

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INFORMÁTICA
SANTIAGO - CHILE



“SHERLOCK: A TOOL FOR DETECTING THE MINIMUM
ACCESS PERMISSIONS THAT AN APP REQUIRES USING
HIS PLAY STORE DESCRIPTION AND ITS CONTEXT AS AN
INPUT AND NATURAL LANGUAGE PROCESSING AS A
TECHNIQUE”

BENJAMÍN IGNACIO PIZARRO PIZARRO

MEMORIA PARA OPTAR AL TÍTULO DE
INGENIERO CIVIL EN INFORMÁTICA

Profesor Guía: Felipe Beroíza
Profesor Correferente: Julio Sotelo

Junio - 2024

DEDICATORIA

Para mi madre, Nancy Pizarro, y mi hermana, Jenniffer Laurie por su apoyo incondicional.

AGRADECIMIENTOS

La elaboración de este documento ha sido un proceso complejo que no habría sido posible sin la colaboración y el apoyo de varias personas. Quisiera aprovechar esta oportunidad para expresar mi gratitud a quienes me han brindado su ayuda y respaldo en diferentes etapas de este trabajo. En primer lugar, quiero expresar mi agradecimiento a mi madre, cuyo apoyo constante ha sido fundamental para la realización de este trabajo. Su dedicación y esfuerzo han sido invaluable para mi desarrollo académico y personal. A mi hermana, le agradezco por su compañía y disposición para ayudarme en todo momento. Su respaldo ha sido esencial para mantenerme enfocado y motivado durante este proceso. A mi pareja, quiero agradecerle por su comprensión y apoyo continuo. Su paciencia y aliento han sido cruciales para superar los desafíos que se presentaron a lo largo de este proyecto. Les agradezco sinceramente por su ayuda y apoyo. Este logro no habría sido posible sin su colaboración.

RESUMEN

Resumen— En la actualidad, con la masificación del uso de dispositivos inteligentes, existe un gran número de usuarios inexpertos en el área de las tecnologías. Con el fin de proporcionar más información a los usuarios que la necesitan, se desarrolló una herramienta que analiza las descripciones de las aplicaciones de la Play Store, con el objetivo de identificar los permisos que utilizarán dichas aplicaciones según las funcionalidades descritas. Para llevar a cabo el análisis de las descripciones, se empleó un clasificador de textos multi-etiqueta, utilizando el modelo preentrenado BERT, considerando dichos permisos como clases. A pesar de las limitaciones asociadas con los datos encontrados para el entrenamiento del clasificador, al realizar el análisis de permisos en 10 aplicaciones de la Play Store, se obtuvo un 50 % de falsos positivos. Además, se encontraron casos donde, debido al contexto de la aplicación, el clasificador predijo permisos que no estaban especificados en el archivo Manifiesto de la aplicación. Estos resultados, aunque no son perfectos, representan un avance hacia una herramienta que pueda concientizar al usuario común sobre los riesgos de las aplicaciones.

Palabras Clave— NLP, permisos, seguridad, Android.

ABSTRACT

Abstract— Currently, with the widespread use of smart devices, there is a large number of inexperienced users in the field of technology. In order to provide more information to users in need, a tool was developed to analyze the descriptions of applications on the Play Store, aiming to identify the permissions that such applications will use based on the described functionalities. To carry out the analysis of the descriptions, a multi-label text classifier was employed, using the pre-trained BERT model, with these permissions considered as classes. Despite limitations related to the data available for training the classifier, when analyzing permissions in 10 applications from the Play Store, a 50 % false positive rate was obtained. Additionally, there were cases where, due to the context of the application, the classifier predicted permissions that were not specified in the application's Manifest file. These results, although not perfect, represent progress towards a tool that can raise awareness among the general user about the risks of applications.

Keywords— NLP, permissions, security, Android.

GLOSARIO

ID: Identificador.

UTFSM: Universidad Técnica Federico Santa María.

SO: Sistema operativo.

EE.UU.: Estados Unidos.

APP: Aplicaciones.

NLP: Natural Language Processing. En español, Procesamiento de lenguaje natural.

API: Application Programming Interfaces. En español, Interfaces de programación de aplicaciones.

LSTM: Long short-term memory.

IA: Inteligencia artificial.

LSI: Índice Semántico Latente.

BERT: Bidirectional Encoder Representation from Transformers. En español, Representación bidireccional del codificador a partir de transformadores.

DP: Deep Learning. En español, Aprendizaje profundo.

