediante Tor. Tiene algunos errores de implementación y no parece estar mantenido por el autor.

privoxy[10]: Proxy web que modifica los datos y cabeceras de las peticiones web con el objetivo de mejorar la privacidad.

SOCKS [28] y ***SOCKS5***: Proxy sencillo que permite reenviar conexiones TCP y UDP.

6 https://www.python.org/about/
7 https://www.anaconda.com/
8 https://linux.die.net/man/1/torify
9 https://github.com/erdiaker/torrequest
10 https://www.privoxy.org/

El servicio Tor implementa por defecto un proxy SOCKS y SOCKS5, que únicamente requiere ser habilitado para funcionar. Dado que la interacción a través de la red Tor será para hacer una serie de peticiones a través de HTTP y HTTPS sin muchos requisitos adicionales, utilizaremos este sistema por defecto para solucionar el problema.

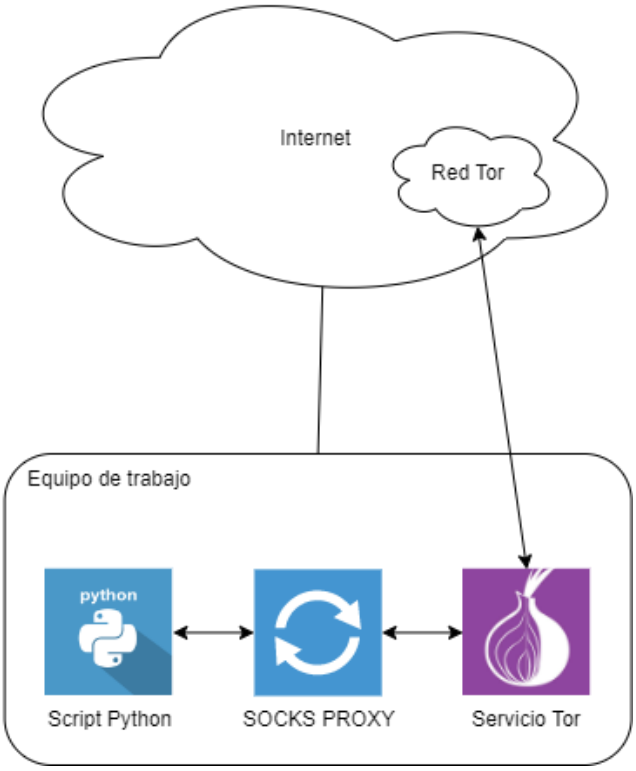


Ilustración 7. Acceso automatizado a Tor mediante SOCKS PROXY

Otro problema es realizar la autenticación de usuario en los foros. Para ello se realizará un inicio de sesión desde el navegador Tor y se obtendrán las cookies de sesión, las cuales se colocarán en el script de Python de la manera adecuada para su uso. De esta manera, el script de Python tendrá permiso para leer los foros de interés.

4.3. Procesamiento y descarga de mensajes

Los foros seleccionados de la web oscura tienen el esquema tradicional de foro que consta de sub-foros, hilos y publicaciones.

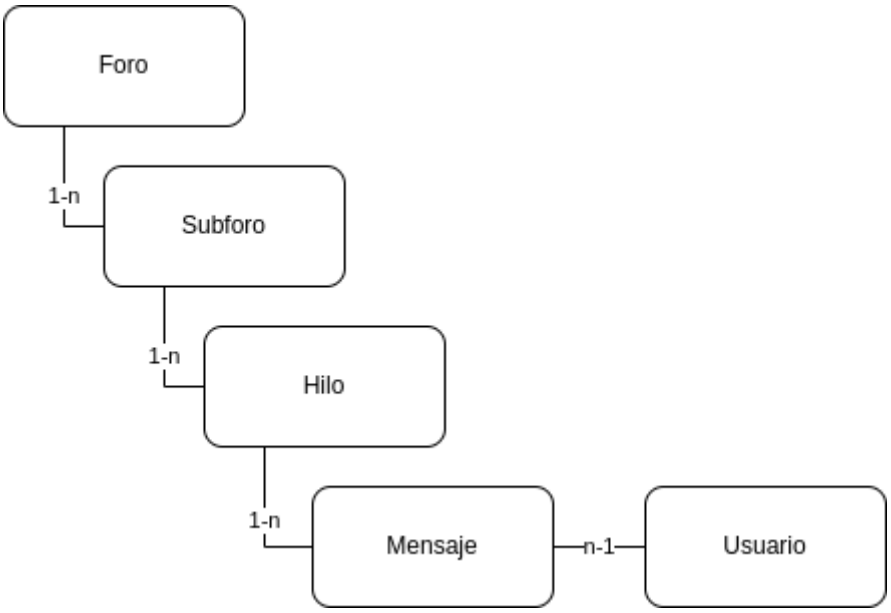


Ilustración 8. Jerarquía de un foro

Para obtener los mensajes y poder trabajar con ellos, se requiere un sistema que realice la siguiente secuencia para cada foro:

Dado un foro, obtener la lista de sub-foros.

