ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

PROPUESTA TEÓRICO / PRÁCTICA DEL CICLO DE VIDA DEL GROOMING DESDE EL PUNTO DE VISTA DE UN ATAQUE APT (ADVANCED PERSISTENT THREAT).

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

JÁCOME JIMÉNEZ RUBÉN ANDRÉS

ruben.jacome@epn.edu.ec

DIRECTORA:

TORRES OLMEDO JENNY GABRIELA

jenny.torres@epn.edu.ec

QUITO, ECUADOR

NOVIEMBRE 2019

DECLARACIÓN

Yo, Rubén Andrés Jácome Jiménez, declaro bajo juramento que el trabajo aquí descrito es de mi autoría, y que no ha sido presentado previamente para ningún grado o calificación profesional; además declaro que he consultado las referencias bibliografías que se incluyen en el presente documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Rubén Andrés Jácome Jiménez
AUTOR DEL PROYECTO

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Rubén Andrés Jácome Jiménez, bajo mi supervisión.

Jenny Gabriela Torres Olmedo
DIRECTORA DEL PROYECTO

AGRADECIMIENTO

Mi eterna gratitud y admiración hacia mis tutores Jenny Torres y Patricio Zambrano, así como a las personas que con su sabiduría, predisposición y conocimiento, aportaron significativamente a la realización de este proyecto.

DEDICATORIA

Dedico este esfuerzo a mis padres, Rubén Darío y María del Carmen. Así como también a todas aquellas personas extraordinarias, que con su amor y paciencia han estado presentes en este incesante camino de aprendizaje y descubrimiento, en especial a Anita Gabriela, Milton Javier y Javier Sebastián.

ÍNDICE DE CONTENIDO

RESUMEN

En el campo de la Seguridad de la Información, existen diversas áreas de estudio que se encuentran en desarrollo. La Ingeniería Social es una de ellas, la cual aborda los desafíos multidisciplinarios de la Seguridad Cibernética. Hoy en día, los ataques asociados con la Ingeniería Social son diversos, incluidas las llamadas Amenazas Persistentes Avanzadas (APTs, por sus siglas en inglés). Estas han sido objeto de numerosas investigaciones; sin embargo, los ataques cibernéticos de naturaleza similar al Grooming se han excluido de estos estudios. En la última década, se han realizado varios esfuerzos para comprender la estructura y el enfoque del Grooming desde el campo de las Ciencias de la Computación con el uso de algoritmos de aprendizaje computacional. Sin embargo, estos estudios no se encuentran alineados con el campo de la Seguridad de la Información. En este trabajo, el estudio del Grooming se formaliza como un ataque de Ingeniería Social, contrastando sus etapas o estaciones con los ciclos de vida asociados con las APT. Para lograr este objetivo, se utiliza una base de datos de chats de ciber pedófilos reales; esta información es refinada y se aplica el modelado de tópicos de Asignación de Dirichlet Latente (LDA, por sus siglas en inglés) para determinar el número de etapas del ataque. Posteriormente, se asigna un contexto lingüístico a cada una de estas etapas, y con el uso del aprendizaje automático, se entrena un modelo lineal para obtener el 97,6% de precisión de entrenamiento. Con estos resultados, se determina que el estudio del Grooming puede apoyar la investigación asociada con la Ingeniería Social y contribuir a nuevos campos de estudio para la Seguridad de la Información.

Palabras clave: Seguridad de la información, Ingeniería social, APT, Grooming, LDA.

ABSTRACT

In the field of Information Security, there are several areas of study that are under development. Social Engineering is one of them, which addresses the multidisciplinary challenges of Cyber Security. Nowadays, the attacks associated with Social Engineering are diverse, including the so called Advanced Persistent Threats (APTs). These have been the subject of numerous investigations; however, cybernetic attacks of similar nature as Grooming have been excluded from these studies. In the last decade, various efforts have been made to understand the structure and approach of Grooming from the field of Computer Science with the use of computational learning algorithms. Nevertheless, these studies are not aligned with the Information Security field. In this work, the study of Grooming is formalized as a Social Engineering attack, contrasting its stages or phases with life cycles associated with APTs. To achieve this goal, we use a database of real cyberpedophile chats; this information is refined and the Latent Dirichlet Allocation (LDA) topic modeling is applied to determine the stages of the attack. Once the number of stages is determined, we proceed to give them a linguistic context, and with the use of machine learning, a linear model is trained to obtain 97.6% of training accuracy. With these results, it is determined that the study of Grooming could support research associated with Social Engineering and contribute to new fields of study for Information Security.

Keywords: Information security, Social engineering, APT, Grooming, LDA.

INTRODUCCIÓN

El objetivo de este documento es describir a detalle la contribución técnica brindada al proyecto de investigación doctoral dirigido por el MSc. Patricio Zambrano, en referencia al estudio del Grooming. Los resultados preliminares de la investigación han sido publicados en el artículo científico denominado "*Technical Mapping of the Grooming Anatomy Using Machine Learning Paradigms: An Information Security Approach*" [1], el cual se incluye en el Anexo I.

La investigación contempló tres áreas principales de estudio: el tratamiento de datos, el modelamiento del fenómeno y la aplicación de algoritmos de aprendizaje de máquina para clasificación automática. Cada una de estas áreas o etapas involucró un número determinado de actividades relacionadas y dependientes entre sí. La secuencia de las etapas no fue estricta, ya que durante el desarrollo del proyecto fue imprescindible avanzar y retroceder entre ellas.

El aporte técnico brindado en cada una de las áreas de estudio se describe a continuación:

1. Tratamiento de datos

Con el objetivo de recolectar y pre-procesar el conjunto inicial de datos, se adoptaron los lineamientos del Proceso Estándar de la Industria para Minería de Datos (CRISP – DM, por sus siglas en inglés).

2. Modelamiento del fenómeno

Con el objetivo de modelar computacionalmente el fenómeno del Grooming, en esta sección se analizó e implementó un modelo estadístico para descubrir los tópicos abstractos presentes en el conjunto de datos, utilizando *Matlab*.

3. Aplicación de técnicas de aprendizaje de máquina para clasificación automática

Con el objetivo de validar y evaluar el modelamiento del fenómeno, se implementaron tres diferentes algoritmos de aprendizaje de máquina para realizar clasificación automática de texto, y en base a los resultados obtenidos se determinaron las conclusiones y recomendaciones respectivas para el presente proyecto.

En las secciones siguientes se presentan a detalle las actividades realizadas en cada una de las etapas para lograr los objetivos planteados.

METODOLOGÍA

1. Tratamiento de datos

Perverted Justice (PJ, por sus siglas en inglés) es una organización sin fines de lucro cuyo objetivo principal es combatir el acoso sexual de menores a través de Internet [2]. PJ pone a disposición pública una base de datos, la cual se compone de una extensa recopilación de conversaciones reales de Grooming provenientes de diversos sitios de chat en línea. El objetivo es realizar operaciones anzuelo, en donde voluntarios aparentan ser niños o adolescentes en salas de chat y se involucran en conversaciones con potenciales atacantes pedófilos. La base de datos de PJ se pone a disposición de la comunidad científica y la sociedad en general a través de su portal web.

La investigación requiere de un conjunto de datos inicial, coherente con la naturaleza del problema, estos datos se relacionan con conversaciones de texto provenientes de aplicaciones de mensajería instantánea. Para determinar la cantidad adecuada de registros a ser descargada y analizada, se recurre al estudio del estado del arte y se determina que las investigaciones relacionadas contemplan entre un mínimo de 44 hasta un máximo de 269 registros de conversaciones [3, 4]. En función de este análisis y de las propiedades de la base de datos PJ, se determina el número de 100 registros como valor promedio para construir el conjunto de datos.

Los recursos utilizados durante el desarrollo de esta investigación se presentan en la Tabla 1.

Recursos HW	Descripción
Procesador	Intel Core i5-2320 @3.00 GHz
Memoria instalada (RAM)	10,0 GB
Tipo de sistema	Sistema operativo de 64 bits, procesador x64

Recursos SW	Descripción	Versión
Windows	Sistema operativo desarrollado por Microsoft.	Windows 10 Pro
Perl	Lenguaje de programación.	Perl 5 Versión 28 Sub versión 1 (v5.28.1).
Matlab	Entorno de computación numérica y lenguaje de programación.	R2019a Versión 9.6
Text Analytics Toolbox (Matlab)	Módulo de herramientas para minería de datos y aprendizaje automático.	R2019a Versión 9.6
Excel	Hoja de cálculo y lenguaje de programación VBA.	Excel 2013 Versión 15.0

Bases de Datos	Descripción	Portal Web
Perverted Justice	Repositorio virtual de conversaciones de pedofilia en línea.	http://www.perverted-justice.com

Tabla 1. ***Recursos utilizados la investigación.***

1.1 Recolección de datos

El portal web de PJ publica su base de datos a través de la web, y los registros de conversaciones se encuentran en formato HTML. Los 100 registros determinados fueron descargados manualmente y almacenados en un único directorio para su posterior pre-procesamiento. Es importante mencionar que para la selección de las conversaciones se toma en cuenta la relevancia, explicitud y extensión (número de caracteres por conversación y número promedio de palabras por línea de mensaje) del texto contenido en las mismas.

Las etiquetas HTML proporcionan una guía para examinar los datos, así como también constituyen un distintivo sustancial al momento de limpiar y estandarizar el formato de los mismos. La Figura 1 muestra un fragmento de un registro obtenido desde PJ en formato HTML.

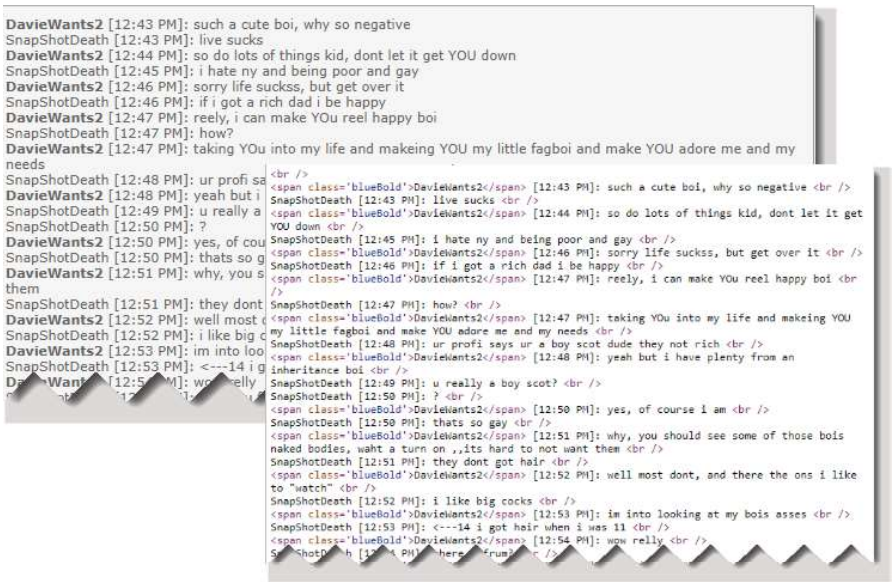


Figura 1. ***Fragmento de conversación desde PJ en formato HTML.***

1.2 Análisis de los datos

Una vez finalizada la recolección y almacenamiento, se examinan las propiedades descriptivas de los registros de datos adquiridos, con el objetivo de obtener una perspectiva general de los mismos. Estos registros contienen una cantidad variable de líneas de mensaje, y a su vez cada línea de mensaje abarca una determinada cantidad de palabras. En la Tabla 2 se describen las propiedades generales identificadas en el conjunto de datos.

Propiedades	Conjunto de Datos
Número total de chats	100
Número total de líneas de mensaje	416777
Número total de palabras	3'334221
Promedio de líneas por chat	4167
Promedio de palabras por chat	34877
Promedio de palabras por línea	8,37

Tabla 2. ***Estadísticas descriptivas del conjunto de datos.***

Se evidencia que las líneas de mensaje en las conversaciones obedecen a una estructura común. En la Figura 2 se expone el fragmento de un registro de conversación, a través del cual se puede apreciar los elementos principales que componen dicha estructura. Estos elementos, en orden de aparición en la línea de mensaje, son: nombre del remitente, marca de tiempo y mensaje de texto, tal como se puede observar en la Tabla 3.

```
trianglelover (6:42:06 PM): hi
kira_kicks_1990 (6:42:11 PM): hello
trianglelover (6:42:15 PM): you a model?
kira_kicks_1990 (6:42:21 PM): omg no lol
trianglelover (6:42:33 PM): oh ok.....cause you look like one
kira_kicks_1990 (6:42:41 PM): wow thats sweet of u to say!
kira_kicks_1990 (6:42:42 PM): asl?
trianglelover (6:42:53 PM): oh I'm an old guy.....Hopewell
```

Figura 2. ***Fragmento de un registro.***

Línea de mensaje	trianglelover	(6:42:15 PM):	you a model?
Estructura	Entidad remitente: **Atacante**	Marca de tiempo	Mensaje de texto

Línea de mensaje	kira_kicks_1990	(6:42:21 PM):	omg no lol
Estructura	Entidad remitente: **Víctima**	Marca de tiempo	Mensaje de texto

Tabla 3. ***Estructura de una línea de mensaje***

Un análisis más detallado de los datos evidenció que los atacantes lideran las conversaciones, eligen el tema de discusión y la mayoría de las veces obligan a las víctimas a responder preguntas no éticas. Las víctimas en la mayoría de los casos simplemente siguen el tema de la conversación con respuestas típicas como "*yes*", "*no*", "*maybe*", "*we will see*", entre otras. Con el objetivo de estudiar efectivamente el fenómeno del Grooming a través de los patrones conductuales de los atacantes, se determina analizar únicamente sus mensajes, excluyendo aquellos escritos por las víctimas, que en este caso corresponden a voluntarios de PJ.

1.3 Preprocesamiento de los datos

En base a los lineamientos de la metodología CRISP – DM [5], la siguiente fase corresponde al pre-procesamiento de datos. En esta etapa, se llevan a cabo varias actividades para construir el conjunto de datos final, es decir, los datos se analizan y se introducen posteriormente en las herramientas de modelado, a partir de los datos sin procesar iniciales. Para ello, se desarrollan tres scripts que permiten automatizar las actividades de pre-procesamiento.

Los dos primeros scripts se desarrollan en Perl con la finalidad de extraer los datos relevantes para la investigación, de todo el conjunto de datos. El tercer script se desarrolla en *Matlab* con el objetivo de limpiar y estandarizar el formato de los datos. A continuación, se detalla cada uno de los scripts:

Script I

Con el objetivo de extraer los mensajes del atacante, es necesario poder distinguirlos claramente de los mensajes de la víctima. Para establecer dicha distinción se hace uso de las etiquetas HTML presentes en los registros recolectados. Este primer script denominado *regexFormatTranscripts.pl*, a través del uso de expresiones regulares permite estandarizar el formato de los datos. El código fuente del script *regexFormatTranscripts.pl* se incluye en el Anexo II, y el mismo permite realizar las siguientes operaciones sobre los datos:

- Eliminar marcas de tiempo.
- Identificar las líneas de mensaje correspondientes al atacante, y diferenciarlas colocando el nombre de usuario entre los símbolos '{' y '}'.
- Identificar las líneas de mensaje correspondientes a la víctima, y diferenciarlas colocando el nombre de usuario entre los símbolos '[' y ']'.

El resultado de la ejecución del script sobre los datos se puede evidenciar en la Figura 3, donde se muestra un fragmento de conversación original y su correspondiente resultado luego de la ejecución del script.

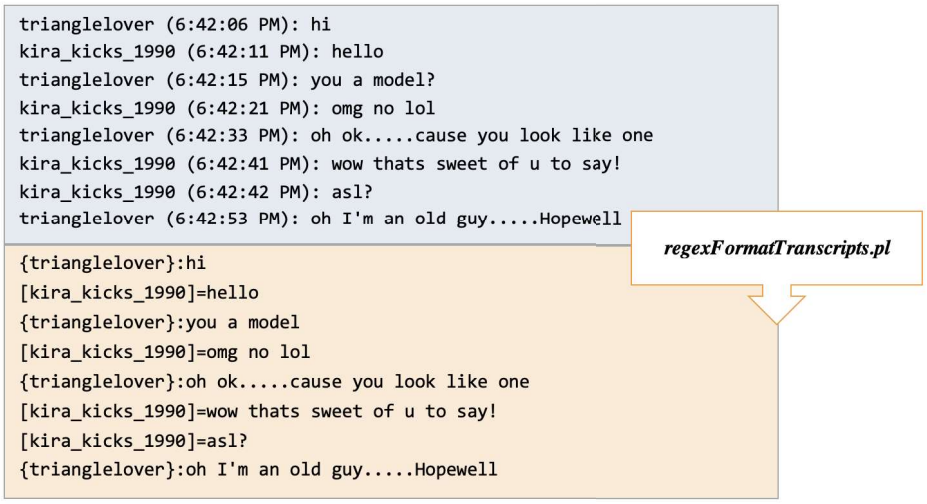


Figura 3. ***Fragmento de conversación antes y después de la ejecución del script I***

Script II

Una vez que se logra distinguir claramente el remitente de los mensajes, es necesario extraer aquellos que son significativos para el caso de estudio. Como se determinó

anteriormente, la información relevante corresponde a los mensajes de los atacantes. Un segundo script denominado *regex_getAttacker.pl* hizo posible extraer los mensajes provenientes del atacante, desde las conversaciones de chat. En el desarrollo de este script se utilizan expresiones regulares y lenguaje de programación *Perl*. El código fuente del script *regex_getAttacker.pl* se incluye en el Anexo III, y el mismo permite realizar las siguientes acciones sobre los datos:

- Extraer los mensajes del atacante.
- Almacenar los mensajes extraídos en una estructura unificada de datos.

El resultado de su ejecución sobre el conjunto de datos se observa en la Figura 4, en la cual se muestra gráficamente la acción del script II sobre el fragmento de ejemplo.

```
hi
you a model
oh ok.....cause you look like one
oh I'm an old guy.....Hopewell
```

Figura 4. ***Fragmento de conversación después de aplicar el script II***

Los datos extraídos se almacenan en una estructura unificada, la cual contiene todos los mensajes de los atacantes. Para ello, se agrupan los datos en un archivo de tipo *CSV* denominado *chatLogs.csv,* el cual se constituye como la estructura base que contiene el conjunto de datos de texto a ser analizados posteriormente. La estructura de datos contenida en *chatLogs.csv* se muestra en la Figura 5. El campo *chat_no* corresponde al número de registro, el campo *message_no* corresponde al número de la línea de mensaje dentro del registro, y el campo *message_text* contiene el texto del mensaje correspondiente.

chatLogs

chat_no	message_no	message_text
Number	Number	Text
chat_no	message_no	message_text
1	1	hello there my sweet
1	2	how are you doing
1	3	yes it is
1	4	glad to meet you
1	5	did you look at my profile
1	6	look at it
1	7	I do like your shape and looks
1	8	just look up aladiesmasseur
1	9	ok
1	10	a young 50 I do massage work
1	11	eureka here too california
1	12	I am just off of california st.
1	13	do the guys play with you at school

Figura 5. ***Estructura unificada de datos.***

Script III

Una vez extraídas las líneas de mensaje de los atacantes, se observa que contienen gran cantidad de ruido que afecta negativamente al análisis. Para afrontar este tipo de problemas, la metodología CRISP – DM recomienda realizar un formateo y estandarización de los datos, lo cual incluye eliminar caracteres ilegales, acortar las palabras a un máximo límite, trasladar a los verbos a su forma base, entre otras transformaciones.

Bajo estas consideraciones, se construye una función denominada *preprocessText.m*, la cual realiza la limpieza y estandarización de los datos automáticamente. Esta función puede ser aplicada para preparar diferentes colecciones de datos bajo el mismo procedimiento. El código fuente de la función *preprocessText.m* se incluye en el Anexo VI del presente documento, y la misma permite realizar las siguientes acciones sobre los datos:

- Convertir el texto a minúsculas.
- Separar el texto en *tokens*.
- Eliminar puntuación.
- Eliminar *stop words* ("*and*", "*of*", y "*the*").
- Eliminar palabras con 2 o menos caracteres.
- Eliminar palabras con 15 o más caracteres.
- Normalizar las palabras.

En la Figura 6 se puede apreciar el resultado de aplicar esta función sobre una cadena de texto, almacenada en la variable *newText*.

```
newText = "have to know the right people. you can make all GIRL movies
with them :D";
newDocuments = preprocessText(newText)
```

```
newDocuments =
  tokenizedDocument:

   6 tokens: know right people make girl movie
```

Figura 6. ***Función preprocessText.m sobre una cadena de texto***

Con el objetivo de contrastar el contenido del conjunto de datos depurado con el conjunto de datos original, se crea un modelo *bag-of-words* para cada conjunto. Esta representación simplificada de los datos permite realizar un análisis de frecuencia, el cual se ilustra gráficamente utilizando la función *wordcloud* en *Matlab*, como se observa en la Figura 7.

Figura 7. ***Conjunto de datos inicial vs. Conjunto de Datos pre-procesado***

2. Modelamiento del fenómeno

El análisis del estado del arte evidencia que la mayoría de autores han estudiado el fenómeno del Grooming desde la perspectiva psicológica [3][4][6][7]. En dichos estudios, se analiza el Grooming como un proceso compuesto por un número determinado de fases, las cuales se definen a partir de teorías psicológicas. La decisión de los autores añade un alto grado de subjetividad en sus investigaciones, ya que los resultados obtenidos no son verificables, al menos no computacionalmente. Por esta razón, la investigación se centra en analizar al Grooming desde la perspectiva de las Ciencias de la Computación, de manera que todos los resultados obtenidos pueden ser computacionalmente verificables y reproducibles.

