# Diseño e implementación de un sistema de identificación de ataques del tipo Smishing

María Paula Moreno
Estudiante maestría en seguridad de la información
Universidad de los Andes
Bogotá, Colombia
maria.moreno@uniandes.edu.co

Nicolás Rodríguez Menjura
Estudiante maestría en seguridad de la información
Universidad de los Andes
Bogotá, Colombia
ne.rodriguez10@uniandes.edu.co

***Resumen*— Este articulo explora cómo crear un sistema que analiza mensajes de texto con el fin de identificar SMS malicioso y las URLs contenidas, esto se realiza mediante una solución desplegada en nube y un modelo de ML que clasificará el mensaje. Los resultados revelan que los mensajes maliciosos contienen ciertos patrones como mayor inclusión de números o solicitan acciones inmediatas con el fin de presionar al usuario final. De este modo se sugiere siempre utilizar este sistema para analizar la URL con el API VT y tener más información antes de responder o acceder al mensaje. Este estudio contribuye a los controles de seguridad y sobre todo a la concientización sobre el uso de SMS y la prevención contra ataques tipo smishing.**

***Palabras clave*— *API, Machine Learning, Phishing, Redes móviles, Short Message Service (SMS), Smishing***

## I. Introducción

En el desarrollo de la propuesta para el proyecto de integración de la maestría en seguridad de la información de la universidad de los Andes, se ha identificado el problema relacionado con el aumento de ataques cibernéticos que hacen uso de servicios de mensajería en redes móviles para suplantar organizaciones y, de este modo, ganar la confianza de los usuarios para robar su información personal o estafarlos. A continuación, se presentarán los diversos aspectos que llevaron a la caracterización de la problemática para dar lugar a la propuesta de solución.

### *A. Actualidad del problema*

El término "smishing" es una combinación de las palabras "SMS" (Short Message Service) y "phishing" (engaño). El smishing es un tipo de ataque que utiliza mensajes de texto (SMS) fraudulentos, porque, haciéndose pasar por organizaciones conocidas, como bancos, empresas de telecomunicaciones, agencias gubernamentales o plataformas en línea, llegan a engañar a las personas para obtener información personal, financiera y/o confidencial. La información suministrada por la víctima es posteriormente utilizada por los atacantes para acceder a sus cuentas, robar información financiera o realizar actividades maliciosas en su nombre [1].

En un contexto donde los diferentes servicios se están apoyando cada vez más en el uso de los servicios de mensajería, como el SMS o WhatsApp, para divulgar información a sus usuarios o bien como mecanismo complementario en la autenticación para acceder a dichos servicios, se pronostica que la tendencia de ataques del tipo smishing continuará en aumento.

Adicional a las diversas campañas de concientización que se pueden realizar para divulgar buenos hábitos en los usuarios para prevenir este tipo de ataques, como ya lo hacen algunas entidades financieras [2]-[3], es esencial abordar este problema desde una perspectiva tecnológica, esto con el fin de mitigar el riesgo que presentan los usuarios en su día a día y facilitarles el reconocimiento de los mensajes que son legítimos. Es en este punto cuando la incorporación de sistemas de identificación de ataques smishing se presenta como una solución prometedora. Dicha identificación estará en la capacidad de analizar mensajes recibidos y enlaces sospechosos, de detectar patrones de suplantación y, finalmente, de indicar al usuario cuándo es seguro acceder al contenido compartido en un SMS o de un mensaje de WhatsApp [3].

Actualmente, varias revistas y portales de noticias resaltan el impacto del smishing como una técnica de estafa popular y clásica. El portal Xataka indica, un famoso caso de estafa haciéndose pasar por el servicio técnico de Netflix, indicando “hay un pago que ha fallado y qué debes volver a intentarlo. Lo hace mediante un SMS en el que adjuntan una dirección web que te lleva a la página del atacante, que es donde te van a intentar robar el dinero”. Este es solo uno de los ejemplos, donde grandes compañías son suplantadas y los atacantes buscan robar información de los clientes o directamente su dinero [4].

Según la revista ACIS, 9 de cada 10 personas leen los mensajes de texto que reciben, y cerca del 75% de los consumidores aceptan recibir información y notificaciones de las empresas mediante SMS, lo que indicarían mayor exposición a recibir información y poder engañar a las personas debido a su efectividad y simplicidad. Además, también este portal indica que el 90% de los mensajes son leídos dentro de los primeros 3 minutos luego de su recepción [5].

Esta información nos permite entender que los smishing es una técnica de ataque común debido a que su canal de comunicación es regularmente utilizado por los consumidores.

Además, la cantidad de información que reciben estos, podría ayudar a confundir y ser engañados rápidamente con smishing.

## II. Propuesta de solución

Según el problema previamente presentado se diseñará e implementará una solución con la cual los usuarios de telefonía móvil, que pueden ser víctimas de smishing, puedan verificar de forma ágil y sencilla si un mensaje recibido es legítimo y si su contenido es malicioso. Esta solución podrá ser utilizada mediante una interfaz de usuario, en donde los usuarios podrán reenviar el contenido del mensaje recibido. La solución se encargará de realizar el análisis del contenido y posteriormente su clasificación apoyándose en modelos de Machine Learning y servicios externos de análisis de malware. Finalmente, los usuarios recibirán en la misma interfaz de usuario, la respuesta con el resultado obtenido luego de que la solución realice la clasificación y verificación del contenido del mensaje enviado por el usuario.

### *A. Objetivo de la solución*

Diseñar e implementar una solución tecnológica de análisis e identificación de SMS maliciosos basándose en su contenido y patrones conocidos, con el propósito de mitigar el riesgo de ataques del tipo smishing y de proporcionar a los usuarios de telefonía móvil una herramienta confiable para el reconocimiento de mensajes maliciosos.

### *B. Arquitectura de la solución*

En la Figura 1. se ilustra la arquitectura de la solución la cual busca proporcionar una manera eficiente de poder recibir los mensajes de texto de los usuarios, para posteriormente preprocesarlos a través de Lambda y poder utilizar fuentes externas para consultar las URLs incluidas en los mensajes de texto, y paralelamente procesar el mensaje de texto con el Modelo de ML previamente seleccionado para detectar si ese mensaje pueda ser malicioso o no.