Para cada subforo, obtener la lista de páginas que contiene.

Recorrer cada página del subforo en busca de los hilos que contiene.

Para cada hilo, obtener la lista de páginas del hilo.

Obtener cada mensaje para cada página del hilo.

Cada mensaje obtenido debe almacenarse en un fichero de texto CSV.

Dado que se prevé un gran número de peticiones a cada foro, se volcará a disco la información obtenida en cada iteración, de modo que ante un posible bloqueo por parte de los administradores del sitio, se conozca el punto de interrupción del proceso para poder continuarlo en intentos sucesivos.

Las peticiones HTTP se realizarán mediante el uso de la biblioteca ***requests***[11] de Python. Se debe tener en cuenta la asignación de las cabeceras HTTP al inicio del

[11] https://requests.readthedocs.io/en/latest/

script con el objetivo de simular del mejor modo posible el comportamiento de un navegador Tor.

Antes de iniciar el proceso principal del script, se introducirá una comprobación de la IP origen que se encuentre a través de la conexión proxy con el objetivo de comprobar que la conexión con la red Tor es correcta y la IP es anónima antes de continuar la ejecución.

La respuesta de cada petición HTTP servirá como entrada para la biblioteca ***beautifulsoup***[12], la cual permite manipular y buscar fácilmente en el ***DOM*** [29].

Una vez localizados los mensajes de cada hilo, se añadirán a un objeto *dataset* de la librería *pandas*, el cual dispone de un sencillo método con el cual exportar a CSV. Esta exportación se hará con el modo de operación *append* de modo que en cada iteración se añadirán un conjunto de mensajes adicionales al fichero CSV sin borrar el contenido anterior.

4.4. Estructura de directorios

Para cada foro estudiado se creará una carpeta con su nombre. Dentro de la misma, se creará una carpeta por cada subforo del cual se vayan a extraer mensajes. Dentro de la carpeta del subforo, se creará un fichero CSV con los mensajes obtenidos y un fichero de registro (log) que irá registrando el paso a paso de la ejecución.

[12] https://www.crummy.com/software/BeautifulSoup/bs4/doc/

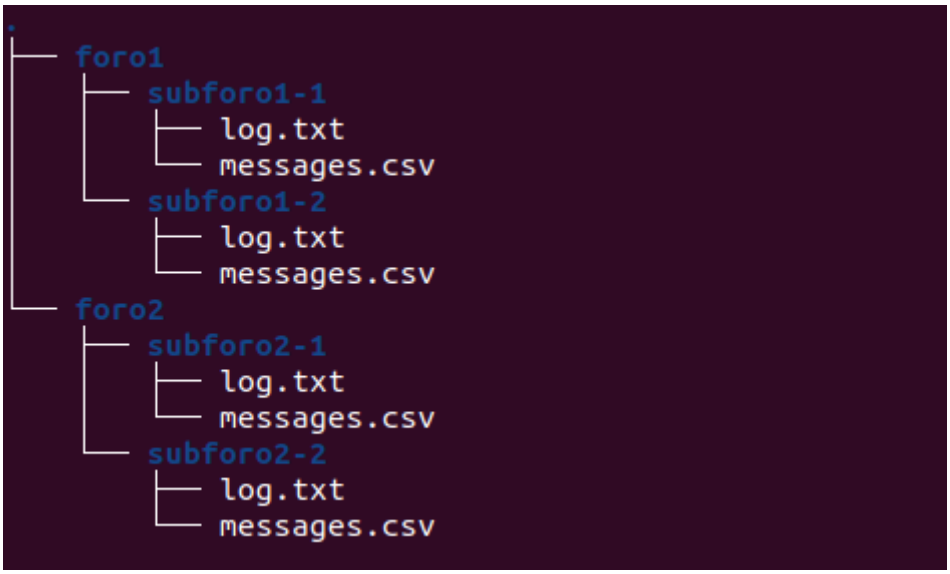


Ilustración 9. Estructura final de archivos en la extracción de mensajes

4.5. Diseño de los experimentos

Antes de los experimentos se realiza un estudio de características basadas en el léxico. Este análisis servirá como aproximación al problema y de base comparativa para otros posibles experimentos posteriores.