Una investigación previa [8] determinó que el Grooming forma parte de los vectores de ataque de la Ingeniería Social. Este vector de ataque se encuentra estrechamente relacionado con las Amenazas Persistentes Avanzadas (APTs, por sus siglas en inglés), debido a que permanece durante prolongados períodos de tiempo sin ser detectado manteniendo acceso continuo a datos confidenciales y manipulando, en algunos casos, a la víctima mediante diversas técnicas. Bajo este contexto se determina que el Grooming debe ser evaluado como una APT, en cuanto a estructura y funcionamiento.

Las APTs así como el Grooming son ataques especializados que se efectúan bajo una estructura procedimental de varias etapas. Estas etapas conforman las diversas cadenas de ataque o ciclos de vida propuestos en la literatura. Se analizaron varios modelos de ciclo de vida de las APT [9 – 12], así como también del Grooming [13 – 15]. Estos modelos se utilizan como referencia para estudiar y definir un modelo de ciclo de vida del Grooming desde la perspectiva de la Seguridad de la Información.

2.1 Modelado de tópicos

Con el objetivo de determinar técnicamente las etapas del Grooming, se aplica una herramienta de minería de texto conocida como modelado de tópicos. El modelado de tópicos es un tipo de modelado estadístico que permite descubrir estructuras semánticas ocultas en un cuerpo de texto. Existen varias técnicas de modelado de tópicos, entre los más relevantes se encuentran el Análisis Semántico Latente (LSA, por sus siglas en inglés), el Análisis Semántico Latente Probabilístico (PLSA), el Modelo de Tema Correlacionados (CTM) y la Asignación Latente de Dirichlet (LDA).

Existen estudios que evalúan distintas técnicas de modelado de tópicos con el objetivo de determinar su rendimiento [16, 17], en ellos se determina que LDA es la técnica de modelado de tópicos más eficiente en aplicaciones que utilizan texto como principal fuente de datos. Bajo estos criterios y en función de la naturaleza de la investigación, se decide utilizar LDA para el modelado de tópicos.

LDA permite modelar la estructura de tópicos en colecciones de datos discretas, donde cada documento se considera como una combinación de tópicos. *Matlab* incorpora el algoritmo de LDA en su módulo de analítica de texto, y permite ajustarlo e implementarlo para diferentes tipos de aplicaciones. Para aplicar el modelo LDA sobre el conjunto de datos y descubrir las agrupaciones o tópicos de palabras, es necesario determinar antes el número de tópicos adecuado para ajustar el modelo. El número de tópicos en un documento está estrechamente relacionado con la naturaleza del texto que lo conforma, y la elección acertada en el número de tópicos está basada en un modelo estadístico.

Para escoger un número adecuado de tópicos, se compara la calidad de ajuste del modelo LDA con diferentes números de tópicos. Es posible evaluar la calidad de ajuste del modelo LDA calculando la perplejidad de un conjunto de documentos retenidos, que para este caso corresponden a una décima parte del conjunto de datos. La perplejidad es una medida de precisión que indica qué tan bien un modelo describe a un conjunto de documentos. Una perplejidad menor sugiere un mejor ajuste del modelo.

El objetivo de determinar adecuadamente el número de tópicos es garantizar que la perplejidad sea mínima en relación a otras posibles cantidades. Esta no es la única consideración, los modelos que se ajustan a un mayor número de tópicos pueden tardar más en converger. En consecuencia, el cálculo de la calidad de ajuste del modelo (*perplexity*) y el tiempo de convergencia (*time elapsed*) son las medidas decisivas en la elección del número de tópicos.

2.2 Implementación del modelo LDA

Utilizando *Matlab*, se ajusta el modelo LDA para un rango de valores compuesto de varios números de tópicos. Este rango de números de tópicos se determina a partir del análisis de los diferentes modelos de ciclo de vida APT y Grooming, entre los cuales se evidencia que el mínimo número de etapas es 2, como en el modelo propuesto por Lancaster [18], y el máximo 8, como se observa en los modelos propuestos por Mandiant, BSI y Sdapt [18]. Por tanto, el rango de valores para este caso de estudio se establece desde 2 hasta

10 tópicos, con el fin de observar el comportamiento de los datos en esa esfera de agrupamiento.

La Figura 8 muestra parte del código utilizado para llevar a cabo la selección del número de tópicos.

```
numTopicsRange = [2 4 6 8 10];
for i = 1:numel(numTopicsRange)
    numTopics = numTopicsRange(i);

    mdl = fitlda(bag,numTopics, ...
        'Solver','savb', ...
        'Verbose',0);

    [~,validationPerplexity(i)] = logp(mdl,documentsValidation);
    timeElapsed(i) = mdl.FitInfo.History.TimeSinceStart(end);
end
```

Figura 8. ***Ajuste del modelo LDA para un rango de tópicos***

Se compara el tiempo de convergencia y la perplejidad del modelo con el conjunto de documentos de prueba retenido. En la Figura 9, se evidencia la perplejidad y el tiempo transcurrido para cada número de tópicos dentro del rango determinado. La perplejidad se coloca en el eje izquierdo y el tiempo en el eje derecho, siendo el eje horizontal el número de tópicos respectivo.

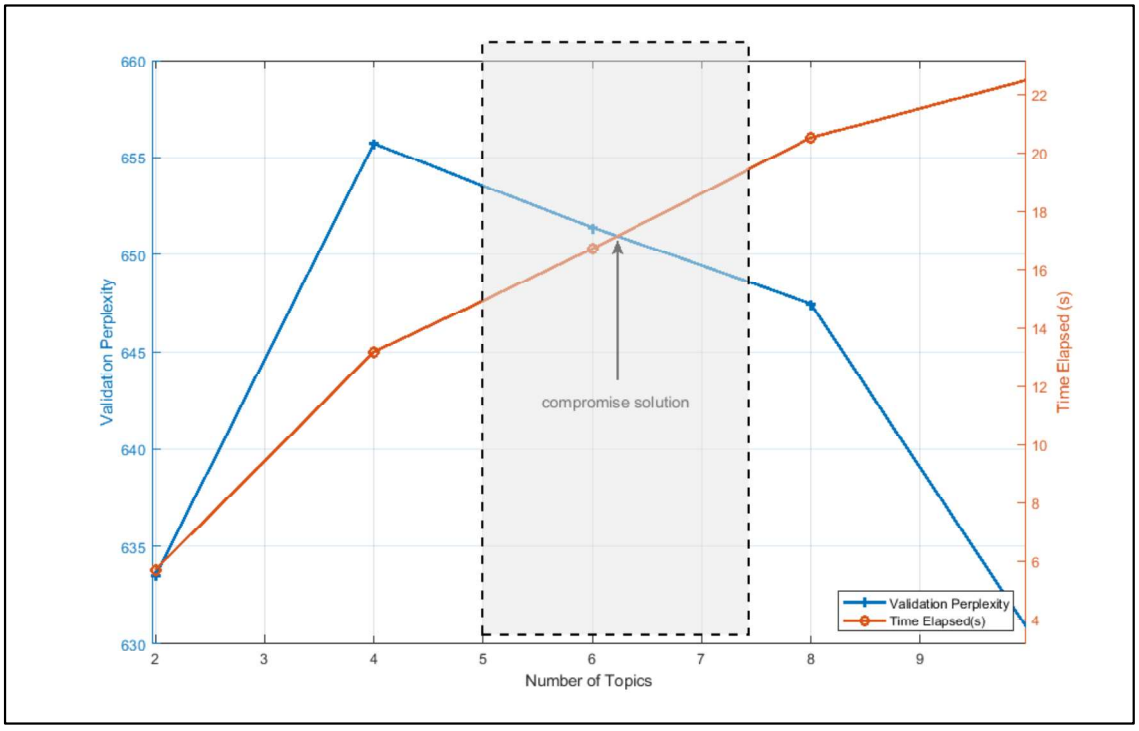


Figura 9. ***Solución de compromiso (Perplejidad vs. Tiempo)***

Mediante la gráfica obtenida, se establece una solución de compromiso entre la perplejidad de validación y el tiempo de convergencia. Una solución de compromiso para un problema con criterios en conflicto es una solución viable, que es la más cercana al ideal, y ayuda a los investigadores a tomar una decisión acertada. Se determina que ajustar el modelo LDA con seis tópicos es la elección idónea, ya que la perplejidad es menor comparada con los demás números de tópicos, y el tiempo de convergencia razonable con la cantidad de datos.

Utilizando *Matlab*, se crea un modelo *bag-of-words* a partir de los datos pre-procesados, y se ajusta el modelo LDA con un número de seis tópicos, previamente definidos. En la Figura 10, se observa el resultado del modelado de tópicos a través de nubes de palabras.

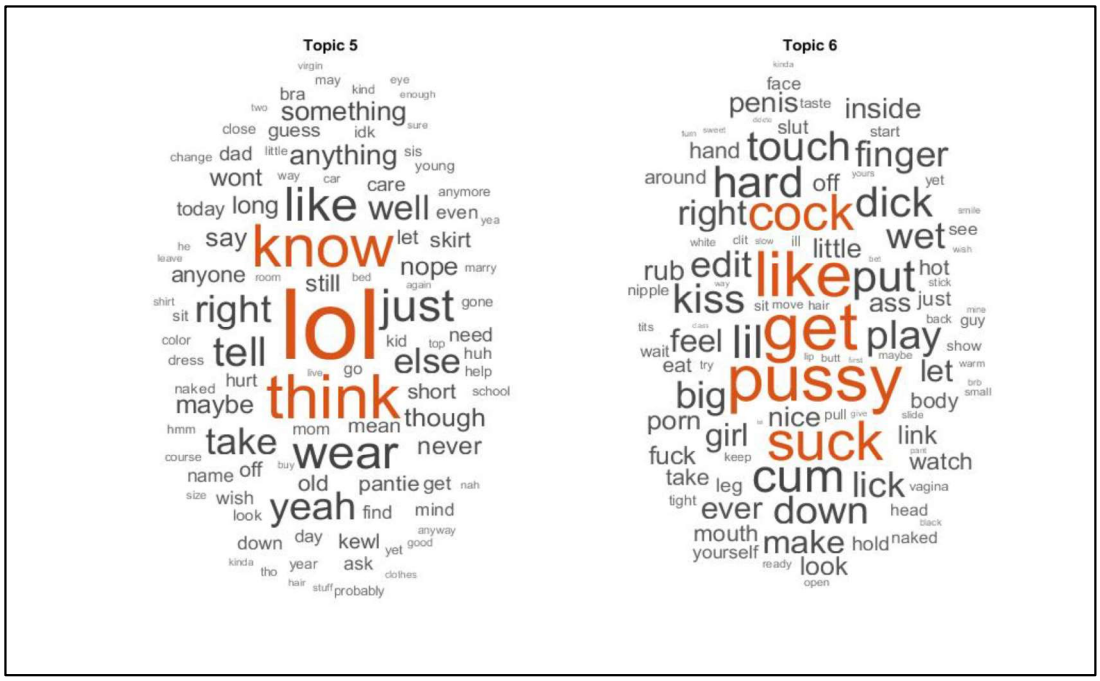


Figura 10. ***Tópicos LDA***

Un tópico consiste en una colección de palabras que se agrupan en base a su naturaleza. Definir los temas específicos alrededor de los cuales se construyen los tópicos constituye una ardua tarea, debido a que únicamente se cuenta con palabras agrupadas y sus respectivas frecuencias. Sin embargo, una observación rápida de las agrupaciones de palabras permite identificar rasgos significativos en las mismas. Por ejemplo, al observar el grupo de palabras correspondientes al Tópico 6, se evidencia que la mayoría de ellas son de carácter explícitamente sexual.

Con el objetivo de afinar la búsqueda de significado en cada una de las agrupaciones de tópicos, es necesario complementar el análisis LDA con un análisis lingüístico, el cual fue desarrollado como parte de la investigación, y los resultados obtenidos son fundamentales para el desarrollo de las secciones siguientes, en las cuales se hace referencia a las etapas del ciclo de vida del Grooming, determinado en la investigación[1].

2.3 Comprobación del modelo LDA

Con el objetivo de comprobar el funcionamiento del modelo LDA construido, se procede a aplicar el modelo directamente sobre cinco documentos específicos. Cada documento se corresponde con una línea de mensaje extraída del conjunto de datos. Los documentos a analizar son:

1. *“love give massage”*
2. *“nice warm lotion body”*
3. *“love”*
4. *“give nice ass rub”*
5. *“yeah why”*

El resultado de la aplicación del modelo LDA sobre los documentos se puede observar de forma gráfica en la Figura 11, donde se evidencia que cada documento es considerado como una mezcla de tópicos, los cuales se presentan en proporciones distintas. De esta forma existe la probabilidad de que, en un documento, un tópico se halle en mayor proporción, lo cual permite inferir que dicho documento trata de ese tópico en particular.

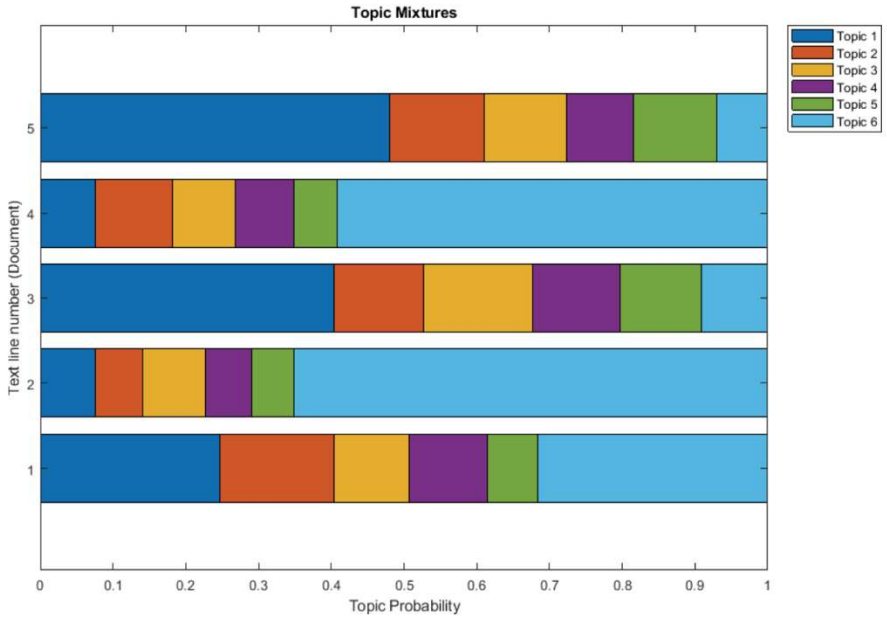


Figura 11. ***Mezcla de tópicos LDA en diferentes documentos***

Se comprueba también el modelo LDA con un documento nuevo, es decir, una línea de texto que no forma parte del conjunto de datos. El nuevo documento contiene el mensaje de texto: *"this will be our little secret.. do not tell your parents about me.. I can get in trouble"*. El mensaje fue analizado aplicando el modelo LDA y se obtuvo el resultado representado en la Figura 12. El gráfico de barras resultante demuestra que el documento analizado se compone de una mezcla de los seis tópicos identificados, pero se observa que es más probable que el documento pertenezca específicamente al tópico número tres.

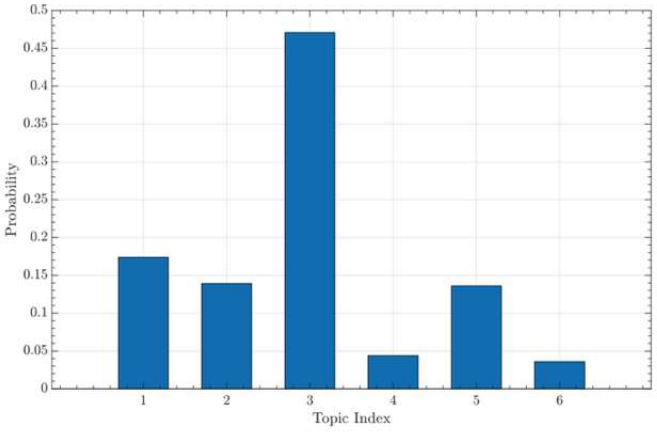


Figura 12. ***Presencia de tópicos en un nuevo documento.***

3. Aplicación de modelos de aprendizaje de máquina

Como resultado del análisis lingüístico realizado por colaboradores expertos de la investigación [1], se determinaron las intenciones comunicacionales del atacante en cada uno de los grupos de tópicos identificados. Estas intenciones comunicacionales dieron lugar a la definición conceptual de cada una de las estaciones que conforman el ciclo de vida del Grooming. Para aplicar este modelo de ciclo de vida sobre el conjunto de datos, se decide aplicar técnicas de aprendizaje de máquina, para lo cual es necesario previamente contar con el conjunto de datos etiquetado acorde con las estaciones del Grooming definidas.

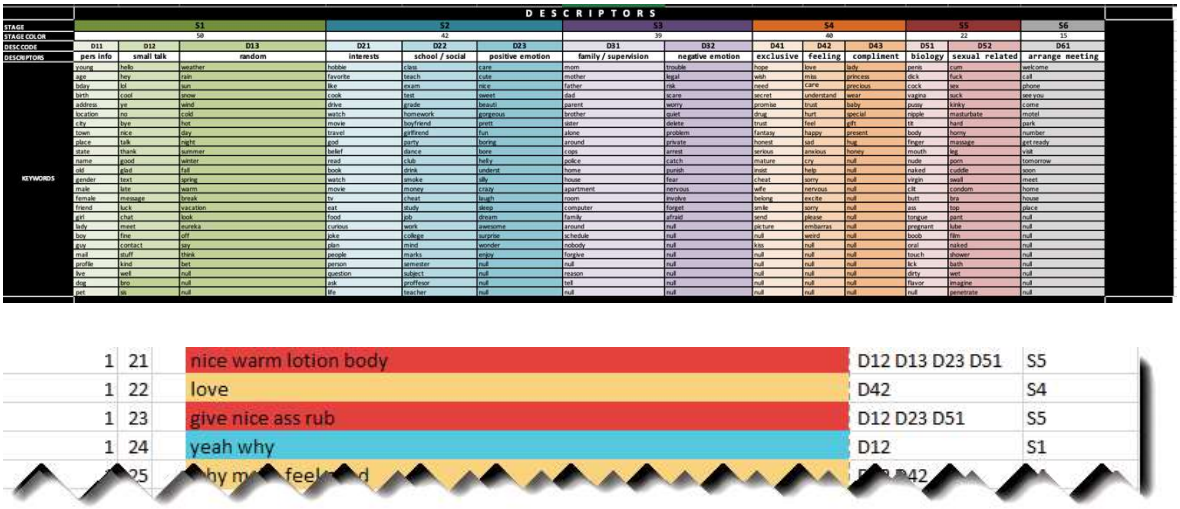
3.1 Etiquetado de los datos

Con el objetivo de etiquetar las líneas de mensaje dentro del conjunto de datos, se desarrolla un método basado en descriptores. Un descriptor corresponde a un identificador que se le asigna a cada una de las intenciones comunicacionales identificadas durante el análisis lingüístico [1].

El número de líneas de mensajes contenidas en el conjunto de datos está en el orden de los millones, lo cual dificulta la posibilidad de un etiquetado manual, proceso que algunos autores optan por realizar en estudios relacionados [3]. Se construye un script en *VBA*, el cual permite identificar descriptores en las líneas de mensaje y en base a ellos asignar automáticamente una etiqueta, correspondiente a la estación del Grooming respectiva.

Una línea de mensaje puede contener uno o varios descriptores, la presencia y frecuencia de los mismos conduce al etiquetado del mensaje dentro una estación específica del Grooming.

En la Figura 13 se muestra un fragmento del proceso aplicado sobre el conjunto de datos para identificar los descriptores presentes en cada una de las líneas de mensaje. Una vez identificados los descriptores, se etiqueta automáticamente cada línea de mensaje con su estación del Grooming respectiva, en base a la presencia y frecuencia de dichos descriptores.



1	21	nice warm lotion body	D12 D13 D23 D51	S5
1	22	love	D42	S4
1	23	give nice ass rub	D12 D23 D51	S5
1	24	yeah why	D12	S1

Figura 13. ***Identificación de descriptores y asignación automática de etiquetas***

Las etiquetas se añaden a la colección unificada de datos denominada *chatLogs.csv*. Esta colección obedece la estructura que se observa en la Figura 14, donde el campo añadido *message_stage* contiene la estación en la que se identifica cada una de las líneas de mensaje.

chatLogs

chat_no	message_no	message_text	message_stage
Number	Number	Text	Number
1	63	the one by the zoo we could park and...	S6
1	64	just my roommate a female	S3
1	65	do you like other girls she could join us	S2
1	66	hello there	S1
1	67	how is your night	S1
1	68	cool	S1
1	69	sorry for getting away with myself ear...	S4
1	70	I am doing good	S1
1	71	I mean the way I talk't to you	S1
1	72	got a call and had to go do a massage	S6
1	73	no went to her house	S3

Figura 14. ***Conjunto de datos etiquetados***

3.2 Clasificación automática utilizando un modelo estadístico

Una vez que el conjunto de datos está etiquetado, es posible aplicar técnicas de aprendizaje de máquina para clasificación automática. Estas técnicas utilizan diferentes algoritmos para detectar patrones en los datos y obtener un nivel de aprendizaje que permite categorizar los datos automáticamente en base al conocimiento adquirido.

Clasificador estadístico

Existen estudios que evalúan diversas técnicas de clasificación con el objetivo de determinar su rendimiento en distintas aplicaciones [19], en los cuales se ha evidenciado que los clasificadores estadísticos son los más eficientes en aplicaciones de clasificación de texto. Bajo estos criterios y en función de la naturaleza de la investigación, se decide ajustar un modelo de clasificación lineal para el aprendizaje automático.