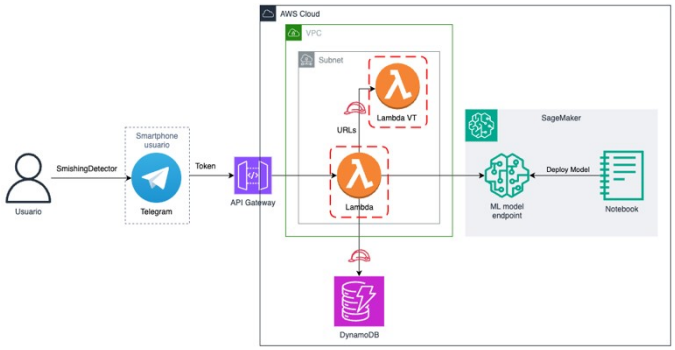


*Figura 1. Arquitectura en AWS.*

Del mismo modo, en la Figura 2. se puede visualizar cómo es el flujo de comunicación entre los diferentes componentes durante la ejecución de la solución:

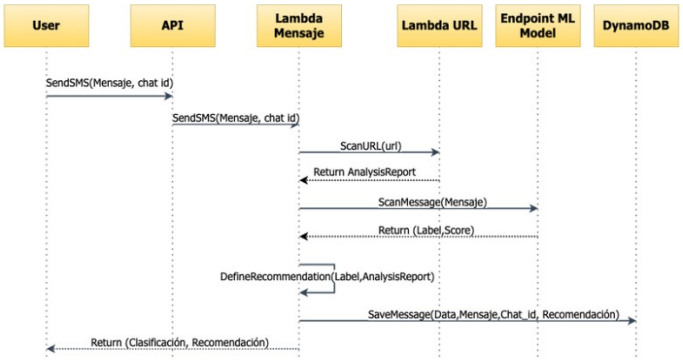


*Figura 2. Flujo comunicación entre componentes*

### *C. Niveles de criticidad*

Se incluirá la definición de los niveles de criticidad utilizados por la solución para enviar una respuesta al usuario luego de que este le envíe un mensaje de texto al sistema para que sea analizado.

#### *1) Mensaje Legítimo:*

**Características**: Los mensajes legítimos son aquellos que no representan ninguna amenaza y son enviados por fuentes confiables.

**Criterio**: El sistema asignará esta categoría al mensaje cuando el resultado en la clasificación del contenido resulte ser propia de un mensaje legítimo y cuando el análisis de dominio también indica que no es malicioso o bien, cuando el sistema no identifica alguna URL en el contenido del mensaje.

**Ejemplos**: Mensajes de bancos, instituciones gubernamentales, empresas reconocidas y contactos de confianza.

**Recomendaciones**: Estos mensajes pueden ser entregados al usuario sin ningún tipo de advertencia y se consideran seguros.

#### *2) Mensaje sospechoso:*

**Características**: Los mensajes sospechosos son aquellos que generan dudas sobre su autenticidad, pero no pueden ser confirmados como maliciosos de inmediato.

**Criterio**: El sistema asignará esta categoría al mensaje cuando el resultado del sistema no es concluyente y se requiere de una acción adicional del usuario para determinar la autenticidad del mensaje. Esto se puede presentar cuando la clasificación indique que se trata de un mensaje legítimo pero que contenga una URL cuyo análisis indica que no se puede concluir si el dominio es o no malicioso, o bien, cuando el resultado de la clasificación del contenido indique que es malicioso al tener los patrones comunes en mensajes utilizados en ataques del tipo Smishing pero que el análisis del dominio sea inconcluso o imposible de realizar. Se envía esta categoría para prevenir que los usuarios descarten mensajes legítimos de fuentes confiables.

**Ejemplos**: Mensajes que contienen enlaces no reconocidos, solicitudes inusuales o información inconsistente.

**Recomendaciones**: Se debe alertar al usuario sobre la posible sospecha y sugerir precaución al interactuar con el mensaje. Puede recomendarse una verificación adicional antes de seguir cualquier enlace o proporcionar información.

*3) Mensaje malicioso:*

**Características:** Los mensajes maliciosos son aquellos que representan una amenaza clara para la seguridad del usuario. Por lo general, intentan engañar o estafar al receptor.

**Criterio**: El sistema asignará esta categoría al mensaje cuando la clasificación del contenido del mensaje y el análisis del dominio indican que el mensaje es malicioso o bien, cuando se sabe que el dominio es malicioso a pesar de que el resultado de clasificación del mensaje indique lo contrario.

**Ejemplos**: Mensajes que solicitan información personal o financiera sensible, contienen enlaces a sitios de phishing o intentan suplantar identidades de organizaciones legítimas.

**Recomendaciones**: Estos mensajes deben ser identificados y bloqueados de inmediato para proteger al usuario. Se debe proporcionar una advertencia clara y no permitir que el usuario interactúe con el contenido peligroso.

### *D. Implementación de la solución*

A continuación, se presenta de forma detallada la metodología utilizada para realizar la implementación de los diferentes componentes indicados en la arquitectura de la solución para poder prestar el servicio de detección de mensajes fraudulentos a los usuarios finales.

*1) Bot Telegram*

Para facilitar la accesibilidad y mejorar la experiencia del usuario final en la interacción con el sistema diseñado, se optó por utilizar un aplicativo de mensajería instantánea como interfaz gráfica. En este caso, el aplicativo Telegram proporciona funcionalidades para realizar la creación y el diseño de bots que permitan a los usuarios interactuar con servicios web específicos. El uso de un bot de Telegram proporcionará a los usuarios una interfaz intuitiva, de fácil acceso y compatible con diversos sistemas operativos para acceder y aprovechar el servicio que ofrece nuestro sistema.

*2) API Gateway y Lambda*

Antes de crear el endpoint público del API Gateway en AWS que va a permitir la interacción entre el bot de Telegram con el backend de nuestra solución de una forma segura y escalable, es indispensable crea una función AWS Lambda, la cual va a contener el código que estará almacenado en la nube y será ejecutado cada vez que un usuario envíe un mensaje de texto por medio del bot de Telegram.