La información de la que se dispone es, para cada mensaje, el contenido y el nombre de usuario de su autor. El problema se plantea como una tarea de clasificación [30] la cual puede ser resuelta mediante aprendizaje supervisado, como concretaremos más adelante.

La tarea de clasificación planteada consiste en, dado un mensaje, determinar a qué “etiqueta” o, en este caso, a qué autor pertenece. El algoritmo utilizado se basará en el análisis léxico del texto para evaluarlos y clasificarlos. Nos limitamos, por tanto, a una atribución de autoría basada en el léxico.

4.5.1. Preprocesamiento de los datos obtenidos

Las reglas de preprocesamiento a aplicar se basan en el tipo de experimentos que se realizan posteriormente. En este caso realizaremos experimentos basados en el léxico, es decir, estudiando las palabras empleadas y su frecuencia, por lo tanto, se hará una limpieza de elementos que no formen parte del vocabulario y otros elementos que pueden afectar al conteo, como las URL. En este último caso se sustituirán mediante una expresión regular por la cadena literal “url”.

4.5.2. Realización de experimentos

Para la realización de los experimentos propuestos mediante aprendizaje supervisado se requerirá realizar la división del conjunto de los mensajes en dos partes independientes: una primera parte para entrenar el modelo y otra segunda parte para la evaluación de los resultados obtenidos. Esta división se realizará en una proporción del 70%-30% respectivamente.

Tras realizar la partición será necesario adecuar los textos para que sean un tipo de entrada válida para los algoritmos, esto es, transformarlos en vectores numéricos que los representen. Para realizar esta transformación se utilizará TF-IDF [35] al ser uno de los métodos más populares y utilizados.

Una vez están los vectores preparados, se procede a entrenar y evaluar los modelos a partir de los algoritmos RandomForest [23] y XGBoost [27], ya comentados anteriormente.

A nivel técnico se hará uso de ***scikit-learn***[13], una biblioteca de código abierto para aprendizaje automático que soporta tanto aprendizaje supervisado como no supervisado. Incorpora múltiples utilidades que facilitan las tareas de preprocesamiento, selección de modelos o evaluación, entre otras.

Cada experimento dispondrá de tres variantes según la cantidad de autores a clasificar. Haremos una prueba con los textos de 20, 10 y 2 autores respectivamente. Esto nos permitirá comprobar empíricamente si la cantidad de etiquetas puede ser determinante en este tipo de algoritmos cuando trabajamos con características basadas en el léxico.

5. IMPLEMENTACIÓN

En la fase de implementación se trata de llevar a cabo la solución planteada, en este caso se trata de una serie de experimentos basados en el diseño y tecnologías anteriormente estudiadas. Se obtendrán los datos, se procesarán y generarán los resultados.

[13] https://scikit-learn.org/

5.1. Preparación del entorno de trabajo

La documentación acerca de la preparación del entorno de trabajo queda plasmada en el Anexo I de este mismo documento, donde se detalla cómo instalar una infraestructura de Python con los módulos y dependencias requeridos.

5.2. Acceso a los foros

Los foros de la web oscura, por lo general, requieren que estemos registrados para poder acceder a la información que contienen. Los sistemas de registro suelen ser convencionales, si bien muchas veces requieren una última validación por parte del administrador del sitio, lo cual puede dilatarse en el tiempo e incluso nunca llegar a producirse.

Dread: Se ha descartado por los problemas técnicos que presenta los últimos meses, lo que lo hace completamente inaccesible.

The Majestic Garden: Se ha generado una clave PGP de acuerdo a los requerimientos y se ha seguido el proceso de registro. No obstante, el proceso de validación manual por parte del administrador no ha resultado favorable, por lo que se descarta al no haber conseguido acceso.

CryptBB: Se ha realizado una fase de registro mediante usuario y contraseña. Posteriormente, se han planteado una serie de desafíos técnicos sencillos que requieren unos conocimientos mínimos de programación y conceptos relacionados con la ciberseguridad como las funciones resumen. Una vez superados los desafíos la cuenta ha sido aprobada por la administración del sitio y se ha obtenido acceso al mismo.

The Hub: Se ha realizado una fase de registro mediante usuario y contraseña, pero nunca se ha pasado la validación por parte de la administración del sitio.

Suprbay: Dada la cantidad de mensajes publicados en los subforos públicos de este, se ha obviado el proceso de registro.

5.3. Obtención del conjunto de mensajes

Tanto *CryptBB* como Suprbay se basan en un sistema de gestión de foros open source llamado *MyBB*[14]. Esto permite que gran parte de los scripts desarrollados para la obtención de mensajes de los mismos puedan ser reutilizables en ambos.