A continuación, se describe el proceso para entrenar un clasificador estadístico basado en conteo de frecuencia de palabras, partiendo de un modelo *bag-of-words*, creado a partir del conjunto de datos pre-procesado. El modelo resultante permite predecir la estación específica del Grooming a la que pertenece una determinada línea de mensaje. El código empleado para la aplicación del algoritmo de clasificación se adjunta en el Anexo V.

1. La estructura unificada de datos *chatLogs.csv*, la cual contiene los datos pre-procesados y etiquetados, se carga a *Matlab* y se extraen los datos de texto.

2. Se construye un histograma de distribución de clase para identificar la frecuencia de cada una de las estaciones del Grooming en el conjunto de datos, de acuerdo a las etiquetas asignadas. El resultado se muestra en la Figura 15.
3. El conjunto de datos se secciona en dos conjuntos, un conjunto de entrenamiento y un conjunto excluido para prueba y validación.
4. El modelo de clasificación que toma como entrada el modelo *bag-of-words* fue ajustado y entrenado. La Figura 16 muestra una parte del código utilizado en este proceso.

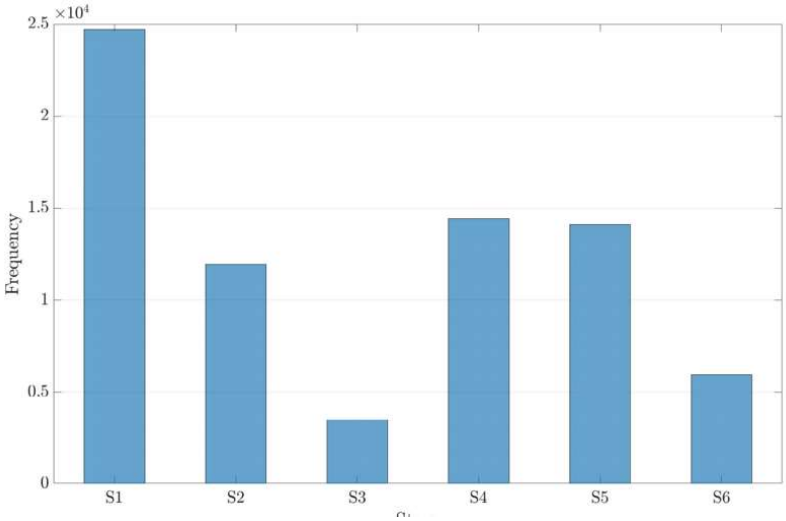


Figura 15. ***Histograma de frecuencias de las estaciones del Grooming***

```
XTrain = bag.Counts;
mdl = fitcecoc(XTrain,YTrain,'Learners','linear')
```

```
mdl =

  classreg.learning.classif.CompactClassificationECOC
      ResponseName: 'Y'
        ClassNames: [S1    S2    S3    S4    S5    S6]
    ScoreTransform: 'none'
    BinaryLearners: {15×1 cell}
      CodingMatrix: [6×15 double]


  Properties, Methods
```

Figura 16. ***Entrenamiento del clasificador lineal***

5. Una vez creado y entrenado, el clasificador se aplica sobre el conjunto de datos para predecir las etiquetas de las líneas de mensaje. Se calcula la precisión de la clasificación, siendo esta la proporción de etiquetas que el modelo predijo correctamente. Se obtuvo un resultado del 97.61%, como se puede observar en la Figura 17.

```
documentsTest = preprocessText(textDataTest);
XTest = encode(bag,documentsTest);
```

```
YPred = predict(mdl,XTest);
acc = sum(YPred == YTest)/numel(YTest)
```

```
acc =

0.9761
```

Figura 17. ***Resultados del modelo de clasificación lineal***

6. Se crea una matriz con nuevos documentos para comprobar el funcionamiento del modelo. Estos nuevos documentos se crean intencionalmente de forma que la pertenencia de cada uno de ellos a una de las estaciones del ciclo de vida del Grooming sea explícita. Las estaciones del Grooming fueron definidas como parte de la investigación general [1], luego de realizar un análisis lingüístico con los grupos de tópicos determinados previamente. Cada uno de los seis tópicos se corresponde con una estación del Grooming, como se puede observar en la Figura 18. Los documentos creados y sus respectivas estaciones se muestran en la Tabla 4.

	Documento	**Estación del Grooming**
1	*"is your dad usually around"*	*S3*
2	*"do u wanna come in like an hour"*	*S6*
3	*"wish you had some more body pics"*	*S5*

Tabla 4. ***Documentos de evaluación.***

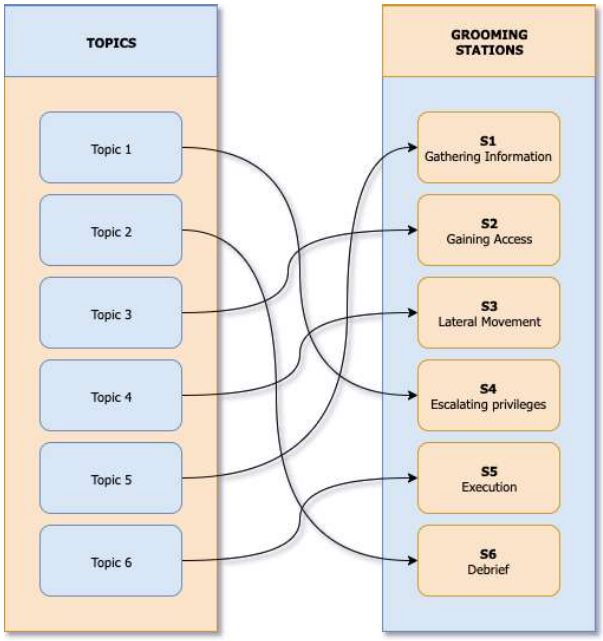


Figura 18. ***Correspondencia de tópicos con estaciones del Grooming [1]***

7. Se aplica el clasificador sobre los nuevos documentos y se obtiene como resultado exactamente las mismas etiquetas definidas previamente, como se observa en la Figura 19. De esta forma se logra comprobar el funcionamiento del modelo, justificando así el alto grado de precisión obtenido.

```
str = [ ...
    "is your dad usually around?"
    "do u wanna come in like an hour?"
    "wish you had some more body pics"];
documentsNew = preprocessText(str);
XNew = encode(bag,documentsNew);
labelsNew = predict(mdl,XNew)
```

```
labelsNew =

  3×1 categorical array

     S3
     S6
     S5
```

Figura 19. ***Resultados del modelo de clasificación lineal con nuevos datos***

Clasificadores basado en redes neuronales

Existen algunos estudios que aplican técnicas de clasificación basadas en redes neuronales a problemas de clasificación de texto, obteniendo resultados favorables [20]. A pesar de que estos algoritmos se utilizan principalmente en aplicaciones que cuentan con imágenes como datos, se ha evidenciado que pueden ser aplicados en múltiples problemas. Bajo estos criterios y en función de la naturaleza de la investigación, se decide implementar dos nuevas técnicas de clasificación basadas en redes neuronales.

Haciendo uso de los algoritmos implementados en *Matlab*, se ajusta un clasificador basado en una red de aprendizaje profundo Long Short Term Memory (LSTM, por sus siglas en inglés), y un clasificador basado en una red neuronal convolucional (CNN, por sus siglas en inglés). A continuación, se describe el proceso desarrollado para ajustar, entrenar y evaluar dichos clasificadores.

Clasificador LSTM

Los datos de texto son naturalmente secuenciales. Un fragmento de texto es una secuencia de palabras, que pueden tener dependencias entre ellas. Con el objetivo de aprender y usar dependencias a largo plazo para clasificar los datos secuenciales, se utiliza una red neuronal LSTM.

Una red LSTM es un tipo de red neuronal recurrente (RNN, por sus siglas en inglés) que puede aprender dependencias a largo plazo entre los pasos de tiempo de datos secuenciales [20]. Para ingresar texto a una red LSTM, primero se convierten los datos de texto en secuencias numéricas. Para lograr esto se utiliza un codificador de palabras que asigne documentos a secuencias de índices numéricos.

Matlab, a través de *Text Analytics Toolbox* incluye el algoritmo de LSTM y permite ajustarlo para aplicaciones específicas. A continuación, se describen los pasos generales que se realizaron para crear, entrenar y utilizar la red LSTM:

1. Importar y pre-procesar los datos.
2. Convertir las palabras en secuencias numéricas usando una codificación de palabras. Para ello es necesario recurrir a la función *wordEncoding*.
3. Crear y entrenar la red LSTM. El código utilizado se adjunta en el Anexo VI. La Figura 20 muestra parte de las instrucciones utilizadas para la creación de la red. La Figura 21 muestra el proceso de entrenamiento, el cual realiza un total de 4080

iteraciones en un tiempo de 16 minutos y 11 segundos, dando como resultado una precisión de validación del 95.91%.

```
inputSize = 1;
embeddingDimension = 100;
numWords = enc.NumWords;
numHiddenUnits = 180;
numClasses = numel(categories(YTrain));

layers = [ ...
    sequenceInputLayer(inputSize)
    wordEmbeddingLayer(embeddingDimension,numWords)
    lstmLayer(numHiddenUnits,'OutputMode','last')
    fullyConnectedLayer(numClasses)
    softmaxLayer
    classificationLayer]
```

```
layers =

  6x1 Layer array with layers:

     1   ''   Sequence Input          Sequence input with 1 dimensions
     2   ''   Word Embedding Layer    Word embedding layer with 100 dimensions and 9936 unique words
     3   ''   LSTM                   LSTM with 180 hidden units
     4   ''   Fully Connected         6 fully connected layer
     5   ''   Softmax                softmax
     6   ''   Classification Output   crossentropyex
```

Figura 20. ***Creación de la red LSTM***

```
net = trainNetwork(XTrain,YTrain,layers,options);
```

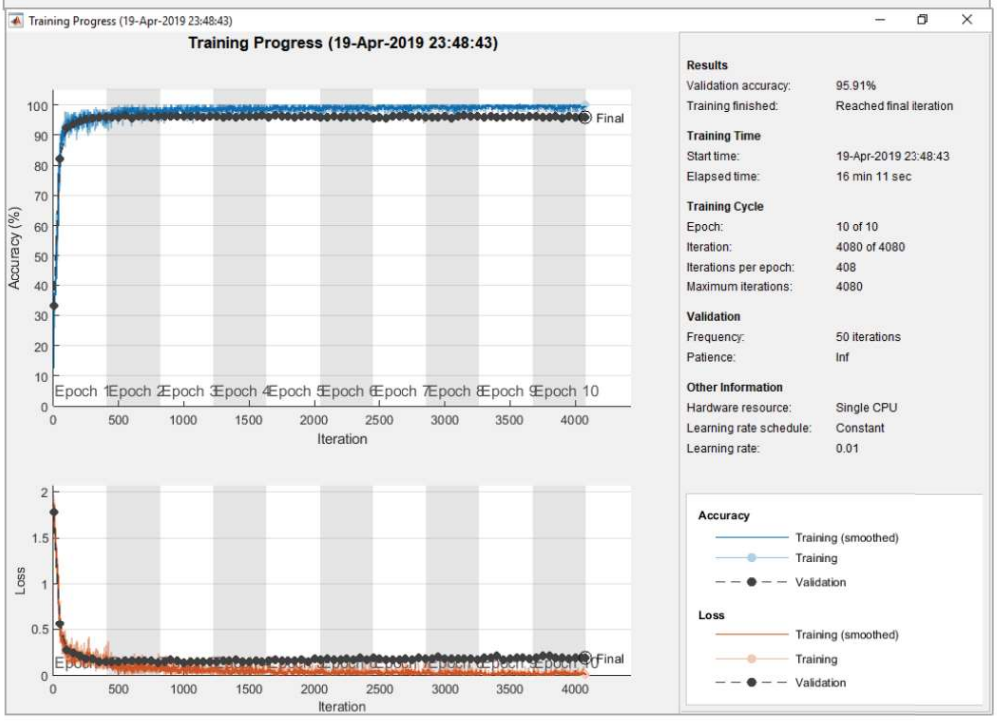


Figura 21. ***Entrenamiento y resultados de la red LSTM***

Clasificador CNN

Para clasificar los datos de texto usando convoluciones, es necesario convertir los datos de texto en imágenes. Al igual que la red LSTM, *Text Analytics Toolbox* incluye el algoritmo de CNN y permite ajustarlo para aplicaciones específicas. A continuación, se describen los pasos generales que se realizaron para crear, entrenar y utilizar la red CNN:

1. Cargar un conjunto de palabras embebidas previamente entrenado utilizando la función *fastTextWordEmbedding.*
2. Seccionar el conjunto de datos en un conjunto de entrenamiento y un conjunto de validación y pruebas. La división se realiza en una proporción de 70 – 30, respectivamente.
3. Crear un *datastore* tabular a partir del conjunto de datos de entrenamiento.
4. Definir la arquitectura de red para la tarea de clasificación, la cual se puede visualizar en la Figura 22.
5. Entrenar la red a través de la función *trainNetwork*. En la Figura 23 se puede visualizar el proceso de entrenamiento de la red, el cual completó 10 iteraciones en un tiempo de 147 minutos y 18 segundos, dando como resultado una precisión de validación del 96.11%.
6. Probar la Red con los datos de prueba retenidos, y realizar predicciones en los datos de prueba utilizando la red entrenada.

```
for j = 1:numBlocks
    N = ngramLengths(j);
    lgraph = connectLayers(lgraph,"max"+N,"depth/in"+j);
end

figure
plot(lgraph)
title("Network Architecture")
```

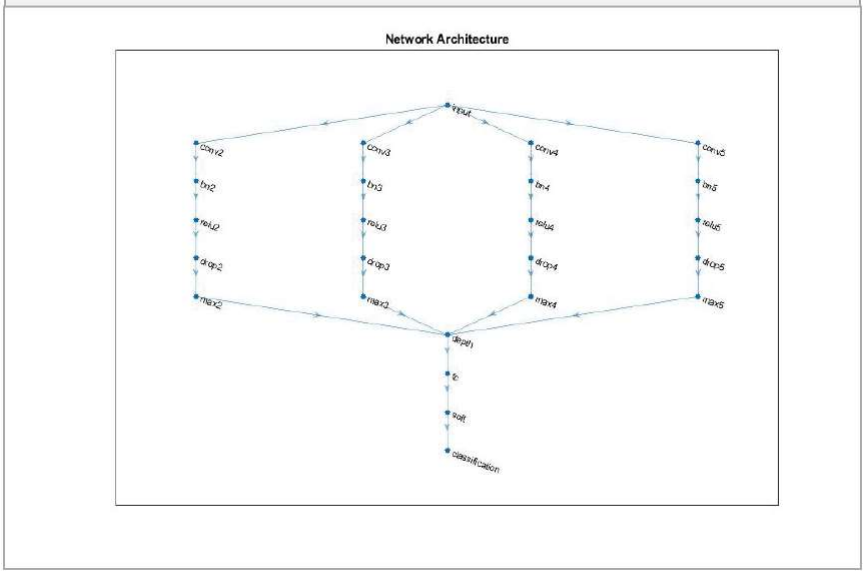


Figura 22. ***Arquitectura de la red neuronal convolucional***

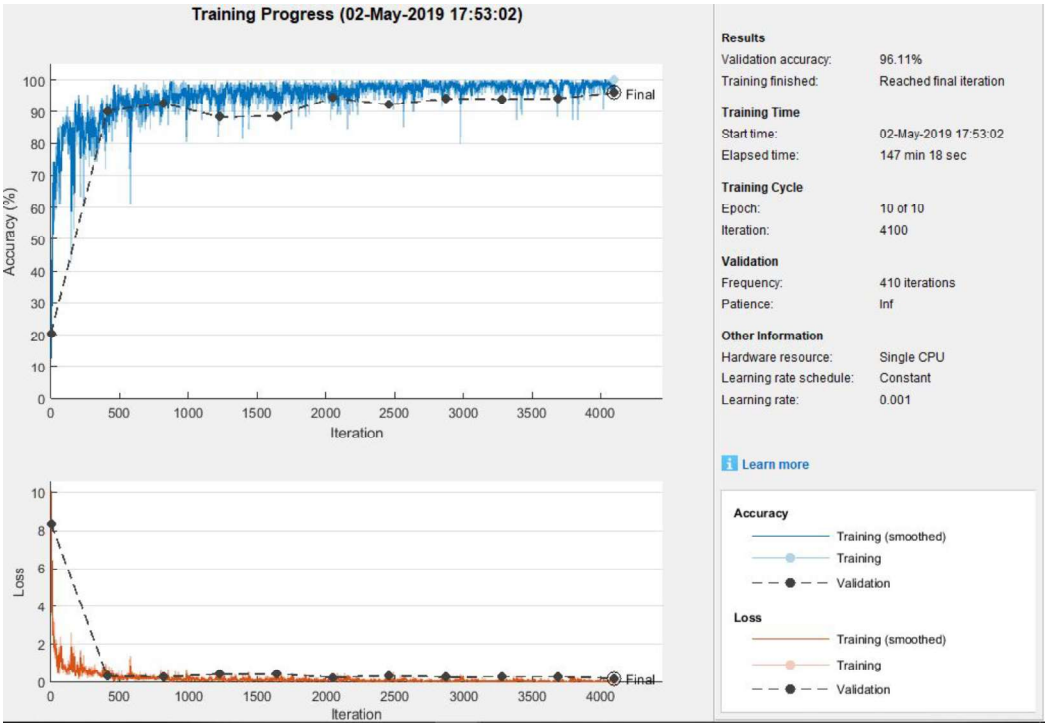


Figura 23. ***Entrenamiento y resultados de la red CNN***

RESULTADOS Y DISCUSIÓN

Los resultados del estudio actual sugieren que evidentemente el Grooming puede y debe ser analizado como un vector de ataque de la Ingeniería Social. El estudio del fenómeno del Grooming desde la perspectiva de la Seguridad de la Información hace posible abordar el problema desde un ámbito técnico científico. Esto permite crear una base de conocimiento objetiva, donde todos los resultados obtenidos son verificables, reproducibles y reutilizables.

Se realizaron varias tareas concentradas en tres principales áreas de estudio utilizando datos provenientes de PJ. Este conjunto de datos es desafiante debido a su naturaleza. El lenguaje utilizado en las conversaciones es informal, el vocabulario se compone de jerga propia de los chats en Internet, errores de taquigrafía, emoticones, símbolos de puntuación y faltas ortográficas. Por esta razón, el pre-procesamiento y limpieza de los datos se constituyó como un paso esencial antes de que el análisis pueda ser realizado.

Para el modelado de tópicos con LDA se probaron diferentes modelos *bag-of-words* compuestos por trigramas, bigramas y unigramas, siendo estos últimos los que permitieron obtener el mejor ajuste del modelo. Esto se debe a que las propiedades inherentes de los datos dificultan el análisis de los mismos como frases multi palabras (*n-gramas*). No obstante, el análisis de los datos como palabras aisladas (*unigramas*), basado en la frecuencia de repetición de las mismas, condujo a un descubrimiento de tópicos coherente con el fenómeno estudiado.

Los resultados de clasificación automática sugieren que, independientemente de la complejidad del algoritmo, es el tipo de datos utilizados el cual determina el modelo de clasificación idóneo. De esta forma, se determinó que el modelo de clasificación lineal o estadístico era el más eficiente, debido a que se trata de un problema de clasificación de texto. Los algoritmos basados en redes neuronales también obtuvieron altos porcentajes de clasificación, sin embargo, conllevaron una utilización exhaustiva de tiempo y recursos computacionales.

CONCLUSIONES Y RECOMENDACIONES

Para trabajos futuros, es recomendable recopilar un conjunto de datos mucho más extenso y, si fuera posible, uno con víctimas reales involucradas. Esto brindaría la oportunidad de modelar el estado mental de la víctima con un enfoque similar al que se ha presentado en este estudio. La capacidad de predecir la vulnerabilidad de las víctimas en interacciones a través de redes sociales podría tener un gran impacto en la mitigación del Grooming, ya que podría usarse para desarrollar estrategias de intervención para prevenir el ataque o detectarlo oportunamente.

Afortunadamente, los resultados de este estudio pueden servir de base para el desarrollo de nuevas investigaciones, al proporcionar información detallada sobre cómo detectar patrones de lenguaje y estrategias que los atacantes utilizan cuando acechan víctimas potenciales en línea.

REFERENCIAS BIBLIOGRÁFICAS

[1] Zambrano, P., Torres, J., Tello-Oquendo, L., Jácome, R., Benalcázar, M. E., Andrade, R., & Fuertes, W. (2019). Technical Mapping of the Grooming Anatomy Using Machine Learning Paradigms: An Information Security Approach. IEEE Access, 7, 142129-142146.

[2] Perverted Justice. (n.d.). Retrieved from http://www.perverted-justice.com/.

[3] P. J. Black, M. Wollis, M. Woodworth, and J. T. Hancock, ‘‘A linguistic analysis of grooming strategies of online child sex offenders: Implications for our understanding of predatory sexual behavior in an increasingly computer-mediated world,’’ Child Abuse Neglect, vol. 44, pp. 140–149, Jun. 2015.

[4] N. Pendar, ‘‘Toward spotting the pedophile telling victim from predator in text chats,’’ in Proc. Int. Conf. Semantic Comput. (ICSC), Sep. 2007, pp. 235–241.

[5] Azevedo, Ana Isabel Rojão Lourenço, and Manuel Filipe Santos. "KDD, SEMMA and CRISP-DM: a parallel overview." IADS-DM (2008).