*3) Almacenamiento del mensaje*

Una vez se analiza el mensaje, el sistema almacena el mensaje y la respuesta que se entrega al usuario en una base de datos de DynamoDB. Amazon DynamoDB es una base de datos NoSQL de clave-valor sin servidor y completamente administrada. En esta base datos se almacena la información del chat, un id, el mensaje recibido y el análisis que proporciona el sistema tanto de texto, como de la URL, y la respuesta final que entregó.

*4) API Virus Total (VT)*

El API de VirusTotal es utilizada para validar las URLs contenidas en los mensajes, virustotal.com que es una plataforma antivirus que funciona integrando más de 50 antivirus más conocidos y más de 60 motores de detección. De esta manera las URL embebidas podrán ser validadas mediante el llamado al API de VT.

Para separar esta funcionalidad, se crea una lambda que realiza el llamado al API mediante un token que permite la autenticación del sistema y su consumo. De este API se consumen 2 funciones relacionadas con URLs:

- **Scan URL** - Esta función permite crear un análisis de una URL o archivo enviado a VirusTotal. Esta función recibe como parámetros la URL y el API key.
- **Get a URL analysis report** - Esta función permite obtener el reporte previamente generado con el fin de leer los resultados. De este reporte se toma la variable stats, la cual resume los resultados de los distintos antivirus. Basado en esta información se define un rango de acuerdo con el número de reportes de URL, como maliciosa, sospechosa o inconclusa.

*5) Amazon SageMaker*

Dado que el sistema requiere de un modelo de Machine Learning para realizar la clasificación del mensaje basado en su contenido, Amazon SageMaker resulta ser el servicio de AWS adecuado dado que permite construir, entrenar y desplegar los modelos de forma rápida y sencilla. Para efectos del presente proyecto, la función Lambda se deberá encargar de obtener el mensaje de texto enviado por el usuario y de consumir el servicio proporcionado por el endpoint de SageMaker una vez el modelo previamente entrenado esté listo para realizar predicciones.

*6) Modelo de clasificación*

*a) Dataset de entrenamiento*

Dentro de la solución también se tiene como objetivo analizar el modelo de ML que permita clasificar los Mensajes de texto correctamente.

En lo que respecta al dataset “SMS phishing dataset for machine learning and pattern recognition” de Mendeley Data [6], se encuentra que contiene 5971 mensajes de texto etiquetados en tres categorías: legítimo (Ham), Spam o Smishing, siendo 4844 mensajes legítimos, 489 mensajes spam y 638 mensajes Smishing. Un punto para destacar de este dataset es que incluye la extracción de características que son útiles para el entrenamiento y la evaluación de modelos de Machine Learning, dichos atributos están asociados a la presencia de URL, correo electrónico o número de teléfono en el mensaje de texto. Sin embargo, para efectos del alcance del presente proyecto, se encuentra que este dataset tiene pocas referencias públicas sobre su uso en el entrenamiento de modelos de Machine Learning, por este motivo, se toma le decisión de mantener los ejemplos de mensajes de texto de este dataset como muestras que pueden ser utilizadas en las pruebas de un modelo entrenado con un dataset distinto cuya aplicabilidad tenga una reputación conocida. Sin duda, los ejemplos de mensajes de Smishing que contienen URL son de gran ayuda para la evaluación del modelo.

Ahora bien, en lo que respecta al dataset “SMS spam Collection” [7], contamos con un conjunto de mensajes de texto etiquetados y recopilados específicamente para la investigación sobre spam de SMS. Este dataset contiene 5574 mensajes en

inglés, clasificados en dos categorías: legítimos (ham) o spam. Los mensajes de spam (425) fueron extraídos manualmente del sitio web Grumbletext, un foro del Reino Unido donde los usuarios denuncian mensajes de spam. Para los mensajes legítimos, se incluyeron 3375 mensajes recopilados en la Universidad Nacional de Singapur y también se agregaron 450 mensajes legítimos de la tesis doctoral de Caroline Tag, y del SMS Spam Corpus v.0.1 Big, con 1002 mensajes legítimos y 322 mensajes de spam. A diferencia del dataset de Mendeley Data [6], este dataset [7] tiene una gran cantidad de referencias publicadas sobre su usabilidad en el entrenamiento y evaluación de modelos de Machine Learning para la clasificación de mensajes y la prevención de ataques del tipo Smishing, por este motivo, para el presente proyecto, se optó por utilizar este dataset en el entrenamiento del modelo que utilizará el prototipo de la solución.

Una vez seleccionado el dataset [7] adecuado para entrenar el modelo, otros trabajos de investigación [8][9] demuestran el motivo por el cual es necesario realizar un análisis de datos para identificar las características y patrones relevantes de los mensajes que están etiquetados como legítimos y maliciosos. El análisis exploratorio de datos (EDA por sus siglas en inglés), es el proceso en el cual el analista de datos es capaz de visualizar y entender los datos con los que cuenta y a partir de esto crear hipótesis que le permitan definir y refinar los modelos que se van a entrenar con estos datos. De este modo, se puede llegar a omitir mensajes que contengan características que no representen alguna contribución a la predicción del modelo entrenado, lo cual permite que se refine el conjunto de mensajes utilizados en el entrenamiento para obtener predicciones más acertadas y, finalmente, a lograr obtener una representación en vectores numéricos de longitud fija de los mensajes enviados por los usuarios, los cuales van a facilitar el entrenamiento de los modelos.

Por ejemplo, haciendo un recuento de las etiquetas contenidas en el dataset, tal como se ve representado en la Figura 3, se encuentra una proporción desbalanceado de mensajes de spam (13.4%) versus lo que son legítimos (86.6%). Esto permite identificar que es probable que los modelos utilizados en el presente trabajo requieran hacer uso de técnicas de re-sampling.