5.4. Análisis del conjunto de mensajes obtenido

Los datos han sido obtenidos de tres orígenes diferentes. También se ha construido un cuarto conjunto de datos mediante la combinación de los otros tres, quedando tal y como se muestra en la siguiente tabla.

Conjunto	# Mensajes	Longitud media mensajes (caracteres)	Desviación típica longitud mensajes	Autores únicos
CryptBB A	2.595	347,39	1.108,21	880
CryptBB B	3.461	399,27	768,16	1.040
Suprbay	16.256	613,68	1.288,73	1.881
Total	22.312	549,45	1.206,47	3.500

Tabla 6. Resumen del conjunto de datos obtenido

En base a la tabla anterior podemos extraer la siguiente información:

El conjunto de datos con mayor número de mensajes es a su vez el de mayor longitud media en los mismos.

La longitud del mensaje tiene una gran dispersión en todos los conjuntos de datos obtenidos.

La suma total de usuarios únicos encontrada en cada uno de los conjuntos es 3.801 (880+1.040+1.881). La diferencia entre este valor y los 3.500 usuarios únicos encontrados en el conjunto total se debe a que hay 301 usuarios que se encuentran en más de un conjunto de datos simultáneamente.

[14] https://mybb.com/

5.5. Preprocesamiento del conjunto de datos

Con el objetivo de aprovechar la mayor cantidad de los mensajes obtenidos, continuaremos trabajando con el conjunto total de los datos para el resto de los experimentos realizados. Se mostrará el proceso de limpieza llevado a cabo mediante la siguiente muestra:

```
'\n\nhttps://pirates-forum.org/showthread.php?tid=734\n\nYou know you wanna\r\n\t'
```

Ilustración 10. Mensaje original

En primer lugar, se ha llevado a cabo la limpieza de caracteres especiales como el retorno de carro y la tabulación, pues aparecen en múltiples ocasiones. Quedaría del siguiente modo:

```
'  https://pirates-forum.org/showthread.php?tid=734  You know you wanna   '
```

Ilustración 11. Limpieza inicial

Posteriormente, se reemplazan las URL por la cadena “URL”.

```
'  URL  You know you wanna   '
```

Ilustración 12. Reemplazo de URL

Como último paso, se ha pasado todo a minúsculas y se han eliminado espacios innecesarios.

```
'url  you know you wanna'
```

Ilustración 13. Preprocesado final del mensaje

5.6. Experimentación

Para poder validar la precisión de los experimentos propuestos se escogen métodos de aprendizaje supervisado, los cuales requieren la división del conjunto de datos en dos partes: una para realizar el entrenamiento (train) y otra para validar los resultados (test). Para dicha partición se ha utilizado una proporción del 70%-30% respectivamente.

Para poder trabajar con textos en aprendizaje supervisado es necesario transformarlos en vectores numéricos que permitan entrenar los modelos. Se ha utilizado TF-IDF para realizar este proceso. Durante este proceso se ha realizado la

supresión de “stopwords” con el objetivo de reducir la cantidad de palabras que no aportan información ni contexto al conjunto de datos.

Se han utilizado dos modelos diferentes basados en aprendizaje supervisado: RandomForest y XGBoost. Para cada uno de ellos se han realizado dos experimentos, variando entre ellos el número de usuarios a clasificar según el número total de mensajes escritos. Además, se ha realizado un tercer experimento (basado en el mejor resultado obtenido de los dos anteriores) suprimiendo los mensajes de una extensión inferior a 300 caracteres.

6. RESULTADOS

Una matriz de confusión es una herramienta que permite la evaluación de un problema de aprendizaje supervisado.

	Predicción Positivo	**Predicción Negativo**
Positivo	Verdadero Positivo (TP)	Falso Negativo (FN)
Negativo	Falso Positivo (FP)	Verdadero Negativo (TN)

Tabla 7. Matriz de confusión

A partir de la anterior tabla de confusión y con el objetivo de poder interpretar correctamente los resultados, se define un conjunto de métricas:

Precisión: Intuitivamente, se define como la habilidad del clasificador de no etiquetar una muestra negativa como positiva. Formalmente se define como:

$P = tp / (tp + fp)$

Cobertura (*recall)* [30]: Se interpreta como la habilidad del clasificador para encontrar todas las muestras positivas.

$R = tp / (tp + fn)$

“f1-score”: Se interpreta como la media armónica de los valores de precisión y cobertura, siendo 1 la mejor puntuación y 0 la peor.

$F1 = 2PR / (P+R)$

Support: Número de ocurrencias de cada clase en fase de prueba.