[6] D. Bogdanova, P. Rosso, and T. Solorio, ‘‘On the impact of sentiment and emotion based features in detecting online sexual predators,’’ in Proc. 3rd Workshop Comput. Approaches Subjectivity Sentiment Anal., Jul. 2012, pp. 110–118. [Online]. Available: http://www.aclweb. org/anthology/W12-3717

[7] D. Bogdanova, P. Rosso, and T. Solorio, ‘‘Modelling fixated discourse in chats with cyberpedophiles,’’ Proc. Workshop Comput. Approaches Deception Detection, 2012, pp. 86–90. [Online]. Available: http://www.aclweb. org/anthology/W12-0413

[8] Zambrano, Patricio, Jenny Torres, and Pamela Flores. "How Does Grooming Fit into Social Engineering?." Advances in Computer Communication and Computational Sciences. Springer, Singapore, 2019. 629-639.

[9] P. Chen, L. Desmet, and C. Huygens, ‘‘A study on advanced persistent threats,’’ in Proc. IFIP Int. Conf. Commun. Multimedia Secur. Berlin, Germany: Springer, 2014, pp. 63–72.

[10] B. I. Messaoud, K. Guennoun, M. Wahbi, and M. Sadik, ‘‘Advanced persistent threat: New analysis driven by life cycle phases and their challenges,’’ in Proc. Int. Conf. Adv. Commun. Syst. Inf. Secur. (ACOSIS), Oct. 2016, pp. 1–6.

[11] A. Alshamrani, S. Myneni, A. Chowdhary, and D. Huang, ‘‘A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities,’’ IEEE Commun. Surveys Tuts., vol. 21, no. 2, pp. 1851–1877, 2nd Quart., 2019

[12] L. Shenwen, L. Yingbo, and D. Xiongjie, ‘‘Study and research of APT detection technology based on big data processing architecture,’’ in Proc. IEEE 5th Int. Conf. Electron. Inf. Emergency Commun., May 2015, pp. 313–316.

[13] O’Connell, Rachel. "A typology of child cybersexploitation and online grooming practices." Preston, UK: University of Central Lancashire (2003).

[14] L. Penna, A. Clark, and G. Mohay, ‘‘Challenges of automating the detection of paedophile activity on the Internet,’’ in Proc. 1st Int. Workshop Systematic Approaches Digit. Forensic Eng., 2005, pp. 206–220.

[15] R. C. Hall, ‘‘A profile of pedophilia: Definition, characteristics of offenders, recidivism, treatment outcomes, and forensic issues,’’ Mayo Clinic Proc., vol. 82, no. 4, pp. 457–471, 2007

[16] Jacobi, Carina, Wouter Van Atteveldt, and Kasper Welbers. "Quantitative analysis of large amounts of journalistic texts using topic modelling." Digital Journalism 4.1 (2016): 89-106.
[17] R. Alghamdi and K. Alfalqi, ‘‘A survey of topic modeling in text mining,’’ Int. J. Adv. Comput. Sci. Appl., vol. 6, no. 1, pp. 147–153, 2015.
[18] B. I. Messaoud, K. Guennoun, M. Wahbi, and M. Sadik, ‘‘Advanced persistent threat: New analysis driven by life cycle phases and their challenges,’’ in Proc. Int. Conf. Adv. Commun. Syst. Inf. Secur. (ACOSIS), Oct. 2016, pp. 1–6.
[19] Kumar, Raj, and Rajesh Verma. "Classification algorithms for data mining: A survey." International Journal of Innovations in Engineering and Technology (IJIET) 1.2 (2012): 7-14.
[20] Xing, Zhengzheng, Jian Pei, and Eamonn Keogh. "A brief survey on sequence classification." ACM Sigkdd Explorations Newsletter 12.1 (2010): 40-48.

ANEXOS

I. Artículo científico "*Technical Mapping of the Grooming Anatomy Using Machine Learning Paradigms: An Information Security Approach*" [1]

The following article was published on the journal IEEE Access on September 20th, 2019, under DOI: 10.1109/ACCESS.2019.2942805; and distributed worldwide through IEEE Xplore.

ISSN: 2169-3536

IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 19, 2019, accepted September 3, 2019, date of publication September 20, 2019, date of current version October 10, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2942805

Technical Mapping of the Grooming Anatomy Using Machine Learning Paradigms: An Information Security Approach

PATRICIO ZAMBRANO[1], JENNY TORRES[1], LUIS TELLO-OQUENDO[2], RUBÉN JÁCOME[1], MARCO E. BENALCÁZAR[1], ROBERTO ANDRADE[1], AND WALTER FUERTES[3]

[1]Department of Informatics and Computer Science, Escuela Politécnica Nacional, Quito 170517, Ecuador
[2]Faculty of Engineering, Universidad Nacional de Chimborazo, Riobamba 060110, Ecuador
[3]Faculty of System Engineering, Universidad de las Fuerzas Armada, Sangolquí 171103, Ecuador

Corresponding author: Patricio Zambrano (patricio.zambrano@cpn.edu.cc)

This work was supported by the Ecuadorian Corporation for the Development of Research and the Academy (RED CEDIA) in the development of this study, within the Project under Grant GT-Cybersecurity.

ABSTRACT In the field of information security, there are several areas of study that are under development. Social engineering is one of them that addresses the multidisciplinary challenges of cyber security. Nowadays, the attacks associated with social engineering are diverse, including the so-called Advanced Persistent Threats (APTs). These have been the subject of numerous investigations; however, cybernetic attacks of similar nature as grooming have been excluded from these studies. In the last decade, various efforts have been made to understand the structure and approach of grooming from the field of computer science with the use of computational learning algorithms. Nevertheless, these studies are not aligned with information security. In this work, the study of grooming is formalized as a social engineering attack, contrasting its stages or phases with life cycles associated with APTs. To achieve this goal, we use a database of real cyber-pedophile chats; this information was refined and the Latent Dirichlet Allocation (LDA) topic modeling was applied to determine the stages of the attack. Once the number of stages was determined, we proceed to give them a linguistic context, and with the use of machine learning, a linear model was trained to obtain 97.6% of training accuracy. With these results, it was determined that the study of grooming could support research associated with social engineering and contribute to new fields of information security.

INDEX TERMS Cyber-pedophile, pedophile, grooming, pattern behavior, APT, social engineering.

I. INTRODUCTION

Recently, attacks on the privacy of children and adolescents through technological means have increased considerably. Investigations related to this social problem addressed different topics such as: the study and analysis of children's images in P2P networks, planning of security models in audio-visual devices for child control with access to the Internet, study of vulnerabilities in online video games, development of communication bots for the detection of potential attackers, forensic tools and analysis of pedophile behavior within instant messages [1]–[6]. Several scientific proposals that have studied the behavior of online attackers stand out in the study of instant messaging. Researchers in this field determined that the most common technique applied by attackers is grooming.

The associate editor coordinating the review of this manuscript and approving it for publication was Yassine Maleh.

This technique is characterized by using deceptive linguistic expressions to create environments of false friendship and trust, thus seducing victims to manipulate and gaining control over them. Some authors, supported by previous research and psychological studies apply techniques of text mining and machine learning to determine the nature and different levels of danger of this attack [4], [7]–[9]. However, it has been shown that the results of the research related to the study of grooming, contemplate different lines of research, are not conclusive and support other relevant studies [6], [10]–[12].

Online pedophilia and grooming have been studied for over a decade [1]–[3], [7], [13]. In the scientific field, grooming has been conceptualized as a procedural technique used by cybernetic attackers [1]. On the other hand, it is also regarded as an operational concept, whereby an attacker applies search strategies affinity, while acquires information and sexually desensitizes victims to develop relationships that lead to the

satisfaction of the needs of the offender or attacker [2]. The main motivations of this research are to delimit the technical anatomy of grooming, justifying its relevance and support for future investigations of relevance. These attacks are also known as social engineering semantic attacks and are considered pervasive threats to computer systems, communication, and privacy [14].

In this study, as the first phase, we propose a technological alternative that allows the grooming life cycle to be determined through stations or phases named *topics* with the use of the Latent Dirichlet Allocation (LDA) generative probabilistic framework, belonging to the field of topic modeling within text mining. LDA allows modeling the topic structure of documents and other discrete data collections, where each document is generated as a mix of topics. In this way, LDA assigns a topic to a set of words contained in each document [15]. Then, two experiments are proposed. In *experiment 1*, several topics are determined according to the characteristics of the pre-processed data. To obtain the data and its processing, the recommendations of the CRISP-DM methodology [16] were followed. After determining an optimal number of stations, we proceeded to give them a logical context through *experiment 2*, which uses studies related to linguistics and communicational intentions to order the topics determined by LDA. Within this ordering, several proposals of life cycles of Advanced Persistent Threats (APT) with the topics were related, thus determining the life cycle of the grooming.

The main contributions of this study are summarized as follows

- A psychological and technical profile of the type of attacker associated with online pedophilia is presented;
- Grooming as a vector of attack within social engineering and information security is positioned; this will allow supporting investigations related to determining patterns of malicious behavior online;
- Through the modeling of topics, different stages or seasons of a life cycle of an attack associated with social engineering is determined;
- Application of a linear machine learning algorithm to classify texts binding to the study area.

The article proceeds as follows. Section II introduces some definitions about grooming, its stages, and tools to detect it. Section III establishes the physicological/technical profile of a cyber-pedophile based on the use of technological resources. Section IV presents the related work on the topic. Section V details the methodology we follow, defines the research questions, and introduces the experimental approach. Section VI develops the experiments carried out. Section VII presents the answers to the research questions based on the results of the experimental phase. Finally, Section VIII draws the conclusions and present the future work.

II. WHAT IS GROOMING?

With the advance and use of communication technologies, the evolution of pedophile attacks in cyberspace and their strategies in approaching their potential victims has been evidenced. Within these strategies, there is the attack known as *grooming* that is used to create deceptive trust relationships between victims and attackers. The grooming has been considered as a case study, for more than a decade. From this, related research has generated significant contributions to society [1]–[3], [7], [13].

In the scientific field, grooming has been conceptualized as a procedural technique, criminal activity, or operational concept used by cybernetic attackers, who in some cases have a disorder of sexual preference for children or adolescents. In the development of false friendship, the attacker applies strategies to determine affinities, tastes or activities of interest to the victim, thus developing a relationship of trust where the main objective is sexual desensitization, giving rise to the satisfaction of the needs of the attacker; as the sexual act [1], [2], [12]. As a computer attack technique, grooming can be applied for very long periods, in order to guarantee the cooperation of its victims and minimizing the risk of exposure. Another aspect to be considered within the technique of grooming includes the preparation of relatives close to the victims to create an atmosphere of acceptance and normalization of a potential attack [1], [17], [18].

The study of grooming is comparable to the study and analysis of modern and contemporary computer attacks. The development of proactive measures and the advancement of investigations are limited by several aspects, such as access to databases of pedophile content, victims and relatives and means of communication, such as the Internet.

The online activist Perverted-justice (PJ) Foundation has collaborated extensively with the study of pedophile communications by continuously publishing actual conversations on its website. The primary purpose of this foundation is to eradicate online attackers by exposing the conversations and their actors [2]. With this background, several scientists began to study and analyze the text strings published by this web portal, thus determining psychological and technical behavioral traits when applying grooming in preparation for victims. One of the most significant challenges evidenced in the studies of chat chains is phonetics and phonology, which reveal the fields of study of morphology, syntax, semantics, pragmatics, and discourse.

Studies have determined that online pedophiles tend to seduce their victims through attention, affection, kindness, and even gifts through the use of information technologies [17]. Such is the case that in a survey applied to 437 schoolchildren between the ages of 11 and 13 years, it was determined that the use of the Internet and the chat communication protocol were part of their regular habits. 59% of the participants accepted to have participated regularly in chats with people through the Internet. 24% of people who chatted online admitted having delivered some personal information. These include the home phone number, the mobile phone number, and the home address. The most alarming fact in this study was when 37 children admitted to

making arrangements to meet the person they were chatting with [11].

A. STAGES OF GROOMING

When talking about grooming, an essential strategy for an attacker is the **sexual desensitization**. Kong et al. in [19] consider it as a common strategy that offenders use for a child to access the sexual encounter. This sexual desensitization tends to occur gradually. Usual physical or emotional contact, such as bathing, cuddling, or tickling, can eventually become sexual contact and then possibly more intrusive forms of sexual abuse. In fact, almost two-thirds of the children in the study indicated that at first, the genital contact seemed accidental. It should be noted, however, that some of the victims pointed out that the change from usual physical contact to sexual abuse was abrupt and, therefore, the period of gradual sexual desensitization was small or nonexistent. Attackers also endorsed the use of tactical sexual desensitization. In this study, it was evidenced that around a quarter of the attackers who care for their victims admitted having used these grooming techniques. Besides, almost a third of the attackers admitted to having asked the child for help with something, such as undressing. Almost half admitted having talked about sex with the child or having "accidentally" touched the child. Attackers also admitted to using pornographic videos and magazines to desensitize the child to sex.

It is worth noting that the use of pornography in children with sexual insensitivity is more common among male victims than among women. The offender may tell the child that he is teaching him/her sexual education using photographic resources and the body of the victim. The research also emphasizes that attackers gradually increase physical contact. For example, the offender can start fighting, kissing, massaging, or curling up the child, all while evaluating the child's reaction to touching. If the child feels uncomfortable and asks the offender to stop, he may stop for a moment and then gradually increase the contact. The use of games, for example, *Red Light-Green Light* is also used for this purpose. In this situation, the offender may begin to touch the child's leg until the child protests. Other conventional techniques that the offender can use to desensitize the child is to "accidentally" show his naked body to the child, making sexual comments about the child's body or clothes, or telling him about previous sexual encounters that he or she has had.

Rutgaizer *et al.* [20], justify and assure that in the scientific field, there have been few investigations to understand the behavior patterns of sexual attackers in the different stages of online child harassment. In these stages, we observe the development of deceptive confidence, preparation, and the search for a physical encounter. In this research, characterizing the stages becomes a highly critical aspect, since most of the sexually abused children have been forced to accept physical encounters with the sexual attacker voluntarily. This suggests that understanding the different strategies that an attacker uses to manipulate children's behavior could help to educate them if they are exposed to these types of situations where their integrity is at risk. The research is developed based on Olson's Luring Communication Theory (LCT) [21], where once an attacker has had access to a child, the stages are:

- **Development of deceptive confidence:** consisting of developing a relationship of trust with the child. At this stage, the attacker exchanges personal information such as age, tastes, and distraction activities. This stage allows the attacker to build a shared communication space with its victim. Once a relationship of trust is established, the attacker proceeds to the next stage;
- **Grooming stage:** in this stage, the attacker triggers the sexual curiosity using sexual terms; it is at this moment where the attacker can prepare and catch communicatively the child in an online sexual behavior;
- **Entrapment cycle:** as the grooming process intensifies, the attacker oversees manipulating the victim so that she isolates herself from her friends and family, which promotes and increases the trust relationship between attacker and victim;
- **Physically approach:** in this final stage, the attacker seeks to approach to the minor. The attacker requests information related to the child's location and schedules and their family members.

In [22], Hofman et al. describe 17 descriptors of the grooming process within six stages of work:

- **friendship:** the attacker tries to approach the child by determining similarities, tastes and activities in common, and on the other hand the attacker searches based on photographic evidence requested from the minor and alternative online methods that contrast the information with the child, to confirm that he is a child.
- **relationship:** the attacker and the child talk about the family, the school, the interest and the hobbies of the child in order to exploit them deceptively, making the child believe that they are in a relationship.
- **risk assessment:** in this stage, the author attempts to measure the level of threat and danger by talking to the child. He makes sure that the child is alone and that no one else is reading their conversations.
- **exclusivity:** the attacker tries to gain the child's full confidence. Frequently, the attacker introduces the concept of love and care in this stage.
- **sexual:** the attacker and the child talk about sexual activities and develop sexual fantasy.
- **conclusion:** at this stage, the attacker approaches the child to meet in person.

It is worth noting that the researchers describe that these grooming stages may or may not occur in the same sequence. The frequency, order, and extent of the occurrence of these stages may vary depending on the case.

B. TOOLS AND TECHNOLOGIES TO DETECT GROOMING

Figure 1 describes the diverse tools used to detect grooming. The numbers correspond to a sample of studies described in Table 1. These tools are classified into:

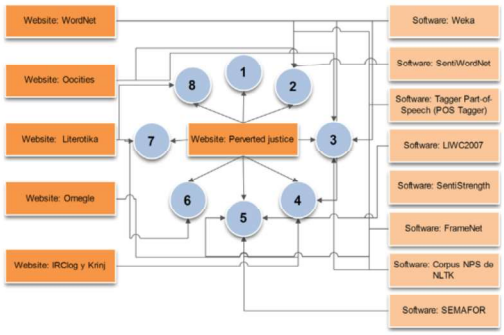


FIGURE 1. Graphic scheme that relates a group of numbered researches (more related to research). All researches use data from Perverted Justice and process the information with a set of technological tools based on their objectives.

TABLE 1. Research related to the study of grooming.

Item	Research	Reference
1	Toward Spotting the Pedophile Telling victim from attacker in text chats	[2]
2	On the Impact of Sentiment and Emotion Based Features in Detecting Online Sexual Attacker Dasha	[9]
3	modeling Fixated Discourse in Chats with Cyber-pedophiles	[27]
4	"Our Little Secret": pinpointing potential attacker	[28]
5	Detecting Child Grooming Behaviour Patterns on Social Media	[14]
6	Exploring high-level features for detecting cyber-pedophilia	[29]
7	Logistic Models for Classifying Online Grooming Conversation	[19]
8	Detecting Online Child Grooming Conversation	[30]

- **Website**.- The investigations related to the field of pedophilia are based on the collection of real conversations of pedophiles for further analysis. PJ Foundation is the main provider of this information. In the analysis of the scientific proposals to detect grooming it could be determined that when creating artificial intelligence algorithms, these should have the minimum error rate, which is why the researchers contrasted their models with chats of pedophile and non-pedophile sexual content. These chats were obtained from websites such as Oocities, IRClog, Krinj, among others.
- **Software**.- The use of specialized software has allowed researchers to extract positive and negative words to analyze the technical profile, behavioral patterns, applied discourse, sentiment analysis, semantic analysis of attackers among others. The use of software specialized in artificial intelligence allowed the researchers to execute and test some algorithms of automatic learning and data mining.

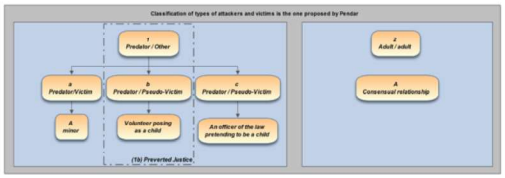


FIGURE 2. Classification of types of attackers and victims [2].

- **Corpus or Database**.- NPS Chat Corpus is a closed set of texts or information intended for scientific research and is part of the Natural Language Toolkit package (NLTK). NLTK is a natural language processing platform that allows researchers to build programs with human language data and thus generate predictions of conversations and behavioral patterns.

III. PSYCHOLOGICAL/TECHNICAL ATTACKER PROFILE

For the study of psychological profiles based on the use of technological resources such as the Internet, one of the pioneering investigations in the classification of types of attackers and victims is the one proposed by Pendar [2] (see Figure 2). In this study, two types of scenarios and their actors are considered. The first scenario is where Attacker/Other (1) are interrelated and the second Adult/Adult (2) where there is a consensual relationship. Three types of actors emerge from the first scenario: (a) Attacker/Victim where the victim is a minor. (b) Attacker/Pseudo-Victim in this case the victim is a volunteer posing as a child and (c) Attacker/Pseudo-Victim where the victim is an officer of the law pretending to be a child.

Ideally, to build a computer system that signals an interaction as suspicious, that is of type (1a), at least it is necessary to have access to representative samples of type 2 interactions also like (1a). However, this research indicates that chat service providers do not usually archive chats files for adults, and even if they did, they would not make those files available to the public. The accumulation of such data requires the informed consent of the participants. In addition, access to chat text files of type (1a) is also very difficult to achieve. Obtaining access to the data types (1c) is not without problems, since legal problems must be resolved in terms of privacy. Therefore, even a simple feasibility study for this type of research proposal faces major problems of data acquisition considering that none of which is necessarily technical. Given these difficulties, the best option is type (1b) interactions that are available online [27], [28].

The website www.perverted-justice.com, which is run by a group of volunteers, aims to make it difficult for pedophiles to take underage victims online. On this website, volunteers are recruited to pretend to be minors (usually from 10 to 15 years old) in chat rooms. When a pedophile has been found, the website publishes online files of all chats with them. In this research, the authors decided to use the aforementioned data to evaluate the feasibility of developing a

computer system to perform the automatic recognition of sexual attackers online with the use of type (1b) text records. There, it was evidenced that they managed to distinguish automatically between the pseudo-victim and the attacker, with the assumption that a positive result would support the hypothesis that it is possible to mark suspect chats online automatically [29], [30].

In [25], Bogdanova et al. determined certain distinctive features of pedophiles online, where around 94% were men who mostly had feelings of inferiority, isolation, loneliness, low self-esteem and emotional immaturity. From this group of criminals, between 60% and 80% suffer from other psychiatric illnesses. In general, pedophiles are less emotionally stable people than mentally healthy people. The research referenced by Hall et al. in [3] classifies male pedophiles as

- **homophilia:** if they are only attracted by male children;
- **heterosexual pedophilia:** if they are attracted by girls; and
- **bisexual pedophilia:** if they are attracted by girls and boys.