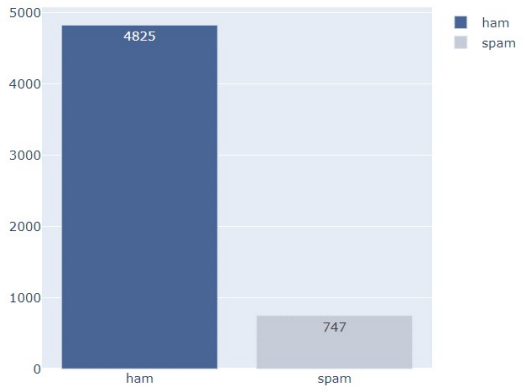


*Figura 3. Cantidad de muestra por categoría en dataset.*

Por otro lado, realizando un recuento de la cantidad de caracteres utilizados en mensajes etiquetados como spam, a partir de los histogramas presentados en la Figura 4 y la Figura 5, se encontró que este tipo de mensaje tiende a utilizar una mayor cantidad de caracteres que los mensajes legítimos. Esto es una característica interesante ya que el modelo de clasificación puede utilizar esta característica para emitir una predicción simplemente con base en la longitud del mensaje, por ejemplo, si los mensajes tienen muy pocos caracteres, es más probable que sean clasificados como legítimos. A continuación, se presentan los histogramas obtenidos en los trabajos de investigación utilizados como referencia [9]:

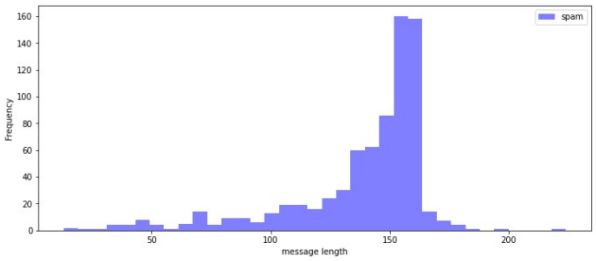


*Figura 4. Histograma cantidad caracteres categoría malicioso*

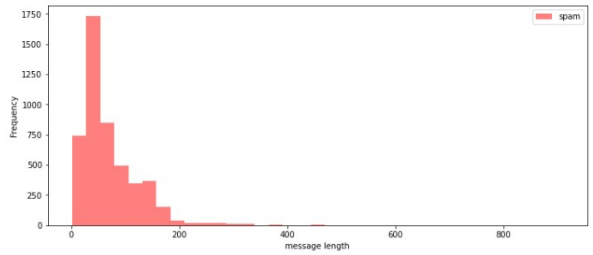


*Figura 5. Histograma cantidad caracteres categoria legítimo*

Del mismo modo, los trabajos también realizan gráficos Word Cloud representados en la Figura 6 y la Figura 7, los cuales permiten visualizar cuáles son las palabras que más se repiten en los mensajes del dataset dependiendo la categoría a la que pertenecen. Por ejemplo y, teniendo en cuenta que el dataset solo cuenta con muestras en inglés, para el caso de mensajes legítimos se identifica las palabras “u”, “come”, “got”, “ur”, “go” y “want” como las que más se repiten en los mensajes. Por otro lado, para el caso de mensajes spam se identifican las palabras “call”, “free”, “text”, “mobile” y “reply” como las más comunes. Este patrón es de vital importancia para el entrenamiento de los modelos dado que, como se mencionaba anteriormente, la intención es crear vectores numéricos de longitud fija que representen los mensajes de entrenamiento y/o los mensajes enviados por los usuarios y, para crear dichos vectores, se utilizan las palabras más frecuentes para poder realizar una predicción.

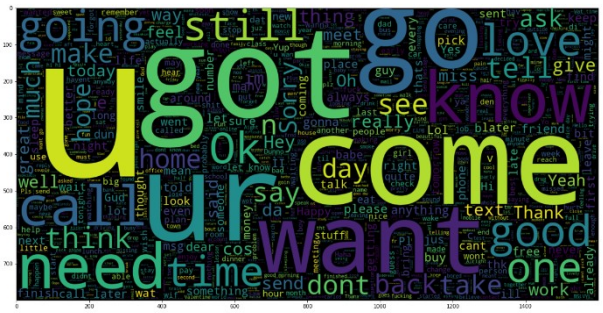

*Figura 6. Wordcloud categoria legítimo*

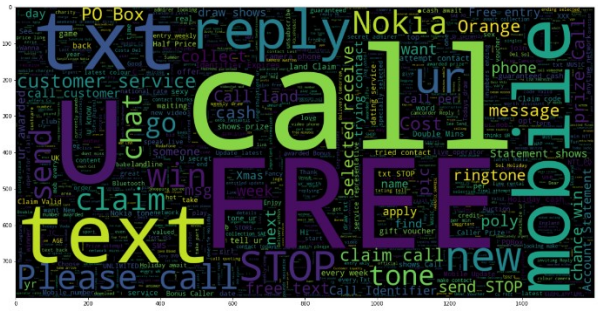


*Figura 7. Wordcloud categoría malicioso*

Finalmente, el análisis de los mensajes del dataset, indica que alrededor de 68% de los mensajes que contienen una URL corresponden a mensajes que fueron etiquetados como spam, es decir, que aún sin identificar si se trata de URL acortadas, se puede concluir que, en la mayoría de los casos, si el mensaje incluye una URL, se trata de un mensaje fraudulento.

Es importante mencionar que, adicional a los patrones y características obtenidas por medio de un análisis estadístico del dataset, existen otras características de forma que resultan ser importantes para que un usuario final pueda identificar un mensaje de texto fraudulento aún sin precisar de un sistema de detección como el que se plantea en el presente trabajo.