6.1. Tablas de resultados

A continuación se muestran los resultados para cada uno de los experimentos planteados: con los 20, 10 y 2 usuarios con mayor número de mensajes respectivamente.

6.1.1. RandomForest para los 20 usuarios con más mensajes

Usuario	precision	recall	f1-score	support
0	0,349	0,788	0,483	396
1	0,369	0,336	0,352	217
2	0,445	0,394	0,418	165
3	0,313	0,238	0,271	151
4	0,894	0,977	0,933	129
5	0,975	0,975	0,975	122
6	0,690	0,492	0,574	118
7	0,347	0,234	0,279	107
8	0,476	0,303	0,370	99
9	0,525	0,429	0,472	98
10	0,661	0,446	0,532	83
11	0,348	0,178	0,235	90
12	0,549	0,368	0,441	76
13	0,196	0,118	0,148	76

14	0,189	0,082	0,115	85
15	0,355	0,134	0,195	82
16	0,175	0,096	0,124	73
17	0,296	0,145	0,195	55
18	0,184	0,156	0,169	45
19	0,897	0,583	0,707	60
accuracy	0,452	0,452	0,452	0,452
macro avg	0,462	0,374	0,399	2327
weighted avg	0,460	0,452	0,431	2327

Tabla 8. Resultados RandomForest 20 usuarios

6.1.2. RandomForest para los 10 usuarios con más mensajes

Usuario	precision	recall	f1-score	support
0	0,516	0,854	0,644	404
1	0,444	0,398	0,420	191
2	0,637	0,431	0,514	167
3	0,407	0,236	0,299	140
4	0,969	0,981	0,975	161
5	0,991	0,983	0,987	116
6	0,769	0,526	0,625	114
7	0,485	0,323	0,388	99
8	0,639	0,355	0,456	110
9	0,636	0,538	0,583	91
accuracy	0,614	0,614	0,614	0,614

macro avg	0,650	0,563	0,589	1593
weighted avg	0,623	0,614	0,598	1593

Tabla 9. Resultados RandomForest 10 usuarios

6.1.3. RandomForest para los 2 usuarios con más mensajes

Usuario	precision	recall	f1-score	support
0	0,778	0,872	0,822	390
1	0,688	0,531	0,599	207
accuracy	0,754	0,754	0,754	0,754
macro avg	0,733	0,702	0,711	597
weighted avg	0,747	0,754	0,745	597

Tabla 10. Resultados RandomForest 2 usuarios

6.1.4. XGBoost para los 20 usuarios con más mensajes

Usuario	precision	recall	f1-score	support
0	0,311	0,725	0,435	414
1	0,214	0,233	0,223	189
2	0,270	0,2	0,230	155
3	0,228	0,154	0,184	149
4	0,844	0,928	0,884	152
5	0,938	0,924	0,931	132
6	0,5	0,252	0,335	115
7	0,205	0,149	0,172	101
8	0,211	0,037	0,063	107
9	0,396	0,494	0,44	89
10	0,514	0,207	0,295	87
11	0,311	0,192	0,237	73
12	0,135	0,067	0,089	75
13	0,136	0,118	0,126	68
14	0,071	0,014	0,024	71
15	0,120	0,189	0,147	74
16	0	0	0	87

17	0,083	0,014	0,024	72
18	0,071	0,018	0,028	57
19	0,676	0,383	0,489	60
accuracy	0,360	0,360	0,360	0,360
macro avg	0,312	0,265	0,268	2327
weighted avg	0,337	0,360	0,321	2327

Tabla 11. Resultados XGBoost 20 usuarios

6.1.5. XGBoost para los 10 usuarios con más mensajes

Usuario	precision	recall	f1-score	support
0	0,389	0,821	0,528	379,000
1	0,323	0,260	0,288	200,000
2	0,392	0,309	0,346	152,000
3	0,263	0,101	0,146	148,000
4	0,910	0,893	0,901	169,000
5	0,947	0,926	0,936	135,000
6	0,600	0,255	0,358	106,000
7	0,410	0,127	0,194	126,000
8	0,211	0,048	0,078	83,000
9	0,667	0,379	0,483	95,000
accuracy	0,492	0,492	0,492	0,492
macro avg	0,511	0,412	0,426	1593,000
weighted avg	0,495	0,492	0,455	1593,000