It also refers to five types of attackers [3]:

- **stalkers:** who approach children in chat rooms to gain physical access to them;
- **cruisers:** who are interested in online sexual abuse and are not willing to meet children off-line;
- **masturbators:** who watch child pornography;
- **networkers or swappers:** who exchange information, pornography and children;

and a combination of the four types. According to their study, the percentage of homosexual pedophiles varies from 9% to 40%. The researchers point out that the percentages indicated above do not imply that homosexuals are more prone to attack children, only that a greater percentage of pedophiles are homosexual or bisexual in orientation towards children. As important aspects of this research, the relationship between the age of the victims in relation to the sexual preference of the attacker and the average number of sexual acts is highlighted. Heterosexual male pedophiles prefer children between the ages of 8 and 10 years and on average have performed 34 sexual acts. Homosexual male pedophiles tend to prefer children between the ages of 10 and 13 and on average have performed 52 sexual acts [9], [31]. Regarding bisexual pedophiles, it is only observed that on average they have committed more than 120 acts. In [3], Hall et al. reference a study focused on the incestuous pedophile attacker where the following results were determined: 27% of all sex offenders assaulted family members. 50% of crimes committed against children under 6 years were committed by a family member, 42% of acts committed against children from 6 to 11 years and 24% against children from 12 to 17 years. An additional study indicates that 68% of child abusers had sexually abused a family member; 30% had sexually abused a stepchild or adopted child; 19% had bothered one or more of their biological children; 18% had bothered a niece or nephew; and 5% had sexually abused a grandchild. In this study incestuous heterosexual pedophiles had abused 1.8 children and committed 81.3 acts, while incestuous homosexual pedophiles had abused 1.7 children and committed 62.3 acts.

With regard to the relationship of victims and attackers, Hall's research [3] makes a first distinction:

- **exclusive pedophiles:** pedophiles only attracted to children; and
- **non-exclusive pedophiles:** pedophiles attracted to both adults and children.

In this study, the authors determine that most pedophiles are part of the non-exclusive group.

On the other hand, Vartapetiance et al. in [24] identify three types of potential attackers [32]:

- **strange:** who is a completely unknown person. This type of attacker does not necessarily want to have long-term access to children. Therefore, to attract children, they are more likely to use threats.
- **known:** which can be teachers, drivers, among others. This type of offender generally has access to children; however, they do not use violence to attract them. They invest a lot of time to create the trust relationship to decrease the likelihood of being identified.
- **family:** such as parents, grandparents, cousins, and siblings. This offender generally has long-term access to children, because he is within the family circle and uses his authority to control the children. This type of relationship is the most dangerous due to the time the abuse may last.

In [4], Bogdanova et al. revealed several language characteristics of attackers based on pedophile conversations through chat: implicit / explicit content. On the one hand, the attackers gradually change the context of a conversation until they get an openly sexual conversation without hiding their intentions. First of all, they start with comfortable talks for the victim, which are accompanied by compliments, childish behavior and jargon. Another characteristic evidenced is the fixed discourse, where the attackers use various conversational strategies to avoid departing from the sexual conversation obtained. On some occasions these attackers manipulate their victims by transferring responsibility and blaming them for any differences or disagreements they have had. To minimize the risk of being prosecuted before the law, some attackers force their victims to eliminate chat conversations or records that have been generated. However, it has also been shown that some attackers cease to be cautious and insistently request physical encounters without measuring the consequences [33].

As a summary of the studies related to the understanding of cyber-attackers, in the scope of pedophilia, Figure 3 outlines the psychological/technical profile.

IV. RELATED WORK

There are many works related to online pedophilia and grooming, which concern machine learning paradigms. In 1997, Durkin [5] determined, in one of his researches, that

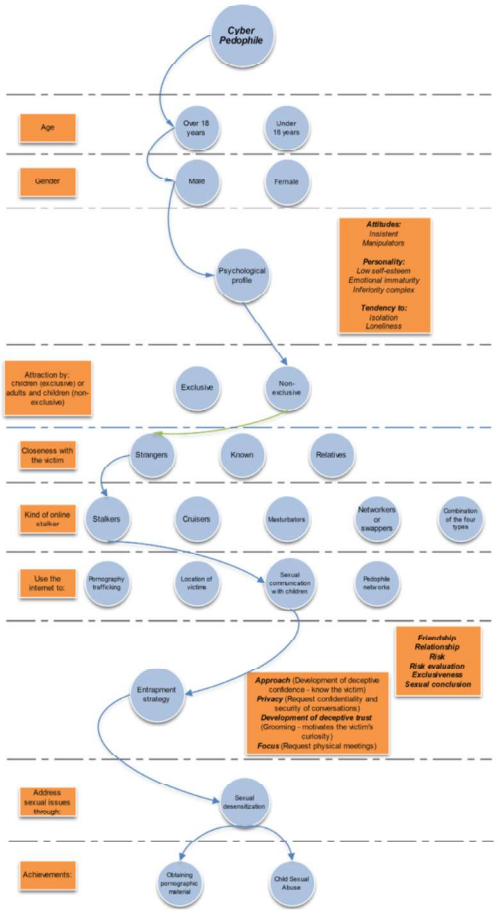


FIGURE 3. Psychological/technical profile of a cyber-pedophile.

one field of study related to online pedophilia, is the "location of victims through chat rooms." From this, the author raised several important contributions with the participation of PJ Foundation. The content provided by this agency allowed the application of natural language processing techniques, artificial intelligence and other technological tools associated with machine learning. The results related to the text analysis has allowed scientists to determine certain features of psychological behavior of attackers in relation to the use of technological tools to access their victims [12], [26], [34].

In [23], Bogdanova et al. address the problem of detecting pedophiles with Natural Language Processing techniques (NLP) and the naive bayes and support vector machines classifiers. This problem becomes even more challenging due to the specificity of the chat data. Chat conversations are very different not only from the written text, but also from other types of interactions in social networks, such as blogs and forums, since chat on the Internet usually involves very fast writing. The data usually contains many errors, spelling mistakes, specific jargon, character flooding, among others. The authors also point out the complexity at the time of processing the data with automated parsers. They include a list of features, which includes feelings and other characteristics based on the content. In their experimental results they describe that the classification based on their characteristics can discriminate pedophiles from non-pedophiles with great precision [13], [24], [25].

In a later investigation, Bogdanova *et al.* [23] propose to model the obsessive discourse of an attacker using lexical chains as a potential feature in the automatic detection of online sexual attackers throughout the conversation. To estimate semantic similarity, they used two parameters: the similarity of Leacock - Chodorow and Resnik. In their results they show considerable variation in the length of lexical chains related to sex according to the nature of the corpus or database. The lexical chains related to sex in the NPS corpus are much shorter, regardless of the similarity of the measure used. The chains in the corpus cybersex are even longer than in the corpus of PJ Foundation. With this premise, they support their hypothesis that this could be a valuable feature in an automated pedophile detection system.

In [13], Cano et al. used a collection of features that aim to characterize attacker conversations in stages of online preparation through the profile of an attacker based on the characterization of: 1) bag of words (BoW); 2) syntactic; 3) polarity of feeling; 4) content; 5) psycho-linguistic; and 6) speech patterns. The main contributions of this article are: (1) proposal of an approach to automatically identify the stages of preparation in an online conversation based on multiple characteristics: lexicon, syntactic, feeling, content, psycho-linguistics; and patterns of discourse. (2) Classification models for each stage, using unique and multiple characteristics. For the generation of the models, they use several software tools, and the use of the child preparation stages proposed by Olson. In their findings, the authors show that the use of the characteristics of the speech pattern alone can achieve on average a gain on the lexical characteristics. (3) Analysis of particularities to identify the most discriminatory characteristics in each stage of grooming.

In [25], the authors suggest a list of high-level features and study their applicability in the detection of cyber-pedophiles. For this purpose, they used a corpus of downloaded chats from PJ Foundation and two sets of negative data of a different nature: cyberspace records available online and the NPS chat corpus. In their analysis, the authors consider that lexical chains are appropriate for modeling the obsessed speech of pedophile chats. To find semantically related terms, they used parameters of semantic similarity. In particular, the similarity of Leacock and Chodorow and the resemblance of Resnik. The results of the research show that NPS data and pedophile conversations can be accurately discriminated against each other with n-grams (characters), while in the

more complicated case of cybersex records high-level characteristics are needed to reach good levels of precision.

Pranoto et al. in [12] try to establish a mathematical logistic model to classify whether an online conversation is a preparation conversation or not. For this purpose, the authors analyzed approximately 160 chat conversations to determine the characteristics of a preparation conversation. These scripts are obtained in a random way from http://www.perverted-justice.com and www.literotika.com. The characteristics are divided into 20 types. The scripts are divided into two sets: 100 scripts for the training set and 59 scripts for the test set. As a result of the research, five most relevant grooming characteristics were identified, and a logistic model was established on this basis. The model is evaluated using the test data set and the results show that the model has acceptable results by the authors.

All the papers present proposals for the analysis of chat conversations with childish pornographic content. It considers different topics that have enable to outline the behavior of an attacker [35]. These topics have addressed aspects of feelings, characteristics based on content, modeling of obsessive discourse using lexical chains, among others. On the other hand, the stages of grooming have been analyzed in syntactic aspects, polarity of feeling, content, psycho-linguistic and discourse patterns.

V. RESEARCH METHODOLOGY

Aiming at determining the life cycle of grooming, we will define stations or phases named *topics*. In the field of text mining is the topic modeling, which allows to analyze a large number of unstructured texts. There are several methods of topic modeling, among the most relevant are Latent Semantic Analysis (LSA), Probabilistic Latent Semantic Analysis (PLSA), Correlated Theme Model (CTM), and LDA. In related literature that compare the performance of LDA with other models in terms of perplexity, it is determined that the performance of LDA is higher than that of other models. Also, it is established that LDA could be applied successfully in various applications aiming at identifying topics in scientific publications, text classification and collaborative filtering [36]–[39]. Under these criteria and based on the nature of our study (text categorization), we decided to use LDA as topic modeling.

First, the LDA generative statistical model is proposed; it allows the modeling of topics. Based on this, two experiments were carried out. In the first experiment, several topics are determined according to the characteristics of the pre-processed data. To obtain the data and its processing, the recommendations of the CRISP-DM methodology were followed [16]. After determining an optimal number of topics, we proceeded to give them a logical context through experiment 2. It uses several studies concerning linguistics and communicational intentions to order the topics determined by LDA. Within this ordering, several proposals of life cycles of APT with the topics were related, thus determining the life cycle of the grooming.

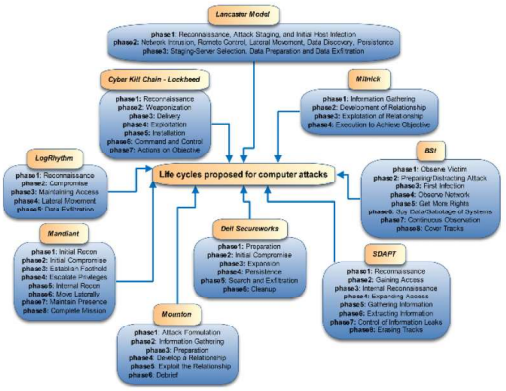


FIGURE 4. Life cycles proposed for computer attacks [43], [48], [49].

A. COMPUTER ATTACK EVALUATION

A computer attack represents any hostile activity against a system or a person, using computer applications or psychological persuasion techniques. Every attack has a target, and the responsibility of scientists is to determine what they are in order to apply defense strategies. It is worth noting that computer attackers are aware of the development and execution of each attack by developing a series of phases, stages, or steps to follow to make an attack successful [40]–[43].

To identify the effects of each attack, they should be interpreted not only as isolated incidents or intrusions but also as operations that, in some cases, contemplate long periods. The stages of a computer attack are represented by models of life cycle applicable to cyber-attacks as illustrated in Figure 4; they are also known as "cyber-attack chains" [44]–[46]. In the scientific field, several authors take as reference the life cycle approach of Lockheed Martin [47], who developed an initial model of cyber-attack chain. Under this criterion, the main contribution of this study is the theoretical/practical definition of the life cycle of grooming, from the point of view of information security.

B. RESEARCH QUESTIONS

Formalizing the concept of grooming within the field of information security will allow researchers to support future research related to social engineering with the contributions generated with grooming. To achieve these objectives, the following research questions are formulated:

- RQ1: With the use of computer learning, can the phases of grooming be determined as a computer attack?
- RQ2: Can grooming be considered an attack vector within the APTs?
- RQ3: Can the studies related to grooming support future research associated with social engineering?

C. EXPERIMENTAL APPROACH

Figure 5 describes our experimental approach in two stages. In the first stage, we pre-process the data obtained from

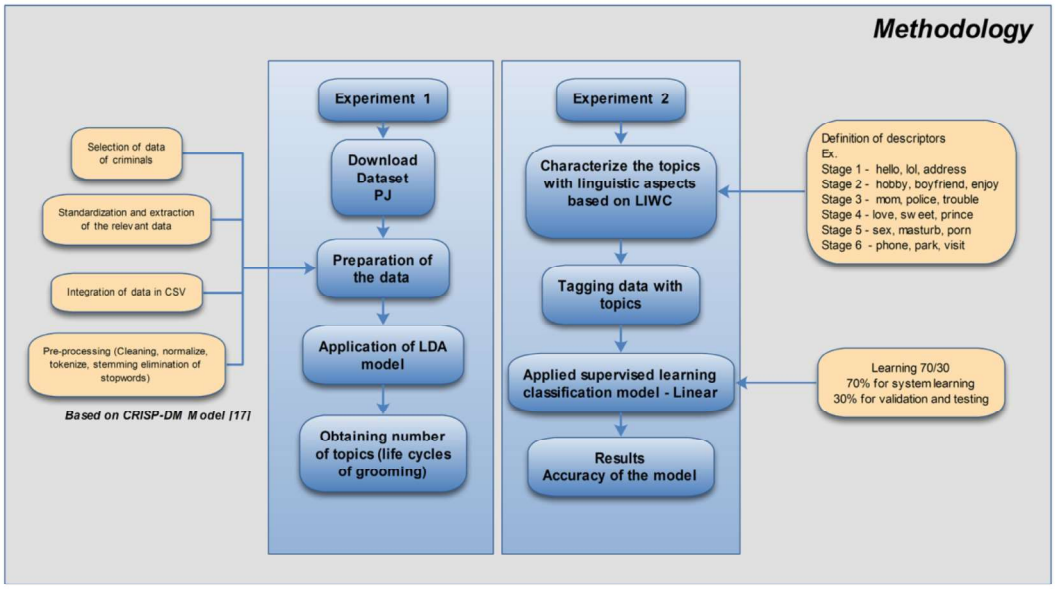


FIGURE 5. Methodology applied to determine the life cycle of grooming.

PJ, then applied a topic modeling (unsupervised learning paradigm) that allowed us to analyze several phases, stages or steps herein *topics*. Due to the nature of the system development, it should be noted that, in this assignment of topics, linguistic analysis is not previously carried out. After this, we contrasted the results obtained with statistical modeling to justify the exact number of topics that will be applied to grooming. Once this information is obtained, we analyzed if these topics correspond to the different stations of the life cycles proposed in the analysis of computer attacks. In the second stage, we assigned a linguistic approach using a set of word categories provided by the Linguistic Inquiry and Word Count software (LIWC) at each stage determined with LDA. LIWC is a program that analyses text. It reads a given text and counts the percentage of words that reflect different emotions, thinking styles, social concerns, and even parts of speech. Once the stages were determined, the system was trained with a linear classification model (supervised learning paradigm) to determine the accuracy of the system.

In both experiments, quantitative and qualitative characteristics were adopted (linguistic assignment); however, in the first experiment, the results obtained by the software were justified with statistical analysis. Within the first experiment, the data was pre-processed with the development of scripts with regular expressions to standardize the format of the data. The treatment of the information was applied to 100 chats of pedophile character with an average of 1200 lines of text per chat, processing a total of chat lines of 128171. The number of chats was determined based on the average of conversations analyzed in similar investigations and rejecting short content conversations.

VI. EXPERIMENTATION

From the scientific method, reproducibility is an essential aspect to be considered. Therefore, each of the phases of the proposed methodology is detailed in the experiments carried out. The data, hardware, and software resources used in the experimentation phase are described in Table 2.

A. EXPERIMENT 1

In this section, several aspects that were considered in the realization of the first experiment are addressed. Within these aspects, the obtaining and processing of the data, the application of the LDA model, and the life cycle of grooming are explained.

1) DATA COLLECTION

100 conversations (128171 chat lines) were downloaded individually in HTML format. The download in this format allowed to use its components (labels) for the pre-processing of the data. The process that was conducted for data collection was:

- **Dataset Download from PJ:** As previously mentioned, 128171 lines of chats downloaded from PJ were used. These records were generated between attackers and pseudo-victims, and their selection was performed based on the representativeness of the data with a manual download for further analysis. The number of records

TABLE 2. Materials used in experimentation.

Hardware Resources		
Resource	***Description***	***Version***
Processor	Intel Core i5-2320 @3.00 GHz	N/A
Installed memory (RAM)	10,0 GB	N/A
OS	64 bits, x64 processor	N/A
Software Resources		
Resource	***Description***	***Version***
Windows	It is an operating system produced by Microsoft as part of its family of Windows NT operating systems.	N/A
Perl	It is a programming language suitable for writing simple scripts and complex applications	Perl 5 Version 28 Sub version 1 (v5.28.1)
MATLAB	It is a numerical computing environment with multiple paradigms and a programming language developed by MathWorks.	R2019a Version 9.6
Text Analytics Toolbox	It is a MATLAB module that provides tools to perform data mining and machine learning	R2019a Version 9.6
Microsoft Excel	It is a spreadsheet developed by Microsoft that has graphic tools, dynamic tables and a programming language for applications.	Excel 2017 Version 15.0
Data Resources		
Resource	***Description***	***Version***
Perverted Justice	It is the largest virtual repository of grooming conversations publicly available for analysis.	http://www.perverted-justice.com

represented the average of data used in related investigations. Conversation records were downloaded individually in HTML format. This format will allow better processing of the data for further analysis.

- **Description of the data:** The superficial properties of the acquired data were examined, and the number of message lines contained in each chat and the number of words contained in each message line were determined.
- **Data exploration:** In this phase, it was identified that all the conversations had a common structure made up of four components: name of the sender (attacker or pseudo-victim), time stamp, message, and annotations of the pseudo-victim. From this analysis, it was determined that the essential components for the proposed study are the name of the sender and the message. The components such as timestamp and annotations were not considered in the study.
- **Verification of data quality:** Determining the quality of the acquired data was challenging because the chats come from various sources, use informal language, the vocabulary consists of slangs, shorthand, emoticons, and contain spelling errors. For this reason, further purification was required, which will be described in the cleaning and pre-processing section.

2) DATA PROCESSING

Within this stage, exploration and verification of the data to be analyzed was carried out. In a previous analysis, it was possible to identify a typical structure in them, which is composed of four parts: the name of the sender (attacker or pseudo-victim), the time stamp, the message, and the annotation (optional description). It is determined that two of the identified parties are essential for the analysis; these are the name of the sender and the message. The remaining two parts, which are the time stamp and the optional annotations, are not relevant to the present case study. The verification of the quality of the data was a challenging aspect given the nature of the chats, because they come from different sources and the language used in the conversations is extremely informal to contain slangs, shorthand errors, emoticons, and misspellings. Therefore, the data requires several pre-processing and cleaning steps before the analysis can be performed:

- **Data selection - Attackers only:** The decision on what data should be used for the analysis is based on several criteria, including their relevance to the objectives of data mining, as well as technical and quality limitations, limits on the volume of data, and types of data [50]. As described in the previous sections, the data records for this case study are conversations between attackers and pseudo-victims (undercover agents). After analyzing the dialogues, it is evident that the attackers lead the conversations and choose the topic of discussion; most of the time, they force the pseudo-victims to answer unethical questions. The pseudo-victims, in most cases, follow the topic of the conversation with typical answers like "yes", "no", "maybe", "we will see", among others. Therefore, to effectively analyze the grooming, it was determined that only the messages of the attackers would be analyzed.
- **Standardization and extraction of relevant data:** To extract only the required data, a distinction was generated between the attacker's messages, the pseudo-victim's messages, and time-stamps, making use of the HTML tags. Using perl-based scripts and regular expressions, only the data relevant to the investigation was filtered.
- **Integration of the data in a CSV file:** By having several files of independent conversations, a unified structure is created from the extracted data; this structure includes all the messages coming from the attackers. For doing so, the data is grouped into a file of type CSV, which is constituted as the base structure that contains all the text data to be treated later.
- **Pre-processing:** In the pre-processing stage, two intermediate threads called cleaning and text standardization are performed. For cleaning the text, punctuation marks and special characters and words that add noise to the study are eliminated. These words are known as stop words and are all those articles, prepositions, conjunctions, pronouns, among others, that do not add meaning to the investigation. After cleaning, the standardization stage is performed where all the text is lowercase, the verbs are taken to their base form, for example "getting" to "get," lemmatization and normalization techniques are applied, to finally eliminate all words that have 2 or less characters or that exceed 15 characters.

The obtained results are illustrated in Figure 6.

FIGURE 6. Word cloud of pre-processed and processed text.

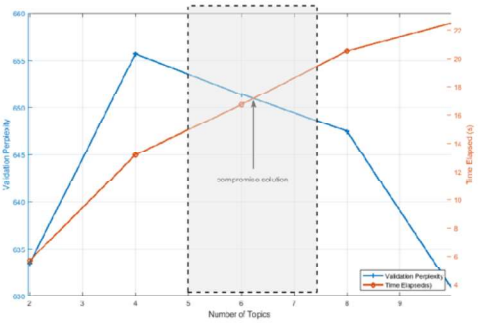


FIGURE 8. Topics assigned to grooming by LDA.

FIGURE 7. Word cloud by topic - LDA model.

3) APPLICATION OF LDA MODEL

LSA [51] and LDA [15] are widely used in NLP applications for similar tasks. These methods use semantic distances or similarities/relationships between terms to form cliches or word chains. LSA and LDA use the joint frequency of the concurrency of words in different bodies, and links between them to find closely related words. Although these methods can be used in a similar way for several NLP tasks such as text summary, answer to questions or topic detection, each one uses different measures and has different meanings. LDA generates topical threads under an earlier Dirichlet distribution, while LSA produces a correlation matrix between words and documents. Under this consideration, LDA has been taken as reference for the determination of topics.