*b) Selección del modelo de ML*

Para desarrollar y analizar el mejor modelo de ML, se prepararon los datos de la siguiente manera:

- Eliminar las tildes de los mensajes de texto ya que esto resulta ser útil para normalizar el texto y asegurar que no haya caracteres especiales.
- Transformar las muestras del dataset en una representación de cada mensaje en palabras (tokens). Esta operación se conoce como tokenización y es común en el procesamiento del lenguaje natural. Tal como se presenta a continuación en la Figura 8:

| | Target | Mensaje | tokens_Mensaje | Texto_Mensaje |
|---|---|---|---|---|
| 0 | 1 | Try to get in into the following link to get a... | [Try, get, following, link, get, price] | Try get following link get price |
| 1 | 1 | Please give your information as soon as possible | [Please, give, information, soon, possible] | Please give information soon possible |
| 2 | 0 | Go until jurong point, crazy.. Available only ... | [Go, jurong, point, ,, crazy, .., Available, b... | Go jurong point , crazy .. Available bugis n g... |
| 3 | 0 | Ok lar... Joking wif u oni... | [Ok, lar, ..., Joking, wif, u, oni, ...] | Ok lar ... Joking wif u oni ... |
| 4 | 1 | Free entry in 2 a wkly comp to win FA Cup fina... | [Free, entry, 2, wkly, comp, win, FA, Cup, fin... | Free entry 2 wkly comp win FA Cup final tkts 2... |
| ... | ... | ... | ... | ... |
| 5569 | 1 | This is the 2nd time we have tried 2 contact u... | [This, 2nd, time, tried, 2, contact, u, ., U, PS... | This 2nd time tried 2 contact u. U PS750 Pound... |
| 5570 | 0 | Will u b going to esplanade fr home? | [Will, u, b, going, esplanade, fr, home, ?] | Will u b going esplanade fr home ? |
| 5571 | 0 | Pity, * was in mood for that. So...any other s... | [Pity, ,, *, mood, ., So, ..., suggestions, ?] | Pity , * mood . So ... suggestions ? |
| 5572 | 0 | The guy did some bitching but I acted like i'd... | [The, guy, bitching, I, acted, like, 'd, inter... | The guy bitching I acted like 'd interested bu... |
| 5573 | 0 | Rofl. Its true to its name | [Rofl, ., Its, true, name] | Rofl . Its true name |

*Figura 8. Tokenización muestras del dataset*

- Eliminar palabras de parada (stopwords) en inglés en cada una de las muestras del dataset. Las palabras de parada son palabras comunes que a menudo no aportan información significativa para tareas de procesamiento de texto, por ende, eliminarlas aumenta considerablemente la precisión de los modelos.
- Determinar que el 20% del dataset se utilizará como muestras de prueba, mientras que el 80% se utilizará
- como muestras de entrenamiento.
- Convertir el dataset desde una representación de texto a un formato de cadenas numéricas (matrices TF-IDF) que pueden ser utilizadas como entrada para modelos de Machine Learning. La matriz TF-IDF asigna un peso a cada palabra en función de su frecuencia en un documento específico y su frecuencia inversa en el conjunto de documentos, permitiendo identificar las palabras más relevantes.

Este análisis se realizó luego de llevar a cabo el entrenamiento de los siguientes modelos de clasificación:

- Support Vector Machine (SVM)
- Logistic regression
- Random Forest (RF)
- Multilayer perceptron (MLP)
- Naive Bayes
- XGBoost

Posteriormente, la evaluación de los modelos entrenados para permitir identificar cuál resulta ser el más adecuado para el presente caso de uso. Para esto, se construye un informe de clasificación, el cual resulta ser útil para comprender cómo se desempeñan los modelos en cada categoría (en este caso, 'legítimo' y 'spam') y para identificar posibles desafíos, como desequilibrios de clases. Las métricas incluidas en el informe de clasificación son las siguientes:

- **Precision:** La proporción de instancias clasificadas como positivas que son verdaderamente positivas.
- **Recall (Sensibilidad): La** proporción de instancias positivas que fueron correctamente clasificadas.
- **F1-score:** La media armónica de precision y recall. Es útil cuando hay un desequilibrio entre las clases.
- **Support:** El número de instancias de cada clase en el conjunto de prueba.

Finalmente, se obtienen los resultados presentados en la Figura 9, en donde para el presente trabajo, se utiliza el indicador F1-Score para elegir el modelo con mejor desempeño, teniendo en cuenta el desbalance presentado en las dos categorías del dataset de entrenamiento.

| *Modelo* | Avg. Precision | Avg. Recall | F1-Score |
|---|---|---|---|
| *Support Vector Machine (SVM)* | 0.98 | 0.92 | 0.95 |
| *Logistic regression* | 0.96 | 0.87 | 0.91 |
| *Random Forest (RF)* | 0.99 | 0.92 | 0.95 |
| *Multilayer perceptron (MLP)* | **0.99** | **0.95** | **0.97** |
| *Naive Bayes* | 0.98 | .089 | 0.93 |
| *XGBoost* | 0.98 | 0.93 | 0.95 |

*Figura 9. Resultados informe precisión modelos entrenados*

*c) Resultados*

Revisando los resultados en la evaluación de los diversos modelos entrenados con el dataset de prueba, se encontró que el

modelo MultiLayer Perceptron (MLP) es el que tiene mejor rendimiento de forma global. Este modelo es un tipo de red neuronal alimentada hacia adelante, lo que significa que la información se mueve a través de la red en una dirección, desde la capa de entrada (cada nodo en esta capa representa una característica única del conjunto de datos) hasta la capa de salida (donde la cantidad de nodos en esta capa depende del tipo de problema, en este caso, se tiene dos resultados posibles, por ende, se tendrán dos nodos), sin ciclos ni retroalimentación [10].

*d) Transformer BERT*

Teniendo en cuenta que el modelo con mejor rendimiento fue el de MLP, se decidió profundizar en un modelo basado en redes neuronales llamado BERT (Bidirectional Encoder Representations from Transformers). Este es un modelo desarrollado por Google, el cual fue introducido en un artículo de investigación en 2018 y se ha convertido en una de las arquitecturas más influyentes en el campo del procesamiento del lenguaje natural (NLP). Este es un modelo previamente entrenado con grandes cantidades de datos no etiquetados, como texto en varios idiomas, y que luego puede ser ajustado para tareas específicas de NLP mediante un proceso llamado "finetuning" en conjuntos de datos más pequeños y etiquetados para esas tareas particulares. BERT y los MLP (Multilayer Perceptrons) están relacionados dado que ambos se basan en implementaciones a partir de las redes neuronales, con la diferencia de que BERT, al igual que otros modelos basados en transformers, utilizan una forma más avanzada de red neuronal que ha superado en rendimiento a muchos modelos anteriores, incluyendo algunos basados en MLP, en tareas específicas de NLP. Es decir, los MLP son una arquitectura más tradicional en comparación con los transformers y BERT [10].