Tabla 12. Resultados XGBoost 20 usuarios

6.1.6. XGBoost para los 2 usuarios con más mensajes

Usuario	precision	recall	f1-score	support
0	0,688	0,984	0,810	380,000
1	0,887	0,217	0,348	217,000
accuracy	0,705	0,705	0,705	0,705

macro avg	0,787	0,600	0,579	597,000
weighted avg	0,760	0,705	0,642	597,000

Tabla 13. Resultados XGBoost 2 usuarios

6.1.7. Top 20 palabras de más peso en cada experimento

RF-20	RF-10	RF-2	XGB-20	XGB-10	XGB-2
gmt	gmt	problem	url	gmt	wrote
0000	0000	maybe	gmt	info	maybe
torrentfreak	boogers	edit	don	originally	old
doesnt	doesnt	small	sharing	quote	real
xprogrammer	specially	control	just	edit	edit
designed	nihilism	ask	people	url	interesting
mp3	2020	old	like	torrent	people
edit	addresses	wrote	source	people	poor
proprietary	torrentfreak	proper	piracy	forum	wants
boogers	wuflu	client	good	just	current
gnu	source	gnu	edit	don	wasn
health	blah	apps	torrent	internet	health
wuflu	interestingly	looks	tpb	hope	definitely
2017	edit	really	think	like	2018
permit	dht	pc	time	mod	just
2019	corporate	real	know	account	sure
useless	gnu	linux	xprogrammer	ll	looks
completely	2018	check	want	register	control
debian	register	aren	thread	want	client
fewer	swarm	mention	need	tpb	think

Tabla 14. Palabras de mayor peso en cada experimento

6.2. Análisis de resultados obtenidos

Como análisis general se observa que, a medida que se reduce el número de autores que se requiere clasificar, aumenta la probabilidad de detectar la autoría de un texto correctamente. Además, se observa cómo Random Forest obtiene mejores

resultados que XGBoost en promedio para el conjunto de datos generado y el conjunto de características escogido.

Llama la atención que los usuarios identificados como 4 y 5 obtienen un excelente porcentaje de detección, lo cual hizo pensar en un primer momento que se trataba de usuarios que escribían mensajes repetitivos en múltiples ocasiones, también conocidos como “spammers”. Tras un análisis de los mismos, se observa que realmente sus textos son variados y especialmente extensos, lo que nos hace descartar la idea.

Por otro lado, se han revisado los mensajes de los usuarios con menor tasa de acierto en busca de una posible explicación evidente a golpe de vista, no obstante, los mensajes de dichos usuarios son aparentemente normales, aunque relativamente cortos.

7. CONCLUSIONES

Según todo lo anterior, podemos concluir que existen múltiples enfoques que permiten relacionar una publicación de un foro de la web oscura con el potencial usuario que lo escribe. Según los estudios referenciados [3] [4] [20] [21], existen múltiples características adicionales que permitirían mejorar el valor de confianza obtenido a la hora de asociar un texto a un usuario, debido a esto se plantean una serie de posibles trabajos futuros.

Se han probado algunas técnicas centradas en el lenguaje, concretamente las basadas en el estudio léxico. Se han estudiado otras como son las basadas en la *estilometría*. También se han encontrado otras interesantes propuestas para obtener información adicional basándonos en el tiempo de publicación.

Se han estudiado, diseñado e implementado una serie de experimentos basados en aprendizaje supervisado con el objetivo de clasificar textos según sus autores, descubriendo además otros estudios [3] que revelan la posibilidad de mejorar los resultados obtenidos mediante otras estrategias y técnicas aplicadas como el aprendizaje profundo y modelos del lenguaje.

Se ha descubierto y desmitificado una gran herramienta como es la web oscura, y más concretamente Tor. Habitualmente se le atribuye un aspecto oscuro y malicioso, dedicado únicamente para gente que pretende hacer el mal. No obstante, una vez conocida la tecnología se descubre un punto de vista en el que no todo lo que se publica en esos lugares es malicioso o pretende dañar al prójimo.

Hay personas que por el lugar donde nacen o por otras circunstancias tienen menos posibilidades para el acceso a la información, o que directamente no pueden dar su opinión libremente ni virtual ni físicamente. Gracias a las herramientas mencionadas anteriormente, concretamente el anonimato y la privacidad que aportan, estas personas obtienen libertad en el acceso a la información y la posibilidad de opinar libremente.

Desde el punto de vista de la ciberseguridad y la investigación del cibercrimen, el conjunto de técnicas, herramientas y métodos estudiados y referenciados en este proyecto pueden ser de gran ayuda para poder identificar y relacionar usuarios que realicen actividades ilícitas a través de la red. Podrían ser un elemento más a tener en cuenta durante un proceso de *fingerprinting* o de búsqueda de información en fuentes abiertas, también conocido como *OSINT* [31].