Tests of the LDA model: As a first step, the LDA model required a dataset sectioning (90% - 10%) to evaluate the quality of adjustment (perplexity vs. time) and be able to determine an optimal number of topics. Note that, in the application of the LDA model sectioning was not required, the model was adjusted to the number of defined topics, through the creation of a bag of words with unigrams, obtaining a classification of words by topic as can be seen in Figure 7.

4) OBTAINING NUMBER OF TOPICS (LIFE CYCLE OF GROOMING)

To determine the number of standard topics of grooming, in contrast to the life cycles of computer attacks described in Figure 4, we proceeded to choose a range of values that contains several numbers of topics, and in its analysis determine an optimal compromise solution based on the perplexity and processing time in the application of the model. To demonstrate the effects of the compensation, the quality of adjustment, and the adjustment time are calculated. If the optimal number of topics is high, a lower value can be chosen to speed up the adjustment process and determining the most appropriate number of topics allowed. The range considered to determine an optimal compromise solution started with two (topics proposed by Lancaster) and ended with eight topics proposed by Mandiant, BSI and Sdapt [43].

The pre-processed CSV dataset was required to execute the LDA algorithm. The algorithm itself required that the dataset be divided into two groups to train and validate the model. In our case study, 90% for training and 10% for validation. The model by its natural defined two groups of bags of words with unigrams.

The optimal value result determined that the number of topics suitable for the analysis of the life cycle of grooming is six, as illustrated in Figure 8. As can be seen, the perplexity and the time elapsed for this number of topics is reasonable. Besides, it can be deduced that an increasing number of topics leads to a better adjustment, but adjusting the model takes longer to converge. As additional data, it could be determined that two additional theoretical topics are not testable through the dataset since the first would define the way attackers look for their victim and as a second topic is the demonstration of the mechanisms or associated techniques to maintain contact after performing sexual encounters. This is because the pseudo-victims cease to have communication with pedophiles once they pose fortuitous encounters.

Operation of the LDA model with 5 text lines from the dataset: To analyze the functioning of the LDA model, we proceeded to evaluate it with 5 lines of text from the dataset: 1.- "love give massage", 2.- "nice warm lotion

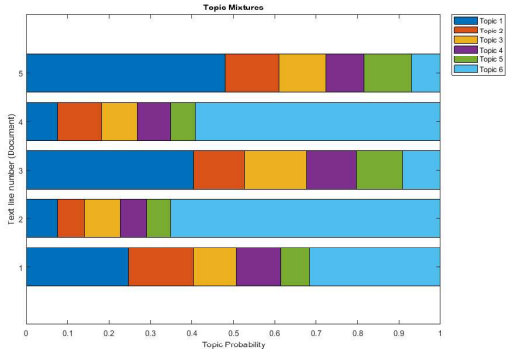


FIGURE 9. Analysis of the operation of the LDA model.

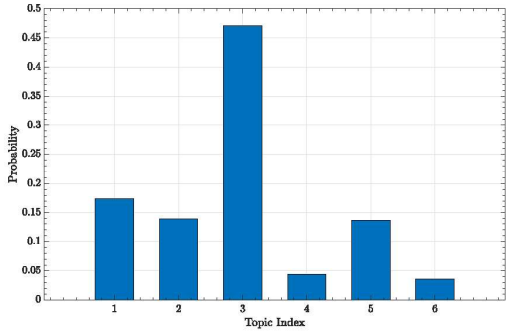


FIGURE 10. Document topic probabilities using the LDA model.

body", 3.- "love", 4.- "give nice ass rub", and 5.- "yeah why". In Figure 9, we can observe the mixture of the six topics present in each one of the lines of text analyzed with the LDA model. It is also observed that, for each analyzed document, the probability of a topic stands out above the others, this allows to infer the belonging of the document to a particular topic, understanding a document as a text message and each topic as a phase of grooming.

Operation of the LDA model with test text independent of the dataset: To test the accuracy of the unsupervised LDA model, a separate string of text was created from the data of the dataset ("this will be our little secret... do not tell your parents about me... I can get in trouble") obtaining the result depicted in Figure 10. In the bar chart, it is observed that in the text used, there are multiple mixtures of topics, and the highest probability of belonging for this text is found in topic number 3.

B. EXPERIMENT 2

In this section, the number of topics determined with linguistic characteristics is related. In this way, a supervised classification model will be applied, and it will be finalized with the analysis of the results and its accuracy.

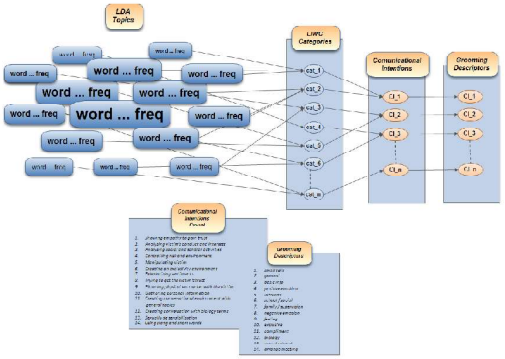


FIGURE 11. Process to characterize and define the stages of the life cycle of grooming.

1) CHARACTERIZING THE TOPICS WITH LINGUISTIC ASPECTS LIWC

As illustrated in Figure 11, part of the process that will characterize and define the stages of the life cycle of grooming, as a first phase it is required to combine the topics obtained with the application of the LDA model, with the categories linguistics provided in [52] and obtained from the LIWC software.

Proposal of descriptors based on communicational intentions: Applying the LDA model with 6 topics grouped words with their respective frequency within each topic; however, they are only words and require a linguistic process, in order to give a meaning to each topic. Based on Figure 11, we will explain the process of characterization of the text of a topic with linguistic aspects. This process was replicated in the rest of the topics as follows

- Obtaining words frequently from each topic;
- The linguistic categories provided by LIWC were compiled as depicted in Figure 12;
- Each word was placed within one or more linguistic categories;
- With the linguistic characteristics obtained, it was possible to infer the communicator intentions of the attacker in the selected topic;
- In the process of determining the "communicational intentions" we observed that in some cases these were repeated, due to the nature of the attack. With this background, we created a structure of descriptors that conceptualized each communal intention formulated;
- To finish with the formalization stage of the descriptors, we codified them. It is worth noting that these descriptors are specific to the analysis of grooming and that later they will be used to label the data set, a step prior to the application of a supervised learning model.

2) TAGGING DATA WITH TOPICS

With the use of the communicational intentions and the descriptors based on the LIWC categories, within our context,

LIWC Categories	Language examples
You	You
Friend	Friend, boyfriend, girlfriend, lover
Social	Adult, anyone, party, outsider, right, story, phone, private, public, gossip
Work	Homework, office, school
Leisure	Art, bands, game, hangout, sport, television, movie

FIGURE 12. Linguistic categories LIWC.

Topics (Stages)	Grooming Descriptors		Communicative intention
	Code	Name	
Topic 1	D11	Basic info	Acquire basic information of the victim.
	D12	Slang	Emphasize the social and contextual understanding of the victim with the use of adolescent jargon.
	D13	Small talk	Distract the victim by creating conversation about unimportant or uncontroversial issues.
Topic 2	D21	Interests	Establish a link with the victim when talking about their personal interests.
	D22	School / social	Acquire specific information of the victim, related to his friends, family, school and social life.
	D23	Positive emotions	Show compassion and understanding to gain the confidence of the victim.
Topic 3	D31	Family / supervision	Inquire about the location, the parent's schedule and the victim's supervision.
	D32	Negative emotions	Ensure the silence of the victim by describing the consequences of revealing the nature of their relationship
Topic 4	D41	Exclusivity	Establish an exclusive relationship with the victim.
	D42	Feelings	Express feelings of love, care and confidence.
	D43	Compliments	Adulate the victim to maintain and increase the level of trust.
Topic 5	D51	Biology	Desensitize the victim in the sexual theme, using biological terms.
	D52	Sexual related terms	Detail the sexual acts you want to perform with the victim or past sexual experiences.
Topic 6	D61	Meeting arrangement	Plan a personal encounter with the victim.

FIGURE 13. Assigning codes to data.

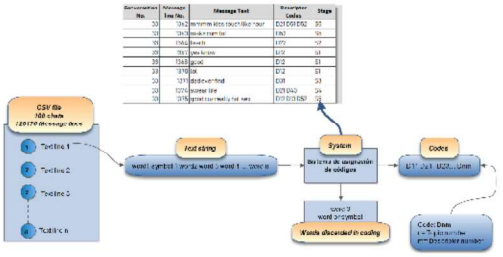


FIGURE 14. Process of data labeling.

TABLE 3. Grooming station.

Communicative Intention	Grooming Stage	
	Definition	Number
Acquire basic information of the victim.	The attacker makes contact and gets to know his target. He uses short talks to subtly gather general information about the victim, such as his age, gender and interests. This stage can be restated several times depending on the level of contact the attacker has with the victim.	S1
Emphasize the social and contextual understanding of the victim with the use of adolescent jargon.		
Distract the victim by creating conversation about unimportant or uncontroversial issues.		
Establish a link with the victim when talking about their personal interests.	The attacker establishes a link with the victim by talking about his friends, family, school and social life. The attacker is compassionate and understanding to try to gain the trust of the victim.	S2
Acquire specific information of the victim, related to his friends, family, school and social life.		
Show compassion and understanding to gain the confidence of the victim.		
Inquire about the location, the parent's schedule and the victim's supervision.	The attacker begins to inquire about the location, the parent's schedule and the victim's supervision, using this information to determine the risk of being caught. It encourages the victim not to reveal the nature of their relationship with others and ensures the silence of the victim with various techniques.	S3
Ensure the silence of the victim by describing the consequences of revealing the nature of their relationship.		
Establish an exclusive relationship with the victim.	The attacker tries to establish a trusting and exclusive relationship with the victim. Affirms that they share a special bond. The concept of love, care and feelings in general are introduced.	S4
Express feelings of love, care and confidence.		
Adulate the victim to maintain and increase the level of trust.		
Desensitize the victim in the sexual theme, using biological terms.	When the attacker is sure that the victim trusts him, he becomes more explicit about his intentions. The attacker can ask about past sexual experiences and detail the sexual acts he wishes to perform with the victim. In this stage the sexual content is predominant.	S5
Detail the sexual acts you want to perform with the victim or past sexual experiences.		
Plan a personal encounter with the victim.	It is in this final stage that the attacker attempts a personal encounter with the victim, agreeing on a physical place and date to meet. Additionally, the attacker creates an environment for maintaining the relationship, which allows the attacker to evade any type of detection, subtly ending the relationship.	S6

we proceeded to assign the corresponding codes to the structure of each line of text. This is made up of one or several descriptors as shown in Figure 13. It should be mentioned that for the code to be successful the system, see Figure 14, must validate that each word contributes to a general idea of the line of text and belongs to the established categories otherwise the system will exclude from your labeling. Additionally, the process that was accomplished for the definition of the communicational intentions allowed to refine the delimitation and distinction of the different groups of topics that will be categorized in the following section.

Life cycles of a computer attack applied to grooming: With the purpose of analyzing grooming as a computer attack, we proceeded to verify the relationship of the most representative life cycles with the communicational intentions described in the research. In order to proceed with this phase, first operational concepts (definition) were determined that contemplate the communicational intentions and in turn were assigned a topic or station number (see Table 3).

Having determined a characteristic definition of each grooming station, once the communicational intentions have been identified, we proceeded to compare them conceptually with each of the stages of the different life cycles most recognized in the scientific field related to computer attacks within the field of information security as illustrated in Figure 15.

- In the analysis of the first grooming station, its correlation was identified with the first station of the life cycle of all the proposals examined. Being **gathering information**, of the SDAP model, which in concept was adapted more to our definition.
- In relation to the second station proposed, the definition found was between stations 1, 2, 3 and 5 of all the models. However, the second station **gaining access** of the SDAP model defined the station better.
- With regard to the third station, this was identified between stations 3, 4, 6, 7, 8 and 2 of the models analyzed. The fourth station of the Logrythm, **lateral movement**, indicates the characteristics of the communicational intention that determined this corresponding grooming station.
- When analyzing the fourth station, the heterogeneity of concepts associated with it was evidenced, being

Cyber Attack Life Cycles			Grooming Stages					
Author	No	Stages	S1	S2	S3	S4	S5	S6
Lockheed	1	Reconnaissance	x					
	2	Weaponization		x				
	3	Delivery			x			
	4	Exploitation					x	
	5	Installation						
	6	Command & Control (C2)				x		
	7	Actions on Objective						x
LogRhythm	1	Reconnaissance	x					
	2	Compromise					x	x
	3	Maintaining Access						
	4	Lateral Movement			x	x		
	5	Data exfiltration		x				
Mandiant	1	Initial Recon	x					
	2	Initial Compromise		x				
	3	Establish Foothold						
	4	Escalate Privileges					x	
	5	Internal Recon		x				
	6	Move Laterally			x			
	7	Maintain Presence				x		
	8	Complete Mission						x
Cyber Kill Chain®	1	Preparation	x					
	2	Initial Compromise		x				
	3	Expansion				x		
	4	Persistence			x			
	5	Search and Exfiltration					x	
	6	Cleanup			x			
SDART	1	Reconnaissance						
	2	Gaining access		x				
	3	Internal Reconnaissance		x				
	4	Expanding Access				x		
	5	Gathering Information	x					
	6	Extracting information					x	
	7	Control of information leaks			x			
	8	Erasing Tracks			x			
IBM	1	Observe Victim						
	2	Preparing/Distracting Attack		x				
	3	First Infection		x				
	4	Observe Network						
	5	Get More Rights				x		
	6	Spy Data/Sabotage of Systems			x			x
	7	Continuous Observation				x		
	8	Cover Tracks			x			
Mitnick	1	Information Gathering	x					
	2	Development of relationship		^		^		
	3	Exploitation of Relationship					x	
	4	Execution to Achieve Objective						x
Lancaster	1	Reconnaissance, Attack Staging, and Initial Host Infection	x	x				
	2	Network Intrution, Remote Control, Lateral Movement, Data Discovery, Persistence			x	x		
	3	Staging Server Selection, Data Preparation and Data Exfiltration					x	x
Mouton	1	Attack Formulation	x					
	2	Information Gathering	x					
	3	Preparation		x				
	4	Develop a Relationship			x	x		
	5	Exploit the Relationship					x	
	6	Debrief						x
Grooming Life Cycle			Gathering Information	Gaining Access	Lateral Movement	Escalating Privileges	Execution	Debrief

FIGURE 15. Operational concepts (definition) related to the communicational intentions.

stations 2, 3, 4, 5, 6, 7 of the different authors, those that coincided conceptually with this station. However, the fourth station of Mandiant, **escalate privileges**, was the one that in position and concept defined the station.

- Stations 4, 2, 5, 6, 3 of the analyzed models showed correspondence with station number 5 of grooming. In this way, Mitnick station 4, **execution to achieve objective**, characterized it more accurately.
- For the selection of the concept of the last station of the grooming, it was observed that almost in all the stations of the models were contrasted with the last or penultimate station. However, the phase proposed by Mouton *et al.* [49] is the best one that describes the end of the attack **debrief**, since in this station the attacker manages the mental state of his victim at his convenience with different strategies.

3) APPLIED SUPERVISED LEARNING CLASSIFICATION MODEL

For the selection of a supervised learning model classification technique, it is advisable to understand the nature of the problem. This is the case of linear classifiers, given that

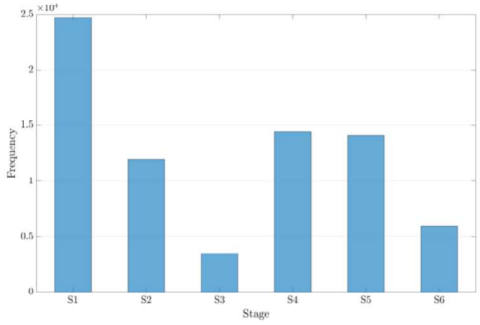


FIGURE 16. Class distribution histogram.

their simplicity and computational appeal are widely used in problems of automatic text classification, an integral part of our research [53], [54].

Another important aspect of this classification is its usefulness in machine learning and data mining consequently text mining. Unlike nonlinear classifiers, such as neural networks, which allocate data to a higher dimensional space, linear classifiers work directly on the data in the original input space. While linear classifiers cannot handle certain complex data types, they may be enough for textual content data. For example, linear classifiers have been shown to offer competitive returns on document data with non-linear classifiers. An important advantage of linear classification is that the training and testing procedures are much more efficient. Therefore, linear classification can be very useful for some large-scale applications [55]–[57].

Below is the process to train a linear classifier that is based on the word frequency count, through a bag-of-words model. Using it as a predictor of the stages of grooming to which a certain text message belongs.

For the application of the linear model the following steps were followed:

- The pre-processed CSV file is complemented with the labels of the stages of the life cycle of the grooming;
- The dataset is loaded in CSV format to MATLAB;
- A class distribution histogram is constructed to show the presence of each of the grooming stations in the dataset, see Figure 16;
- The data set is divided into 2 partitions for training and one set excluded for testing and validation (the training percentage was 70% and 30%, respectively);
- The classification model that takes as input a Bag-of-words model, which contains the pre-processed and labeled data, is constructed and trained;
- The classifier is tested to predict the labels of the test data using the trained model and then the classification accuracy is calculated, this being the proportion of labels that the model predicts correctly;
- An array is created with new data (text messages) to test the model.

FIGURE 17. Model accuracy and test with new data.

4) RESULTS AND MODEL ACCURACY

After applying the proposed steps sequentially, and applying data that are not known by the system, it describes that its accuracy was 97.61% (see Figure 17), thus confirming that the linear model was adapted without major problems to our case study.

5) COMPARISON OF THE PROPOSED LINEAR MODEL WITH DEEP LEARNING MODELS

Research related to the detection of cyber-pedophiles and grooming, support their studies with previous investigations related to artificial intelligence. In some cases, they review the literature of implemented algorithms and in other cases they propose new algorithms aimed at improving classification efficiency. Algorithms such as Support Vector Machine (SVM), Naive Bayes, Decision trees and k-nearest neighbor (KNN) and k-means clustering, have already been evaluated [7], [25], [26], [58]. Regarding the deep learning models applied, we tested 2 different models: A convolutional neural network (CNN), see Figure 18, and a Long short-term memory (LSTM) network, see Figure 19. The classification accuracies we obtained with these models were 96.11% for the CNN and 95.91% for the LSTM network. Based on these results, it is shown that the linear model is the best applied to our case study (classification of texts, see Section VI-B.3) since its accuracy is higher (97.61%) compared with that of the deep learning models.

VII. ANSWERING RESEARCH QUESTIONS

A. WITH THE USE OF COMPUTER LEARNING, CAN THE PHASES OF GROOMING BE DETERMINED AS A COMPUTER ATTACK?

The researches related to grooming have been analyzed from the psychological point of view, this aspect not being supported by agreements of the scientific community allows researchers to determine different phases with a high degree of subjectivity. For this reason, the research was based on a statistical model LDA that allowed to determine a specific number of stations or phases that attackers follow when applying this attack to their victims. With the application of computational learning it became evident that it is feasible to determine if a text belongs to a specific station with a high degree of accuracy.

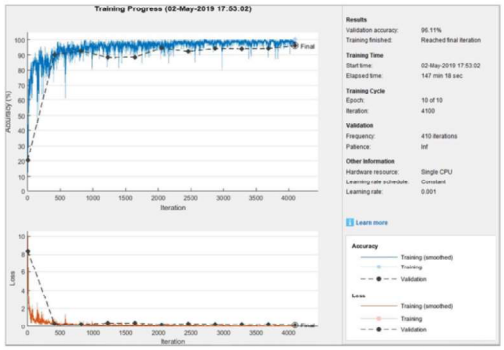


FIGURE 18. CNN network training.

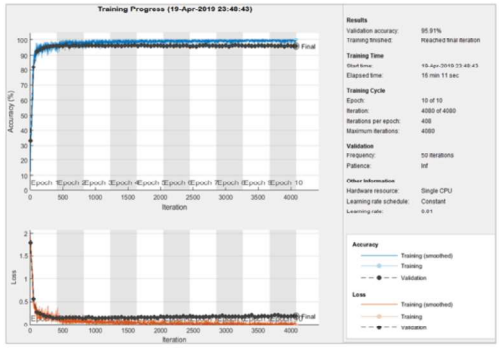


FIGURE 19. LSTM network training.

B. CAN GROOMING BE CONSIDERED AN ATTACK VECTOR WITHIN THE APTS?

In the search to be able to place grooming as an attack vector within social engineering and information security, Krombholz et al. [59], in its taxonomy proposal refers to APT. Chen et al. in [42], clearly describes and justifies the difference between APT attacks and traditional cyber-attacks. In this differentiation they determine that the APT come from highly organized, sophisticated, determined and obstinate attackers who direct their attacks to specific people or organizations, government institutions, commercial companies with the purpose of obtaining competitive advantages, strategic benefits that in some situations cause irreparable damage. All this process is successful based on the repetitiveness of their attack attempts, maintaining discretion and non-invasive

TABLE 4. APT criteria applied to grooming.

Criteria	Attack: Grooming	
	Fulfillment	Description
This attack could have been avoided in more than one way	False	Given the process of execution, grooming is a surprise, and it is unlikely (concerning the nature of the victim) that it can be avoided with minimal countermeasures and security controls.
This attack did not require much adaptation by the attackers	False	The attempts of the attackers, to achieve their objective, require a great adaptation or intense evasive techniques in response to the victim's attempts at defense.
This attack did not show any novelty in its variants.	False	Applied social engineering techniques make a grooming attack successful. The novelty in the attack methods used makes it difficult to detect them with existing tools and techniques.

immediately, but with high resistance capabilities, in the long term in order to meet their objectives.