## III. Evaluación de la solución

Con el fin de validar la solución se realizaron pruebas mediante 3 dispositivos diferentes teniendo como resultado, respuestas iguales, de tal manera que el dispositivo no afecta la respuesta del modelo, realizando más de 100 pruebas al sistema, como se ilustra en la Figura 10, con una concurrencia máxima de 7 invocaciones al mismo tiempo.

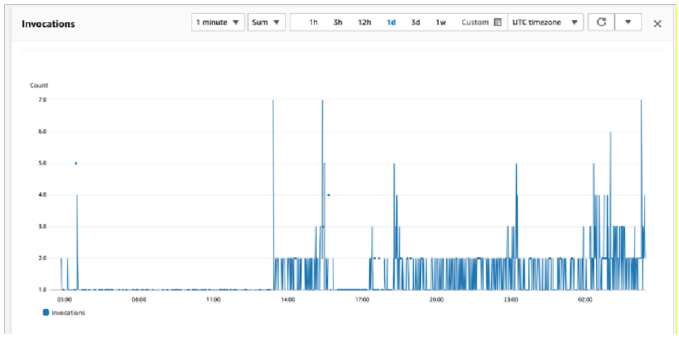


*Figura 9. Recuento invocaciones de la solución.*

Posteriormente se mapea los resultados esperados con los mensajes de texto y las URLs seleccionadas. A continuación, se listan algunos puntos importantes como resultado:

- De un total de 13 mensajes seleccionados, 2 se clasificaron incorrectamente, lo cual corresponde al 26% de las pruebas. El porcentaje de mensajes correctamente clasificados es de 74%.
- Este 26% se analizó y se encontró que algunas URLs, no fueron correctamente detectadas por la solución y/o VirusTotal no pudo validar se criticidad.
- El 100% de los mensajes de texto que no contienen URLs, se clasificaron correctamente.
- Los mensajes que contienen más de 1 URL en el mensaje pueden generar errores a la hora de identificarse por el sistema.
- Algunas puntuaciones como el uso de / o. (Punto), pueden generar falsas URLs que proporcionan errores al sistema.
- El tiempo promedio de respuesta y procesamiento del mensaje es de 20 segundos, tal como se presenta en la Figura 10 a continuación. Esto se debe a que los mensajes que no contienen URLs solo usan el modelo de ML para definir la clasificación del mensaje, agilizando asi el tiempo de respuesta del usuario final. Esto es un buen indicador ya que para mensajes sencillos(solo contienen texto) esta por debajo de lo esperado para este sistema que es de 1 minuto.

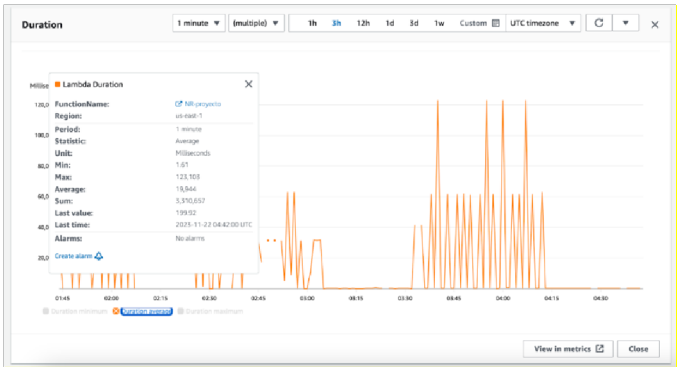


*Figura 10. Tiempo respuesta de la solución.*

## IV. Gestión del proyecto

Con el fin de lograr el desarrollo, la liberación a producción y el mantenimiento de la solución, se requiere de un equipo conformado por diversos roles que puedan contribuir con sus habilidades y conocimientos específicos para alcanzar los objetivos del proyecto. La Figura 11. representa el organigrama con los roles que son relevantes para poder conformar una compañía que sea capaz de proveer la solución propuesta para los clientes finales.

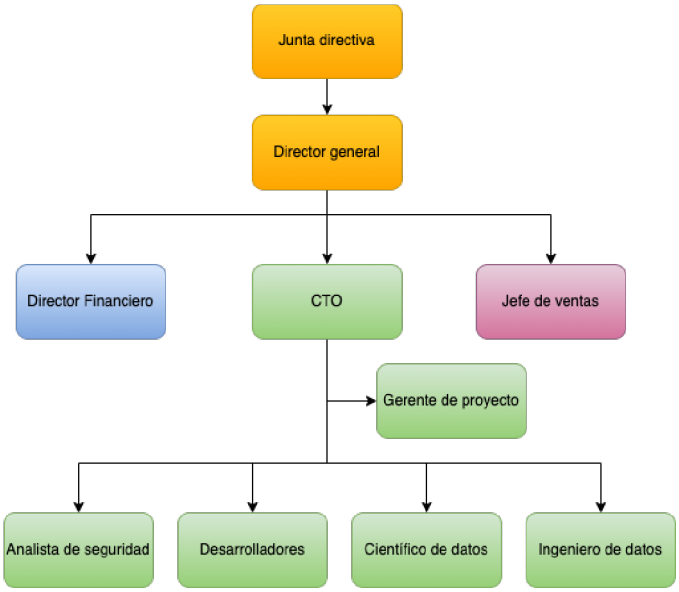


*Figura 11. Organigrama*

### *A. Equipo de trabajo*

El organigrama presenta roles que son esenciales para mantener una compañía que provee la solución a los usuarios finales y clientes corporativos, como lo son el jefe de ventas, que se encargará de promover la solución en el mercado, y, por otro lado, el director financiero, el cual será clave para evaluar los diversos modelos de negocio propuestos y de monitorear rigurosamente los estados financieros de la compañía para asegurar que se tiene un negocio rentable y sostenible en el tiempo. Para efectos del presente trabajo, nos enfocaremos en los roles específicos para la gestión, el desarrollo, la liberación y el mantenimiento de un proyecto cuando alguno de los clientes adquiere la solución, los cuales están identificados como los roles que conforman el nivel del organigrama, siendo todos estos los que reportan directamente al Chief Technology Officer (CTO).