7.1. Trabajos futuros

Si la escritura se convirtiese en un rasgo universalmente aceptado y utilizado para la detección de autoría, podríamos encontrarnos estudios que buscasen todo lo contrario: buscar el modo de escribir más adecuado para “ser invisibles” según nuestro modo de escribir. Existen estudios que ya analizan el enfoque del engaño en la autoría [32].

Sería interesante estudiar los mensajes añadiendo una nueva característica basada en el día, hora y fecha de la publicación, si el usuario escribió entre semana o en fin de semana, si escribe por el día o por la noche, lo cual nos daría una representación de los hábitos del usuario y se añadiría a lo anteriormente estudiado.

Se ha visto que existen foros que disponen de claves PGP asociadas para cada usuario. Se podría tratar de buscar usuarios en foros distintos de la web oscura que tuviesen la misma clave PGP. Con esta nueva información sería posible, por ejemplo,

establecer nuevas premisas partiendo de un conocimiento mayor sobre usuarios que están en dos o más foros con un diferente nombre de usuario.

Otro tema interesante descubierto y no tratado en este proyecto ha sido el uso de algoritmos de aprendizaje profundo y modelos de lenguaje, que también podrían ser aplicados para la detección de autoría en la web oscura [33]. También se podría realizar una combinación de técnicas de clustering con clasificación.

REFERENCIAS

[1] Cooper, A., Tschofenig, H., Aboba, B., Peterson, J., Morris, J., Hansen, M., & Smith, R. (2013). Privacy considerations for internet protocols (No. rfc6973).

[2] Aparici, R., & Osuna Acedo, S. (2013). La cultura de la participación.

[3] Sennewald, B., Herpers, R., Hülsmann, M., & Kent, K. B. (2020, November). Voting for authorship attribution applied to dark web data. In Proceedings of the 30th Annual International Conference on Computer Science and Software Engineering (pp. 217-226).

[4] Juola, P. (2008). Authorship attribution. Foundations and Trends® in Information Retrieval, 1(3), 233-334.

[5] Real Decreto 1125/2003, de 5 de septiembre, por el que se establece el sistema europeo de créditos y el sistema de calificaciones en las titulaciones universitarias de carácter oficial y validez en todo el territorio nacional. Boletín Oficial del Estado, 224, de 18 de septiembre de 2003. https://www.boe.es/eli/es/rd/2003/09/05/1125

[6] Tabla de amortización simplificada. (2022). Agencia Tributaria. Recuperado de https://sede.agenciatributaria.gob.es/Sede/ayuda/manuales-videos-folletos/manuales-practicos/folleto-actividades-economicas/3-impuesto-sobre-renta-personas-fisicas/3_5-estimacion-directa-simplificada/3_5_4-tabla-amortizacion-simplificada.html el 10/10/2022.

[7] Ramadhani, E. (2018, March). Anonymity communication VPN and Tor: a comparative study. In Journal of Physics: Conference Series (Vol. 983, No. 1, p. 012060). IOP Publishing.

[8] Postel, J. (1981). Internet protocol (No. rfc791).

[9] Berners-Lee, T., Cailliau, R., Luotonen, A., Nielsen, H. F., & Secret, A. (1994). The world-wide web. Communications of the ACM, 37(8), 76-82.

[10] Tor Project History. (2022). Recuperado de https://www.torproject.org/about/history/ el 15/10/2022.

[11] Tor Project FAQ. (2019) Recuperado de https://2019.www.torproject.org/docs/faq.html.en#CanExitNodesEavesdrop el 15/11/2022.

[12] This is node joke. Tor battles to fend off swarm of Bitcoin-stealing evil exit relays making up about 25% of outgoing capacity at its height. (2020). Recuperado de https://www.theregister.com/2020/08/12/tor_exit_nodes/ el 7/11/2022.

[13] Over 25% Of Tor Exit Relays Spied On Users' Dark Web Activities. (2021). Recuperado de https://thehackernews.com/2021/05/over-25-of-tor-exit-relays-are-spying.html el 7/11/2022.

[14] Gehl, R. W. (2018). Weaving the dark web: legitimacy on freenet, Tor, and I2P. MIT Press.

[15] Rivest, R. L., Shamir, A., & Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM, 21(2), 120-126.

[16] Base32. (2019). En Wikipedia. Recuperado de https://es.wikipedia.org/wiki/Base32 el 19/09/2022.

[17] Secure Hash Algorithm. (2022). En Wikipedia. Recuperado de https://es.wikipedia.org/wiki/Secure_Hash_Algorithm el 19/09/2022.