One of the attacks considered APTs, is social engineering. Attackers who apply social engineering have shown very diverse behavioral patterns (friendship, empathy, threat, abuse of trust, etc.). These psychological traits, when organized in phases, demonstrate a common behavioral pattern that is persistence. As it has been demonstrated in the research, grooming follows this same behavioral pattern, for this reason being a type of social engineering enters the APT classification.

Alshamrani *et al.* [44] built a list of criteria that determine whether an attack is APT or a common cyber-attack. If the answer to any of these relative criteria is true for the attack case in question, then the attack is not APT. In Table 4, a grooming attack is contrasted with the mentioned list of criteria, in this way, it is framed within the group of APT attacks, in order to analyze it from that perspective.

There is a divergence of criteria when updating the concept of an APT; for this reason, it is difficult to take security measures against these unconventional attacks. On the other hand, organizations such as the National Institute of Standards and Technology (NIST) have not taken into account the new objectives and damages caused by the APT. However, the increasing manifestation of the APT with sophisticated methods and deterministic characteristics make the security industry point out the need to review the definition of APT, to include other domains with new attack targets [43].

In the scientific field, several criteria have been proposed to update the concept of an APT. It starts from a military criterion, to refer to a class of sophisticated attacks, carried out by highly skilled attackers, whose objective is to obtain sensitive information from their victim [40]. The definition of an APT is made up of the combination of three terms:

- Threat: the threat in APT attacks is usually the loss of sensitive data, the impediment of critical components or the breaking of the victim. These are growing threats for many national entities and organizations that have advanced protection systems that protect their data.
- Persistent: APT attackers are very determined, persistent and obstinate. Once they get access to the victim, they try to extend their stay for as long as possible. They use several evasive techniques to avoid detection and follow a slow and discreet approach to increase their likelihood of success.
- Advanced: APT attackers usually have advanced tools and methods, necessary to perform an APT attack. These advanced methods include the use of multiple attack vectors to execute, as well as to keep the attack going.

According to the NIST [41], an APT attacker: (i) pursues its objectives repeatedly over a prolonged period of time; (ii) it adapts to the efforts of defenders to resist it; and (iii) is determined to maintain the level of interaction necessary to achieve its objectives. These objectives are usually the theft of information or the deterioration of critical aspects of a mission or program through multiple attack vectors.

Within the study of the APT [43]–[45], [47], we have observed the interest of contributing to the detailed study of differentiated attacks according to their processes by applying life cycles proposed by industry and academia (see Figure 4) based on concepts of computer attacks.

C. CAN THE STUDIES RELATED TO GROOMING SUPPORT FUTURE RESEARCH ASSOCIATED WITH SOCIAL ENGINEERING?

Taking into account that the studies of social engineering are in continuous development and have as one of their objectives to determine behavioral patterns of the attackers and their victims, it is evident that the present study can support future investigations aligned to the study of social engineering as an APT within the field of information security.

VIII. CONCLUSION AND FUTURE WORK

We have positioned grooming as an attack vector within social engineering and information security. Through the modeling of topics, different stages or seasons of a life cycle of grooming associated with social engineering is determined; this will allow supporting investigations related to identifying patterns of malicious behavior online. Additionally, a psychological and technical profile of the type of attacker associated with online pedophilia has been presented. We have conducted two experiments, the first consists of determining, in a computational way through a statistical model, stations, or cycles related to grooming. The second experiment gives a linguistic concept to the established stations. In the last experiment, a linear model of machine learning was applied, according to the determined linguistic characteristics, aiming at characterizing text pertinent to the case study (online pedophilia), obtaining an accuracy percentage of 97%. All data was selected and downloaded from the Perverted-Justice website. It is worth noting that although the related work does not align directly with information security, we address the topic following an information security approach. For this reason, the research covers several fields aligned to security, such as APT persistent advanced attacks and social engineering. The processing and evaluation of short text lines obtained from instant messaging protocol, through the proposed approach, does not only apply to the case study but can be reproduced in other security-related fields, these can be online bullying, bank fraud, phishing, among others. One of the main challenges in the path of new cases of study is obtaining relevant data related to the research field, for this reason, it is essential to promulgate and to disseminate in the scientific community this type of studies, to gain more interest in the academy and industry. As future

work, we have planned to implement the model in parental control systems for further optimization. As a step before this implementation, the model must be contrasted with data from instant messaging, with texts of adult conversations of a sexual nature and frequent conversations. In this way, the system will have the ability to unlink these conversations from the classification and location of relevant texts to the violation of privacy.

REFERENCES

[1] L. Penna, A. Clark, and G. Mohay, "Challenges of automating the detection of paedophile activity on the Internet," in *Proc. 1st Int. Workshop Systematic Approaches Digit. Forensic Eng.*, 2005, pp. 206–220.
[2] N. Pendar, "Toward spotting the pedophile telling victim from predator in text chats," in *Proc. Int. Conf. Semantic Comput. (ICSC)*, Sep. 2007, pp. 235–241.
[3] R. C. Hall, "A profile of pedophilia: Definition, characteristics of offenders, recidivism, treatment outcomes, and forensic issues," *Mayo Clinic Proc.*, vol. 82, no. 4, pp. 457–471, 2007.
[4] D. Bogdanova, P. Rosso, and T. Solorio, "On the impact of sentiment and emotion based features in detecting online sexual predators," in *Proc. 3rd Workshop Comput. Approaches Subjectivity Sentiment Anal.*, Jul. 2012, pp. 110–118. [Online]. Available: http://www.aclweb.org/anthology/W12-3717
[5] K. F. Durkin, "Misuse of the Internet by pedophiles: Implications for law enforcement and probation practice," *Fed. Probation*, vol. 61, no. 3, p. 14, Sep. 1997.
[6] H. J. Escalante, E. Villatoro-Tello, S. E. Garza, A. P. López-Monroy, M. Montes-Y-Gómez, and L. Villaseñor-Pineda, "Early detection of deception and aggressiveness using profile-based representations," *Expert Syst. Appl.*, vol. 89, pp. 99–111, Dec. 2017.
[7] I. McGhee, J. Bayzick, A. Kontostathis, L. Edwards, A. McBride, and E. Jakubowski, "Learning to identify Internet sexual predation," *Int. J. Electron. Commerce*, vol. 15, no. 3, pp. 103–122, 2011. doi: 10.2753/JEC1086-4415150305.
[8] R. C. W. Hall and R. C. W. Hall, "A profile of pedophilia: Definition, characteristics of offenders, recidivism, treatment outcomes, and forensic issues," *Focus*, vol. 7, no. 4, pp. 522–537, 2009.
[9] C. Katz, "Internet-related child sexual abuse: What children tell us in their testimonies," *Children Youth Services Rev.*, vol. 35, no. 9, pp. 1536–1542, 2013.
[10] J. C. Young and C. S. Widom, "Long-term effects of child abuse and neglect on emotion processing in adulthood," *Child Abuse Neglect*, vol. 38, no. 8, pp. 1369–1381, 2014.
[11] K. MacFarlane and V. Holmes, "Agent-mediated information exchange: Child safety online," in *Proc. Int. Conf. Manage. Service Sci.*, 2009, pp. 1–5.
[12] H. Pranoto, F. E. Gunawan, and B. Soewito, "Logistic models for classifying online grooming conversation," *Proc. Comput. Sci.*, vol. 59, pp. 357–365, 2015. doi: 10.1016/j.procs.2015.07.536.
[13] A. Cano, M. Fernandez, and H. Alani, "Detecting child grooming behaviour patterns on social media," in *Lecture Notes in Computer Science* (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 8851. Cham, Switzerland: Springer, 2014, pp. 412–427.
[14] R. Heartfield, G. Loukas, and D. Gan, "You are probably not the weakest link: Towards practical prediction of susceptibility to semantic social engineering attacks," *IEEE Access.*, vol. 4, pp. 6910–6928, 2016.
[15] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Jan. 2003.
[16] S. Huber, H. Wiemer, D. Schneider, and S. Ihlenfeldt, "DMME: Data mining methodology for engineering applications—A holistic extension to the CRISP-DM model," *Proc. CIRP*, vol. 79, pp. 403–408, Jul. 2019.
[17] S. Yong, D. Lindskog, R. Ruhl, and P. Zavarsky, "Risk mitigation strategies for mobile Wi-Fi robot toys from online pedophiles," in *Proc. IEEE 3rd Int. Conf. Privacy, Secur., Risk Trust IEEE 3rd Int. Conf. Social Comput.*, Oct. 2011, pp. 1220–1223.
[18] P. Elzinga, K. E. Wolff, and J. Poelmans, "Analyzing chat conversations of pedophiles with temporal relational semantic systems," in *Proc. Eur. Intell. Secur. Inform. Conf. (EISIC)*, 2012, pp. 242–249.
[19] A. W. K. Kong, "Tutorial-1: New criminal and victim identification methods for sexual offenses against women and children," in *Proc. IEEE Int. WIE Conf. Elect. Comput. Eng. (WIECON-ECE)*, Dec. 2015, pp. 1–4.
[20] M. Rutgaizer, Y. Shavitt, O. Vertman, and N. Zilberman, "Detecting pedophile activity in BitTorrent networks," in *Lecture Notes in Computer Science* (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 7192. Berlin, Germany: Springer, 2012, pp. 106–115.
[21] L. N. Olson, J. L. Daggs, B. L. Ellevold, and T. K. K. Rogers, "Entrapping the innocent: Toward a theory of child sexual predators' luring communication," *Commun. Theory*, vol. 17, no. 3, pp. 231–251, 2007.
[22] A. Hofmann, U. Barth, I. Haas, and F. Holzwarth, *Detection of Child Sexual Abuse Media: Classification of the Associated Filenames*. Berlin, Germany: Springer, 2013, pp. 1–5.
[23] D. Bogdanova, P. Rosso, and T. Solorio, "Modelling fixated discourse in chats with cyberpedophiles," *Proc. Workshop Comput. Approaches Deception Detection*, 2012, pp. 86–90. [Online]. Available: http://www.aclweb.org/anthology/W12-0413
[24] A. Vartapetiance and L. Gillam, "'Our little secret': Pinpointing potential predators," *Secur. Informat.*, vol. 3, no. 1, pp. 1–19, 2014.
[25] D. Bogdanova, P. Rosso, and T. Solorio, "Exploring high-level features for detecting cyberpedophilia," *Comput. Speech Lang.*, vol. 28, no. 1, pp. 108–120, 2014. doi: 10.1016/j.csl.2013.04.007.
[26] F. E. Gunawan, L. Ashianti, S. Candra, and B. Soewito, "Detecting online child grooming conversation," in *Proc. 11th Int. Conf. Knowl., Inf. Creativity Support Syst. (KICSS)*, 2017, pp. 1–6.
[27] B. Leclerc, R. Wortley, and S. Smallbone, "Getting into the script of adult child sex offenders and mapping out situational prevention measures," *J. Res. Crime Delinquency*, vol. 48, no. 2, pp. 209–237, 2011.
[28] C. D. Marcum, "Interpreting the intentions of Internet predators: An examination of online predatory behavior," *J. Child Sexual Abuse*, vol. 16, no. 4, pp. 99–114, 2007.
[29] K. J. Mitchell, D. Finkelhor, L. M. Jones, and J. Wolak, "Growth and change in undercover online child exploitation investigations, 2000–2006," *Policing Soc.*, vol. 20, no. 4, pp. 416–431, 2010.
[30] K. J. Mitchell, D. Finkelhor, L. M. Jones, and J. Wolak, "Use of social networking sites in online sex crimes against minors: An examination of national incidence and means of utilization," *J. Adolescent Health*, vol. 47, no. 2, pp. 183–190, 2010.
[31] L. Crystal Jiang and J. T. Hancock, "Absence makes the communication grow fonder: Geographic separation, interpersonal media, and intimacy in dating relationships," *J. Commun.*, vol. 63, no. 3, pp. 556–577, 2013.
[32] H. Whittle, C. Hamilton-Giachritsis, A. Beech, and G. Collings, "A review of online grooming: Characteristics and concerns," *Aggression Violent Behav.*, vol. 18, no. 1, pp. 62–70, 2013.
[33] K. J. Mitchell, L. M. Jones, D. Finkelhor, and J. Wolak, "Understanding the decline in unwanted online sexual solicitations for US youth 2000–2010: Findings from three youth Internet safety surveys," *Child Abuse Neglect*, vol. 37, no. 12, pp. 1225–1236, 2013.
[34] L. Penna, A. Clark, and G. Mohay, "A framework for improved adolescent and child safety in MMOs," in *Proc. Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, 2010, pp. 33–40.
[35] M. W. R. Miah, J. Yearwood, and S. Kulkarni, "Detection of child exploiting chats from a mixed chat dataset as a text classification task," in *Proc. Australas. Lang. Technol. Assoc. Workshop*, 2011, pp. 157–165.
[36] W. Wang, P. M. Barnaghi, and A. Bargiela, "Probabilistic topic models for learning terminological ontologies," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 7, pp. 1028–1040, Jul. 2010.
[37] A. Daud, J. Li, L. Zhou, and F. Muhammad, "Knowledge discovery through directed probabilistic topic models: A survey," *Frontiers Comput. Sci. China*, vol. 4, no. 2, pp. 280–301, 2010.
[38] Y. Lu, Q. Mei, and C. Zhai, "Investigating task performance of probabilistic topic models: An empirical study of PLSA and LDA," *Inf. Retr.*, vol. 14, no. 2, pp. 178–203, 2011.
[39] R. Alghamdi and K. Alfalqi, "A survey of topic modeling in text mining," *Int. J. Adv. Comput. Sci. Appl.*, vol. 6, no. 1, pp. 147–153, 2015.
[40] S. Radack, "Managing information security risk: Organization, mission and information system view," Nat. Inst. Standards Technol., Gaithersburg, MD, USA, Tech. Rep. 800-39, 2011.
[41] R. Kissel, *Glossary of Key Information Security Terms*. Gaithersburg, MD, USA: Diane Publishing, 2011.
[42] P. Chen, L. Desmet, and C. Huygens, "A study on advanced persistent threats," in *Proc. IFIP Int. Conf. Commun. Multimedia Secur.* Berlin, Germany: Springer, 2014, pp. 63–72.

[43] B. I. Messaoud, K. Guennoun, M. Wahbi, and M. Sadik, "Advanced persistent threat: New analysis driven by life cycle phases and their challenges," in *Proc. Int. Conf. Adv. Commun. Syst. Inf. Secur. (ACOSIS)*, Oct. 2016, pp. 1–6.

[44] A. Alshamrani, S. Myneni, A. Chowdhary, and D. Huang, "A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1851–1877, 2nd Quart., 2019.

[45] L. Shenwen, L. Yingbo, and D. Xiongjie, "Study and research of APT detection technology based on big data processing architecture," in *Proc. IEEE 5th Int. Conf. Electron. Inf. Emergency Commun.*, May 2015, pp. 313–316.

[46] M. Ussath, D. Jaeger, F. Cheng, and C. Meinel, "Advanced persistent threats: Behind the scenes," in *Proc. Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2016, pp. 181–186.

[47] E. Hutchins, M. Cloppert, and R. Amin, "Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains," *Leading Issues Inf. Warfare Secur. Res.*, vol. 1, no. 1, p. 80, 2011.

[48] K. D. Mitnick, W. L. Simon, and S. Wozniak, *The Art of Deception: Controlling the Human Element of Security*. Indianapolis, IN, USA: Wiley, 2006.

[49] F. Mouton, M. M. Malan, L. Leenen, and H. S. Venter, "Social engineering attack framework," in *Proc. Inf. Secur. South Afr. (ISSA)*, 2014, pp. 1–9.

[50] G. Mariscal, O. Marban, and C. Fernández, "A survey of data mining and knowledge discovery process models and methodologies," *Knowl. Eng. Rev.*, vol. 25, no. 2, pp. 137–166, 2010.

[51] T. K. Landauer and S. T. Dumais, "A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge," *Psychol. Rev.*, vol. 104, no. 2, p. 211, 1997.

[52] P. J. Black, M. Wollis, M. Woodworth, and J. T. Hancock, "A linguistic analysis of grooming strategies of online child sex offenders: Implications for our understanding of predatory sexual behavior in an increasingly computer-mediated world," *Child Abuse Neglect*, vol. 44, pp. 140–149, Jun. 2015.

[53] A. Sharma, R. Sharma, V. K. Sharma, and V. Shrivatava, "Application of data mining—A survey paper," *Int. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 2, pp. 2023–2025, 2014.

[54] P. P. Sondwale, "Overview of predictive and descriptive data mining techniques," *Int. J. Adv. Res.in Comput. Sci. Softw. Eng.*, vol. 5, no. 4, pp. 262–265, 2015.

[55] N. Jain and V. Srivastava, "Data mining techniques: A survey paper," *Int. J. Res. Eng. Technol.*, vol. 2, no. 11, pp. 1163–2319, 2013.

[56] P. Ristoski and H. Paulheim, "Semantic Web in data mining and knowledge discovery: A comprehensive survey," *J. Web Semantics*, vol. 36, pp. 1–22, Jan. 2016.

[57] B. K. Baradwaj and S. Pal, "Mining educational data to analyze students' performance," Jan. 2012, *arXiv:1201.3417*. [Online]. Available: https://arxiv.org/abs/1201.3417

[58] A. E. Cano, M. Fernandez, and H. Alani, "Detecting child grooming behaviour patterns on social media," in *Proc. Int. Conf. social Inform.* Cham, Switzerland: Springer, 2014, pp. 412–427.

[59] K. Krombholz, H. Hobel, M. Huber, and E. Weippl, "Advanced social engineering attacks," *J. Inf. Secur. Appl.*, vol. 22, pp. 113–122, 2015. doi: 10.1016/j.jisa.2014.09.005.

PATRICIO ZAMBRANO received the degree in electronic engineering in telecommunications at Universidad de las Fuerzas Armadas (ESPE), Sangolquí, Ecuador, in 2006, and the master's degree in information and connectivity networks, in 2012. He currently participates in the Ph.D. program of Systems of Escuela Politécnica Nacional, Quito, Ecuador, as a Research Student. In the scientific field, he has made significant contributions to the privacy of information with the help of data analytics and machine learning. He is currently a Full Professor with the Department of Computing and Computer Science, Escuela Politécnica Nacional. His research interests include the security and privacy of the networks, research, and the application of artificial intelligence to new topics of unconventional research.

JENNY TORRES received the M.Sc. degree in computer science security from University Paris-Est Créteil, the Computer Systems Engineering degree from the Escuela Politécninca Nacional (EPN), in 2006, the master's degree in management of networks and telecommunications from the Universidad de las Fuerzas Armadas (ESPE), in 2008, and the Ph.D. degree in computer science from Sorbonne University Campus Pierre and Marie Curie in France, in 2013.

She was an Invited Researcher with the University of Paraná, Curitiba, Brazil. She is currently a Professor and a Researcher with the Faculty of Engineering Systems, Escuela Politécninca Nacional (EPN). Her research interests include computer security, network management, identity management, wireless networks, and open infrastructures. She was a recipient of the SENESCYT Scholarship.

LUIS TELLO-OQUENDO received the Electronic and Computer Engineering degree (Hons.) from the Escuela Superior Politécnica de Chimborazo (ESPOCH), Ecuador, in 2010, and the M.Sc. degree in telecommunication technologies, systems, and networks, and the Ph.D. degree *(cum laude)* in telecommunications from the Universitat Politècnica de València (UPV), Spain, in 2013 and 2018, respectively. In 2011, he was a Lecturer with the Facultad de Ingeniería Electrónica, ESPOCH. From 2013 to 2018, he was a Graduate Research Assistant with the Broadband Internetworking Research Group, UPV. From 2016 to 2017, he was a Research Scholar with the Broadband Wireless Networking Laboratory, Georgia Institute of Technology, Atlanta, GA, USA. He is currently an Associate Professor with the College of Engineering, Universidad Nacional de Chimborazo, Ecuador. His research interests include machine-type communications, wireless software-defined networks, 5G and beyond cellular systems, the Internet of Things, and machine learning. He is a member of ACM. He received the Best Academic Record Award from the Escuela Técnica Superior de Ingenieros de Telecomunicación, UPV, in 2013, and the IEEE ComSoc Award for attending the IEEE ComSoc Summer School at The University of New Mexico, Albuquerque, NM, USA, in 2017.

RUBÉN JÁCOME is currently a Graduate Engineer from the Faculty of Computer Science, Escuela Politécninca Nacional (EPN). He is working in USA in areas related to TIC and information assurance as a Computer Consultant. He has specialized in privacy topics, secure programming, and data analytics.

MARCO E. BENALCÁZAR received the M.S. degree in solar photovoltaic energy systems from the Universidad Internacional de Andalucía, Spain, in 2012, and the Ph.D. degree in electronic engineering from the Universidad Nacional de Mar del Plata, Argentina, in 2014.

From September 2012 to February 2013, he did research on clustering at the Genomic Signal Processing Laboratory, Texas A&M University, College Station, Texas, USA. He was an Engineer in electronic and telecommunications with the Escuela Politécnica Nacional (EPN), Quito, Ecuador, in 2009, where he is currently an Associate Professor and the Director of the Artificial Intelligence and Computer Vision Research Laboratory, Department of Computer Science and Informatics. He has authored or coauthored several articles in clustering, image processing, hand gesture recognition, and applications of artificial intelligence. His current research interests include pattern recognition, machine learning, such as theory and applications, artificial intelligence and computer vision, image processing, and applications of artificial intelligence.