### *B. Metodología de gestión de proyectos*

Scrum es la metodología de gestión de proyectos más adecuada para la ejecución de proyectos de implementación de la solución. En la planificación de los proyectos se establecen "sprints" con una duración de 2 semanas, de modo tal que se pueda responder de forma ágil frente a eventuales cambios y se mantenga una constante revisión de la evolución del proyecto y del cumplimiento de los requerimientos.

### *C. Riesgos del proyecto*

Con base en el contexto en el que se desarrolla la propuesta de valor de la solución, las tecnologías involucradas en su desarrollo y los requerimientos cambiantes por nuevas necesidades del mercado, se presentan a continuación los riesgos identificados para el presente proyecto:

1. Necesidad de incrementar el número de iteraciones previamente proyectado en la planeación del proyecto debido a la necesidad de corregir la efectividad del modelo para la detección de mensajes fraudulentos

2. Cambios a nivel técnico realizados por los proveedores de servicios externos (AWS y VirusTotal) que impacten directamente los requerimientos para llevar a cabo la integración de los componentes de la arquitectura.

3. Interrupciones de servicio por parte de los proveedores de los servicios proporcionados por terceros.

4. Liberación de regulaciones para el manejo de datos que privados que puedan alterar los requerimientos técnicos de la solución para asegurar que cumpla con lo que dicta la regulación.

5. Incremento en costos debido a la necesidad de escalar la solución por un aumento en el volumen de mensajes de usuario y usuarios que se deben soportar, con el fin de mantener el mismo nivel de servicio.

6. La evolución constante de las tácticas de smishing, de soluciones tecnológicas y otras amenazas de seguridad, representan un riesgo debido a que la solución debe poder adaptarse y responder a los nuevos requerimientos que están impuestos por estas.

Se realiza el mapa de calor presentado en la Figura 12 para evaluar la prioridad con la que se debe procurar mitigar los riesgos identificados, encontrando que los riesgos 5 y 6 son los más críticos, seguidos por el riesgo 1, tal como se presenta a continuación:


*Figura 12. Mapa calor riesgos identificados.*

## V. Análisis financiero del proyecto

Esta solución cuenta con 2 tipos de clientes.

- Usuarios que buscan validar mensajes de texto personales, los cuales cuenta con un teléfono móvil.
- Organizaciones que deciden proveer soluciones a sus clientes/usuarios para mitigar el riesgo de que sean víctimas de ataques del tipo Smishing, acá entran las instituciones financieras, los mismos proveedores de servicios de telecomunicaciones, organizaciones de la salud, gobiernos u organizaciones gubernamentales, etc. Para este último tipo de cliente, es importante mencionar que un caso eventual en el que entre en vigor alguna regulación de protección y manejo de datos personales será un factor relevante para que se impulse la necesidad de contar con una solución como la que se propone en el presente trabajo.

*1) Modelo de negocio*

Con base en los usuarios y servicios ofrecidos por la solución, se definen el siguiente modelo de negocio:

- Cada usuario que desee utilizar el bot, podrá utilizar la solución y/o validar hasta 10 mensajes de texto semanales. Posteriormente, el usuario puede acceder a un modelo por suscripción para poder analizar mensajes ilimitados.
- En segundo lugar, para las empresas que deseen tener su propio bot, donde la información que se procesa y almacena pertenezca netamente a la organización usuaria. Se propone utilizar un modelo de licenciamiento y/o pago por aplicación con el fin de proporcionar una solución aislada y de único uso para esa empresa.

*B. Análisis financiero del proyecto*

Para el análisis financiero se realiza una proyección a 5 años, incluyendo gastos variables como la infraestructura de nube y gastos fijos como lo salarios del personal necesario.

Dentro de los costos de la aplicación, se realizó una estimación para los costos de ML, infraestructura, almacenamiento y seguridad que requiere el sistema. Estos costos se definen con una calculadora de AWS, lo cual dio un estimado de $ 65.510.256 COP. y un costo de licenciamiento de VT de $ 1.050.456 COP anual.

En la Figura 13. se evidencia la gráfica que representa cómo los distintos servicios de la solución se requieren y proyectan, junto con la fuerza laboral que se requiere.

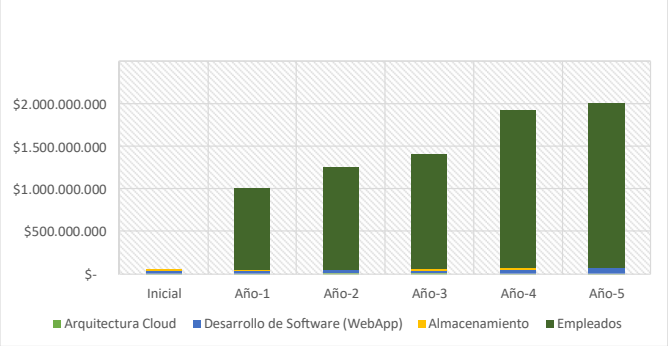


*Figura 13. Proyección de costos*

Adicionalmente, con la información de los gastos que se estiman para la aplicación se decidió realizar un análisis de mercado incluyendo el número potencial de usuarios de esta aplicación y el precio respecto a la competencia. Con esta información se calculó el Mercado potencial (Q = n x p x q) de la siguiente manera:

- n = número de usuarios de telefonía móvil en Colombia y con smartphone para poder descargar la aplicación de Telegram.
- p = Precio de referencia respecto a la competencia.
- q = consumo per cápita de media.

**Mercado Potencial**

Q = n x p x q

n = 82.200.000 -> 4'110.000
p = $5.000
q = 0,04

Q = $1'780.000.000

Con este cálculo se permite obtener el margen bruto, restando los costos de solución previamente definidos.