[18] X.690. (2022). En Wikipedia. Recuperado de https://en.wikipedia.org/wiki/X.690#DER_encoding el 19/09/2022.

[19] Hanks, P. (2013). Lexical analysis: Norms and exploitations. Mit Press.

[20] Bergsma, S., Post, M., & Yarowsky, D. (2012, June). Stylometric analysis of scientific articles. In Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (pp. 327-337).

[21] La Morgia, M., Mei, A., Raponi, S., & Stefa, J. (2018, July). Time-zone geolocation of crowds in the dark web. In 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS) (pp. 445-455). IEEE.

[22] Aprendizaje supervisado. (2022) En Wikipedia. Recuperado de https://es.wikipedia.org/wiki/Aprendizaje_supervisado el 20/10/2022.

[23] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

[24] Safavian, S. R., & Landgrebe, D. (1991). A survey of decision tree classifier methodology. IEEE transactions on systems, man, and cybernetics, 21(3), 660-674.

[25] Árboles de decisión y Random Forest. (2018). Recuperado de https://bookdown.org/content/2031/ensambladores-random-forest-parte-i.html#random-forest el 20/10/2022.

[26] Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., & Chen, K. (2015). Xgboost: extreme gradient boosting. R package version 0.4-2, 1(4), 1-4.

[27] Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).

[28] SOCKS. (2022). En Wikipedia. Recuperado de https://en.wikipedia.org/wiki/SOCKS el 19/09/2022.

[29] Document Object Model. (2022). En Wikipedia. Recuperado de https://es.wikipedia.org/wiki/Document_Object_Model el 20/10/2022.

[30] Osisanwo, F. Y., Akinsola, J. E. T., Awodele, O., Hinmikaiye, J. O., Olakanmi, O., & Akinjobi, J. (2017). Supervised machine learning algorithms: classification and comparison. International Journal of Computer Trends and Technology (IJCTT), 48(3), 128-138.

[31] Tabatabaei, F., & Wells, D. (2016). OSINT in the Context of Cyber-Security. Open source intelligence investigation, 213-231.

[32] Pearl, L., & Steyvers, M. (2012). Detecting authorship deception: a supervised machine learning approach using author writeprints. Literary and linguistic computing, 27(2), 183-196.

[33] Wang, L. Z. (2017). News authorship identification with deep learning.

[34] Eckersley, P. (2010, July). How unique is your web browser?. In International Symposium on Privacy Enhancing Technologies Symposium (pp. 1-18). Springer, Berlin, Heidelberg.

[35] Aizawa, A. (2003). An information-theoretic perspective of tf–idf measures. Information Processing & Management, 39(1), 45-65.

ANEXO I: Preparación del entorno

Durante este proyecto se visitarán foros y se realizarán descargas de contenido de procedencia poco confiable, por lo que se hará todo desde una instalación limpia del sistema operativo Ubuntu 22.04 LTS, ubicado en un disco duro limpio y totalmente separado del sistema principal del equipo.

El nombre de usuario y nombre de equipo utilizados serán aleatorios con el objetivo de no aportar información sobre el autor.

Instalación y configuración de Tor Browser

En primer lugar, se descarga el navegador desde el sitio web oficial. Se recomienda descargar y utilizar la versión en Inglés por privacidad.

```
wget https://www.torproject.org/dist/torbrowser/11.5.7/tor-browser-linux64-11.5.7_en-US.tar.xz
```

Posteriormente, se extrae el archivo de instalación:

```
tar -xvJf tor-browser-linux64-11.5.7_en-US.tar.xz
```

El siguiente paso es instalar y ejecutar el navegador:

```
./start-tor-browser.desktop --register-app
```

Para poder mandar peticiones desde Python a través de Tor con SOCKS y SOCKS5 se requiere modificar el fichero de configuración de Tor.

Para ello, lo primero que haremos será generar una contraseña para autenticar las peticiones que se hacen al proxy.

```
tor –hash-password “<contraseña>”
```

Una vez tenemos el hash de la contraseña, configuramos las siguientes líneas en el archivo de configuración /etc/tor/torrc.

```
ControlPort 9051
HashedControlPassword <hash-contraseña>
```

Preparación del entorno de desarrollo

Para la instalación de *Anaconda* se refiere a la documentación oficial del mismo:

```
https://docs.conda.io/projects/conda/en/latest/user-guide/install/linux.html#install-linux-silent
```

Una vez instalado *Anaconda*, se crea un nuevo entorno de *Python*, con *pip*:

```
conda create -n tfm pip
```

Activamos el entorno de Anaconda recién creado e instalamos las librerías:

```
conda activate tfm
pip install requests beautifulsoup4 pandas scikit-learn
```