II. Código del script *regex_formatChatLogs.pl*

```
#scripts para estandarizar formato de chats extraidos desde perverted justice
#script 1
use Encode;
use utf8;
use HTML::Entities;#Convert html encoding into utf8 plain text and vice versa
use warnings;
use strict;
binmode STDOUT, ":utf8";

if (! defined $ARGV[0]){die "Formato de uso: \nperl regex_format.pl
input_directory\nEjemplo de uso:\nperl regex_remplazo.pl html_input\n";}
main($ARGV[0]);#Calling main procedure

sub main{
  my ($directoryName) =@_;
  #Opening directory and storing file names into an array
  my @fileList=&readFiles($directoryName);
  foreach my $fileName (@fileList){
    my $htmlFileContent = openFile($fileName);
        $htmlFileContent = decode_entities($htmlFileContent);

        #FASE I:       ESTANDARIZAR FORMATO DE CHATS
                #0.1 Limpiar etiquetas existentes dentro de mensajes
                $htmlFileContent =~
s/(M[\)\]]\:)[\s]?(?:\<(?:(?!\>).)*\>){0,3}((?:(?!<).)*)(?:\<\/(?:(?!\>).)*\>){0,3}/$1
$2/gm;

                #0.2 Limpiar comentarios que inician con la etiqueta: <span class='code_c'>
comentario </span>
                $htmlFileContent =~ s/<span
class='code_c'>(?:(?:(?!\<\/span\>).)*\<\/span\>)//gm;

                #0.3 Limpiar etiquetas <b> & </b> dentro de mensajes
                $htmlFileContent =~
s/<b>[\s]?((?:(?!\<\/b>).)*[\s]?)\<\/b>[\s]?(\<br[\s]?\/\>)/$1 $2/gm;

        #Estandarizar nombre atacante
                #1.1 Cuando nombre de atacante tiene etiqueta: <span class='blueBold'>
                $htmlFileContent =~ s/^<span class='blueBold'>[\s]?((?:(?!\<\/span\>).)*)[
]?(?:\<\/span\>)[\s]?(?:[\[\(][^)\]]+[\]\)]\:[\s]?)(?:(?:\<(?!\>).)*\>)?((?:(?!\<).)*)(?:(?
:\<\/(?!\>).)*\>)?(?:\s)?(?:\<br[\s]?\/\>$)/\{$1\}:$2/gm;

                #1.2 Cuando nombre de atacante tiene etiqueta: <b>
                $htmlFileContent =~
s/^<b>[\s]?((?:(?!\<\/b\>).)*)[\s]?(?:\<\/b\>)[\s]?(?:[\[\(][^)\]]+[\]\)]\:)[\s]?(?:(?:\<(?
!\>).)*\>)?[\s]?((?:(?!\<).)*)(?:(?:\<\/(?!\>).)*\>)?[\s]?(?:\<br[\s]?\/\>$)/\{$1\}:$2/gm;

                #1.3 Cuando nombre de atacante tiene etiqueta: <b> & <span class='blueBold'>
                $htmlFileContent =~ s/^(?:<b>)?<span
class='blueBold'>[\s]?((?:(?!\<\/span\>).)*)[
]?(?:\<\/span\>)[\s]?(?:[\[\(][^)\]]+[\]\)]\:[\s]?)(?:(?:\<(?!\>).)*\>)?((?:(?!\<).)*)(?:(?
:\<\/(?!\>).)*\>)?(?:\s)?(?:\<br[\s]?\/\>$)/\{$1\}:$2/gm;

                #0.4 Limpiar etiquetas <b> & </b> dentro de mensajes
                $htmlFileContent =~ s/<b>[\s]?((?:(?!\<\/b>).)*[\s]?)\<\/b>/$1/gm;

                #1.3 Cuando nombre de atacante tiene etiqueta: <b> & <span class='blueBold'>
                $htmlFileContent =~ s/^(?:<b>)?<span
class='blueBold'>[\s]?((?:(?!\<\/span\>).)*)[
]?(?:\<\/span\>)[\s]?(?:[\[\(][^)\]]+[\]\)]\:[\s]?)(?:(?:\<(?!\>).)*\>)?((?:(?!\<).)*)(?:(?
:\<\/(?!\>).)*\>)?(?:\s)?(?:\<br[\s]?\/\>$)/\{$1\}:$2/gm;
```

```
                #1.4 Cuando hay emoticones dentro del mensajes
                $htmlFileContent =~ s/^(?:<b>)?<span
class='blueBold'>[\s]?((?:(?!\<\/span\>).)*)[
]?(?:\<\/span\>)[\s]?(?:[\[\(][^)\]]+[\]\)]\:[\s]?)((?:(?!\<br \/\>).)*)\<br
\/\>/\{$1\}:$2/gm;

        #Estandarizar nombre victima
        $htmlFileContent =~
s/^([^\[\n\(]+)(?:\s)?(?:[\[\(][^)\]]+[\]\)]\:)(?:\s)?(?:\<(?:(?!\>).)*\>)?((?:(?!\<).)*)(?
:(?:\<\/(?!\>).)*\>)?(?:\s)?(?:\<br\s\/\>$)/\[$1\]\=$2/gm;

        #Limpiar etiquetas restantes
        $htmlFileContent =~ s/(?:\<(?:(?!\/\>).)*\/\>)//gm;
        $htmlFileContent =~ s/^[^\[\{]+//gm;

    #Storing the output
    &storeNewFile($fileName, $htmlFileContent);

  }#\for
}

sub trim{
        my ($str) = @_;

        $str =~ s/^ *| *$//g;
        return $str;
}

sub printOrNot{
  my ($openTag, $str ,$closeTag) = @_;

  if ($str !~ /^$/){
    $str = $openTag.$str.$closeTag;
  }

  return $str;
}

#Returns the string if defined, empty otherwise.
sub imprime{
        my ($str) = @_;

        if (!defined $str){
          $str="";
        }
        return $str;
}

#Receives text between pairs of font tags #Produces one string without html tags.
sub removeFontTags{
   my ($str) = @_;

   $str =~ s/\s*<font[^>]*>|<\/font>\s*/ /g;
   $str =~ s/ {2,}/ /g;
   return trim($str);
}

sub storeNewFile{
  my ($fileName, $fileContent) = @_;
  my $directoryName = "limpio";

  if ( -d "$directoryName") {
    print "Directory found \"$directoryName/$fileName\"\n";
```

```
  }
  else {
    print "Directory \"$directoryName\" was not found, but it has just been created.\n";
    mkdir("$directoryName");
  }

  #Removing directory name from the inicial input.
  $fileName =  substr($fileName, index ($fileName, "/")+1, (length ($fileName)-index
($fileName, "/")+1));
  &writeFile("$directoryName\\$fileName", $fileContent);
}

#Creates a new file. #Deletes all content of old file, if existed.
sub writeFile
{
  my($fileName, $content) = @_;#
  #writing content into a file.
  #open FILE, ">$fileName";
  open(FILE, ">:utf8",$fileName) or die "Can't read file \"$fileName\" [$!]\n";
  print FILE $content;
  close (FILE);
}

sub readFiles{
        my ($folder)=@_;
        my @files=<$folder/*>;
        return @files;
}

sub openFile{
        my ($fileName) = @_;
        local $/;#read full file instead of only one line.
        open(FILE, "<:utf8",$fileName) or die "Can't read file \"$fileName\" [$!]\n";
        my $fileContent = <FILE>;
        close (FILE);

        return $fileContent;
}
```

III. Código del script *regex_getAttacker.pl*

```
#!/usr/bin/perl
#Scripts para estandarizar formato de chats extraidos desde perverted justice
#Script 2: Extraer mensajes de atacante
use Encode;
use utf8;
use HTML::Entities;#Convert html encoding into utf8 plain text and vice versa
use warnings;
use strict;
binmode STDOUT, ":utf8";
if (! defined $ARGV[0]){die "Formato de uso: \nperl regex_getAttacker.pl
input_directory\nEjemplo de uso:\nperl regex_remplazo.pl html_input\n";}
main($ARGV[0]);#Calling main procedure

sub main{
  my ($directoryName) =@_;

  #Opening directory and storing file names into an array
  my @fileList=&readFiles($directoryName);
  foreach my $fileName (@fileList){
    my $htmlFileContent = openFile($fileName);
        $htmlFileContent = decode_entities($htmlFileContent);

        #FASE II:      EXTRACCION DE MENSAJES ATACANTE
        $htmlFileContent =~ s/^[\{](?:(?:(?!\}).)*\}\:)(.*)$/$1\n/gm;
        $htmlFileContent =~ s/^[\[](?:(?:(?!\]).)*\]\=)(?:.*)$//gm;
        $htmlFileContent =~ s/^[\s]+//gm;
    #$htmlFileContent =~ s//;

    #Storing the output
    &storeNewFile($fileName, $htmlFileContent);

  }#\for
}

sub trim{
        my ($str) = @_;

        $str =~ s/^ *| *$//g;
        return $str;
}

sub printOrNot{
  my ($openTag, $str ,$closeTag) = @_;

  if ($str !~ /^$/){
    $str = $openTag.$str.$closeTag;
  }

  return $str;
}

#Returns the string if defined, empty otherwise.
sub imprime{
        my ($str) = @_;

        if (!defined $str){
          $str="";
        }
        return $str;
}
```

```
#Receives text between pairs of font tags #Produces one string without html tags.
sub removeFontTags{
   my ($str) = @_;

   $str =~ s/\s*<font[^>]*>|<\/font>\s*/ /g;
   $str =~ s/ {2,}/ /g;
   return trim($str);
}

sub storeNewFile{
  my ($fileName, $fileContent) = @_;
  my $directoryName = "attacker";

  if ( -d "$directoryName") {
    print "Directory found \"$directoryName/$fileName\"\n";
  }
  else {
    print "Directory \"$directoryName\" was not found, but it has just been created.\n";
    mkdir("$directoryName");
  }

  #Removing directory name from the inicial input.
  $fileName =  substr($fileName, index ($fileName, "/")+1, (length ($fileName)-index
($fileName, "/")+1));
  &writeFile("$directoryName\\$fileName", $fileContent);
}

#Creates a new file. #Deletes all content of old file, if existed.
sub writeFile
{
  my($fileName, $content) = @_;#
  #writing content into a file. #open FILE, ">$fileName";
  open(FILE, ">:utf8",$fileName) or die "Can't read file \"$fileName\" [$!]\n";
  print FILE $content;
  close (FILE);
}

sub readFiles{
        my ($folder)=@_;
        my @files=<$folder/*>;
        return @files;
}

sub openFile{
        my ($fileName) = @_;
        local $/;#read full file instead of only one line.
        open(FILE, "<:utf8",$fileName) or die "Can't read file \"$fileName\" [$!]\n";
        my $fileContent = <FILE>;
        close (FILE);
        return $fileContent;
}
```

IV. Código de la función *preprocessText.m*

```
function documents = preprocessTextLite(textData)
% Tokenize the text.
documents = tokenizedDocument(textData);
% Convert to lowercase.
documents = lower(documents);
% Erase punctuation.
documents = erasePunctuation(documents);
end
```

V. Modelo Estadístico para clasificación de texto en Matlab

```
rng('default')
%El archivo chatLogs.csv contiene los datos de texto a ser analizados.
filename = "chatLogs.csv";
data = readtable(filename,'TextType','string');
%Eliminar las filas que con mensajes de texto vacíos.
idx = strlength(data.message_text) == 0;
data(idx,:) = [];

%Convertir las etiquetas en la columna message_stage de la tabla a categóricas y visualizar
la distribución de las clases en los datos utilizando un histograma.
ata.message_stage = categorical(data.message_stage);
figure
h = histogram(data.message_stage);
xlabel("Class")
ylabel("Frequency")
title("Class Distribution")

%Particionar los datos en una partición de entrenamiento y un conjunto retenido de prueba.
Especificar que el porcentaje retenido sea del 30%.
cvp = cvpartition(data.message_stage,'Holdout',0.3);
dataTrain = data(cvp.training,:);
dataTest = data(cvp.test,:);

%Extraer los datos de texto y las etiquetas desde las tablas.
textDataTrain = dataTrain.message_text;
textDataTest = dataTest.message_text;
YTrain = dataTrain.message_stage;
YTest = dataTest.message_stage;

%Utilizar la función preprocessText definida previamente para preparar los datos
de texto.
documents = preprocessText(textDataTrain);
documents(1:5)

%Crear un modelo bag-of-words a partir de los documentos tokenizados.
bag = bagOfWords(documents)

%Eliminar las palabras del modelo bag-of-words que no aparezcan más de dos veces en total,
así como cualquier documento que no contenga palabras, y eliminar las correspondientes
etiquetas.
bag = removeInfrequentWords(bag,2);
[bag,idx] = removeEmptyDocuments(bag);
YTrain(idx) = [];
bag
```

```
%Entrenar el modelo de clasificación lineal multiclase utilizando la función fitcecoc.
Especificar la propiedad Counts del modelo de bolsa de palabras como predictores y las
etiquetas de tipo de stage como respuesta.
XTrain = bag.Counts;
mdl = fitcecoc(XTrain,YTrain,'Learners','linear')

%Predecir las etiquetas de los datos de prueba utilizando el modelo entrenado y calcular la
precisión de la clasificación.
documentsTest = preprocessText(textDataTest);
XTest = encode(bag,documentsTest);

YPred = predict(mdl,XTest);
acc = sum(YPred == YTest)/numel(YTest)

%Predecir utilizando nuevos datos
str = [ ...
"is your dad usually around?"
"do u wanna come in like an hour?"
"wish you had some more body pics"];
documentsNew = preprocessText(str);
XNew = encode(bag,documentsNew);
labelsNew = predict(mdl,XNew)
```

VI. Red Neuronal Recurrente LSTM para clasificación de texto en Matlab

```
%Importar los datos de registros de chats.
filename = "chatLogs.csv";
data = readtable(filename,'TextType','string');
head(data)

%Eliminar las filas de la tabla con mensajes de texto vacíos.
idxEmpty = strlength(data.message_text) == 0;
data(idxEmpty,:) = [];

%Convierte etiquetas a tipo categóricas.
data.message_stage = categorical(data.message_stage);

%Visualizar la distribución de las clases en los datos utilizando un histograma.
f = figure;
f.Position(3) = 1.5*f.Position(3);
h = histogram(data.message_stage);
xlabel("Class")
ylabel("Frequency")
title("Class Distribution")

%Se dividen los datos en una partición de entrenamiento y una partición retenida
para validación y prueba.
cvp = cvpartition(data.message_stage,'Holdout',0.3);
dataTrain = data(training(cvp),:);
dataHeldOut = data(test(cvp),:);

%Particionar nuevamente el conjunto retenido para obtener un conjunto de validación.
cvp = cvpartition(dataHeldOut.message_stage,'HoldOut',0.5);
dataValidation = dataHeldOut(training(cvp),:);
dataTest = dataHeldOut(test(cvp),:);

%Extraer los datos de texto y etiquetas desde las tablas particionadas.
textDataTrain = dataTrain.message_text;
textDataValidation = dataValidation.message_text;
textDataTest = dataTest.message_text;
YTrain = dataTrain.message_stage;
YValidation = dataValidation.message_stage;
YTest = dataTest.message_stage;

%Pre procesar los datos de entrenamiento y los datos de validación utilizando la
función preprocessTextLite.
documentsTrain = preprocessTextLite(textDataTrain);
documentsValidation = preprocessTextLite(textDataValidation)

%Para crear un codificador de palabras, se utiliza la función wordEncoding.
enc = wordEncoding(documentsTrain);

%Rellenar y truncar los documentos para que todos tengan la misma longitud.
documentLengths = doclength(documentsTrain);
figure
histogram(documentLengths)
title("Document Lengths")
xlabel("Length")
ylabel("Number of Documents")

%Convertir los documentos en secuencias de índices numéricos usando doc2sequence
XTrain = doc2sequence(enc,documentsTrain,'Length',15);
XTrain(1:5)

%Convertir los documentos de validación a secuencias.
XValidation = doc2sequence(enc,documentsValidation,'Length',15);
```

```
%Definir la arquitectura de red LSTM.
inputSize = 1;
embeddingDimension = 100;
numWords = enc.NumWords;
numHiddenUnits = 180;
numClasses = numel(categories(YTrain));
layers = [ ...
sequenceInputLayer(inputSize)
wordEmbeddingLayer(embeddingDimension,numWords)
lstmLayer(numHiddenUnits,'OutputMode','last')
fullyConnectedLayer(numClasses)
softmaxLayer
classificationLayer]

%Configurar las opciones de entrenamiento.
options = trainingOptions('adam', ...
'MaxEpochs',10, ...
'GradientThreshold',1, ...
'InitialLearnRate',0.01, ...
'ValidationData',{XValidation,YValidation}, ...
'Plots','training-progress', ...
'Verbose',false);

%Entrenar la red lstm
net = trainNetwork(XTrain,YTrain,layers,options);

%Probar la red entrenada
textDataTest = lower(textDataTest);
documentsTest = tokenizedDocument(textDataTest);
documentsTest = erasePunctuation(documentsTest);

% Convertir los documentos de prueba en secuencias
XTest = doc2sequence(enc,documentsTest,'Length',15);
XTest(1:5)

%Clasificar los documentos de prueba utilizando la red LSTM entrenada.
YPred = classify(net,XTest);

%Calcular la precisión de la clasificación
accuracy = sum(YPred == YTest)/numel(YPred)

%Predecir utilizando nuevos datos.
logsNew = [ ...
"well I think you're very pretty honey"
"I mean I didn't think your parents would approve of me talking to ya anyway"
"so you’re like getting ready to start high school right?"];
documentsNew = preprocessTextLite(logsNew);
XNew = doc2sequence(enc,documentsNew,'Length',15);
[labelsNew,score] = classify(net,XNew);
[logsNew string(labelsNew)]
```

VII. Red Neuronal Convolucional CNN para clasificación de texto en Matlab

```
% Crear un datastore tabular a partir de los datos  en chatLogsTrain.csv.
filenameTrain = "chatLogsTrain.csv";
textName = "message_text";
labelName = "message_stage";
ttdsTrain = tabularTextDatastore(filenameTrain,'SelectedVariableNames',[textName
labelName]);

%Leer las etiquetas de los datos de entrenamiento.
labels = readLabels(ttdsTrain,labelName);
classNames = unique(labels);
numObservations = numel(labels);

%Transformar el datastore utilizando la función transformTextData.
sequenceLength = 100;
tdsTrain = transform(ttdsTrain, @(data)
transformTextData(data,sequenceLength,emb,classNames))

%Crear un datastore transformado que contenga los datos de validación.
ttdsValidation = tabularTextDatastore(filenameValidation,'SelectedVariableNames',[textName
labelName]);
tdsValidation = transform(ttdsValidation, @(data)
transformTextData(data,sequenceLength,emb,classNames))

%Definir la arquitectura de red
numFeatures = emb.Dimension;
inputSize = [1 sequenceLength numFeatures];
numFilters = 200;
ngramLengths = [2 3 4 5];
numBlocks = numel(ngramLengths);
numClasses = numel(classNames);

layer = imageInputLayer(inputSize,'Normalization','none','Name','input');
lgraph = layerGraph(layer);

for j = 1:numBlocks
N = ngramLengths(j);
block = [
convolution2dLayer([1 N],numFilters,'Name',"conv"+N,'Padding','same')
batchNormalizationLayer('Name',"bn"+N)
reluLayer('Name',"relu"+N)
dropoutLayer(0.2,'Name',"drop"+N)
maxPooling2dLayer([1 sequenceLength],'Name',"max"+N)];
lgraph = addLayers(lgraph,block);
lgraph = connectLayers(lgraph,'input',"conv"+N);
end

figure
plot(lgraph)
title("Network Architecture")

%Agregar la capa de concatenación de profundidad, la capa totalmente conectada, la capa
softmax y la capa de clasificación.
layers = [
depthConcatenationLayer(numBlocks,'Name','depth')
fullyConnectedLayer(numClasses,'Name','fc')
softmaxLayer('Name','soft')
classificationLayer('Name','classification')];

%Conectar las capas de agrupación máxima a la capa de concatenación de profundidad.
for j = 1:numBlocks
N = ngramLengths(j);
```

```
lgraph = connectLayers(lgraph,"max"+N,"depth/in"+j);
end
figure
plot(lgraph)
title("Network Architecture")

%Entrenar la red
miniBatchSize = 128;
numIterationsPerEpoch = floor(numObservations/miniBatchSize);
options = trainingOptions('adam', ...
'MaxEpochs',10, ...
'MiniBatchSize',miniBatchSize, ...
'ValidationData',tdsValidation, ...
'ValidationFrequency',numIterationsPerEpoch, ...
'Plots','training-progress', ...
'Verbose',false);

%Probar la red
filenameTest = "chatLogsTest.csv";
ttdsTest = tabularTextDatastore(filenameTest,'SelectedVariableNames',[textName labelName]);
tdsTest = transform(ttdsTest, @(data)
transformTextData(data,sequenceLength,emb,classNames))
%Realizar predicciones en los datos de prueba utilizando la red entrenada
labelsTest = readLabels(ttdsTest,labelName);
YTest = categorical(labelsTest,classNames);
%Calcular la precision de clasificacion en los datos de prueba.
YPred = classify(net,tdsTest);
```

ESCUELA POLITECNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS

CARRERA:

INGENIERÍA EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

ORDEN DE EMPASTADO

De acuerdo con lo estipulado en el Art. 27 del Instructivo para la Implementación de la Unidad de Titulación en las carreras y programas vigentes de la Escuela Politécnica Nacional, aprobado por el Consejo de Docencia el 22 de abril de 2015, y una vez verificado el cumplimiento del formato de presentación establecido, se autoriza la impresión y encuadernación final del Trabajo de Titulación presentado por:

Nombre: Rubén Andrés Jácome Jiménez

Fecha de autorización: 8/01/2020

ESCUELA POLITECNICA NACIONAL
FACULTAD DE INGENIERIA DE SISTEMAS
DECANATO FIS Quito

MSc. Carlos Montenegro
DECANO DE LA
FACULTAD DE INGENIERIA DE SISTEMAS