**Margen bruto**

MB = Ingresos Totales - COGS(Costos)
MB = $1'780.000.000 - $966.000.000
= $814.000.000

%MB = MB / Ingresos Totales *100%
% MB = $814.000.000/ $1'780.000.000 * 100%
0,46%

Esta información permite determinar el punto de equilibrio (Costos/%Margen Bruto), el cual define el momento en que los ingresos de una empresa cubren sus gastos fijos y variables.

**Punto de Equilibrio**

P. E. = Costos / % de margen bruto

$$P.\ E. = \frac{\$966.000.000}{0{,}46\%}$$

P.E. = $2.112.383.292

El punto de equilibrio y la estimación de costos anual permiten indicar que el punto de equilibrio se obtiene al superar el segundo año de ventas con solo suscripciones individuales.

## VI. Conclusiones

Finalmente, luego de llevar a cabo el diseño, implementación y evaluación de la solución propuesta para mitigar el problema identificado, se concluye lo siguiente:

- La elección de utilizar Telegram como interfaz gráfica para los usuarios finales proporciona una experiencia intuitiva y de fácil acceso, mejorando la accesibilidad y la experiencia del usuario en la interacción con el sistema.

- Se destaca la importancia del análisis exploratorio de datos para identificar características y patrones

relevantes en los mensajes que conforman los datasets de entrenamiento y prueba ya que este proceso permite depurar el conjunto de datos disponibles, así como generar hipótesis que puedan ayudar en el proceso de diseño de los modelos.

- A pesar de la elección del MLP como el mejor modelo, se encuentra que hoy en día existen modelos avanzados como BERT (Bidirectional Encoder Representations from Transformers), el cual ofrece grandes ventajas en el procesamiento del lenguaje natural (NLP) al tratarse de un modelo preentrenado con grandes cantidades de datos no etiquetados que puede ser afinado para tareas específicas de NLP.

- El proyecto actual se enfoca en un prototipo (MVP) para evaluar la viabilidad de la solución. Se reconoce la necesidad de integrar elementos en la arquitectura que permitan escalar la solución en caso de una implementación a mayor escala. La escalabilidad es crucial para manejar flujos considerables de mensajes en producción.

- En caso de mantenerse un modelo supervisado como eje principal de la solución, se destaca la importancia de planificar futuros reentrenamientos del modelo para adaptarse a nuevas tácticas y patrones de ataques smishing.

- Es evidente la necesidad de implementar un flujo en el bot de Telegram que permita obtener retroalimentación de los usuarios. Esta retroalimentación sería valiosa para evaluar la precisión y eficacia del sistema, así como para ajustar la base de conocimiento creada por el sistema.

- Las URLs que están incluidas como parte del alcance de esta solución, proporcionaron un nivel de incertidumbre mayor a la hora de clasificar los mensajes, esto se debe modelo de extracción de URLs y la herramienta utilizada para su análisis, VirusTotal. Parece pertinente según las pruebas realizadas, verificar otras herramientas externas que permitan mejorar tanto la extracción como el análisis de URLs contenidas en el mensaje sms.

## REFERENCIAS


[1] IBM, “What is smishing (SMS phishing)?” [Internet]. Disponible en: https://www.ibm.com/topics/smishing#:~:text=Smishing%20is%20a%20social%20engineering,messages%E2%80%94and%20%E2%80%9Cph ishing.%E2%80%9D. Acceso: 12 de agosto de 2023.

[2] Bancolombia. “Smishing: Mensajes de texto engañosos” [Internet]. Disponible en: https://www.bancolombia.com/educacionfinanciera/seguridad-de-la-informacion/smishing. Acceso: 20 de agosto de 2023.

[3] BBVA. “'Phishing', 'vishing', 'smishing', ¿qué son y cómo protegerse de estas amenazas?” [Internet]. Disponible en: https://www.bbva.com/es/innovacion/phishing-vishing-smishing-queson-y-como-protegerse-de-estas-amenazas/. Acceso: 20 de agosto de 2023.

[4] Xataka. “Estafa por sms del falso pago fallado a netflix: Qué Es, cómo funciona y cómo identificar y evitar este timo,” ´ [Internet] Disponible en: https://www.xataka.com/basics/estafa-sms-falso-pago-fallado-a-netflixque-como-funciona-como-identificar-evitar-estetimo#:~:text=Su%20nombre%20significa%20SMS%20phishing,identifi car%20que%20es%20una%20estafa. Acceso: 27 de agosto de 2023.

[5] ACIS. “¡cuidado! Las Cifras de Hurto a través de mensajes de texto cada vez son más frecuentes en el país” [Internet]. Disponible en: https://www.acis.org.co/portal/content/%C2%A1cuidado-las-cifras-dehurto-trav%C3%A9s-de-mensajes-de-texto-cada-vez-son-m%C3%A1sfrecuentes-en-el. Acceso: 27 de agosto de 2023.

[6] Mendeley. “SMS phishing dataset for machine learning and pattern recognition” [Internet]. Disponible en: https://data.mendeley.com/datasets/f45bkkt8pr/1. Acceso: 26 de agosto de 2023.

[7] Kaggle. “SMS Spam collection dataset.” Disponible en: https://www.kaggle.com/datasets/uciml/sms-spam-collection-dataset. Acceso: 26 de agosto de 2023

[8] Jain AK, Gupta BB, Kaur K, Bhutani P, Alhalabi W, Almomani A. “A Content and URL analysis based efficient approach to detect smishing SMS in intelligent systems.” Int J Intell Syst. 2022;1 - 25. doi:10.1002/int.23035

[9] Kaggle. “Spam vs Ham prediction” [Internet]. Disponible en: https://www.kaggle.com/code/saipavansaketh/spam-vs-ham-prediction. Acceso: 19 de noviembre de 2023

[10] Science Direct. “The Digital Twin Paradigm for Smarter Systems and Environments: The Industry Use Cases” [Internet]. Disponible en: https://www.sciencedirect.com/topics/computer-science/multilayerperceptron#:~:text=Multi%20layer%20perceptron%20(MLP)%20is,inpu t%20signal%20to%20be%20processed. Acceso: 19 de noviembre de 2023.