ML: Machine Learning. En español, Aprendizaje automático.

GPU: Unidades de procesamiento gráfico.

BLEU: Bilingual Evaluation Understudy.

SDK: Software Development Kit. En español, Kit de desarrollo de software.

RAM: Random access memory. En español, Memoria de acceso aleatorio.

GB: Gigabytes.

CPU: Central Processing Unit. En español, Unidad de procesamiento central.

TPU: Tensor Processing Unit. En español, Unidad de procesamiento de tensor.

ÍNDICE DE CONTENIDOS

RESUMEN	IV
ABSTRACT	IV
GLOSARIO	VI
ÍNDICE DE FIGURAS	IX
ÍNDICE DE TABLAS	IX
CAPÍTULO 1: INTRODUCCIÓN	1
1.1 MOTIVACIÓN Y CONTEXTO	1
1.2 DEFINICIÓN DEL PROBLEMA	3
1.3 PROPUESTA DE SOLUCIÓN	5
1.4 OBJETIVOS	6
1.4.1 Objetivo general	6
1.4.2 Objetivos específicos	6
1.5 Estructura del documento	7
CAPÍTULO 2: MARCO CONCEPTUAL	8
2.1 MACHINE LEARNING	8
2.1.1 DEEP LEARNING	8
2.1.2 RNN	9
2.1.3 ENTRENAMIENTO DE MODELOS	9
2.2 PROCESAMIENTO DE LENGUAJE NATURAL	11
2.2.1 VISIÓN GENERAL	11
2.2.2 CURVAS NLP	12
2.2.3 ETAPAS DEL PRE-PROCESAMIENTO PARA NLP	14
2.2.4 MACHINE LEARNING EN NLP	16
2.2.5 MÉTRICAS DE EVALUACIÓN	18
2.3 HIPERPARÁMETROS	20
2.4 BERT	21
2.4.1 FINE-TUNNING	22
2.4.2 TOKENIZADOR DE BERT	23
2.4.3 RoBERTa	23
2.5 SISTEMA DE PERMISOS DE ANDROID	24
2.5.1 ESTRUCTURA SISTEMA DE PERMISOS ANDROID	24
2.5.2 PERMISOS ANDROID RIESGOSOS	28
2.6 ESTADO DEL ARTE	30
2.6.1 WHYPER	31
2.6.2 AUTOCOG	31
2.6.3 AC-NET	31
2.6.4 FCDP	32

2.6.5	SMARTPI	33
2.6.6	FIDEDROID	33
CAPÍTULO 3: PROPUESTA DE SOLUCIÓN		36
3.1	ESTRUCTURA DE LA SOLUCIÓN	36
3.2	STACK TECNOLÓGICO	36
3.3	CONJUNTO DE DATOS	39
3.4	MODELO DE APRENDIZAJE AUTOMÁTICO	41
CAPÍTULO 4: VALIDACIÓN DE LA SOLUCIÓN		43
4.1	EXPERIMENTOS	43
4.2	RESULTADOS	44
4.2.1	Resultados clasificación de aplicaciones	46
4.2.2	Resumen de resultados	60
CAPÍTULO 5: CONCLUSIONES		63
REFERENCIAS BIBLIOGRÁFICAS		65

ÍNDICE DE FIGURAS

1	El mapa mundial de Android e iOS.	3
2	NLP pipeline.	15
3	Matriz de confusión.	19
4	Ejemplo ilustrativo de agrupamiento de permisos.	25
5	Grupos de permisos riesgosos.	29
6	Permission granting based on protection level.	30
7	Results of Review to Permission Inference.	34
8	Esquema de la estructura de la solución.	37
9	Diagrama de flujo de la solución.	38
10	Extracto del dataset.	40

ÍNDICE DE TABLAS

1	Descripción seguridad de los datos WhatsApp Messenger en PlayStore. . . .	4
2	API Levels y sus versiones de Android.	27
3	Clasificación permisos Android.	28
4	Stack tecnológico.	39
5	Resultados del experimento con Pytorch Lightning.	45
6	Resultados del experimento con Tensorflow.	45
7	Hiperparámetros utilizados.	45
8	Resumen análisis aplicaciones.	60

CAPÍTULO 1

INTRODUCCIÓN

1.1. MOTIVACIÓN Y CONTEXTO

Actualmente, la mayoría de las personas alrededor del mundo, tienen al menos un *smartphone*, para el año 2022 hay más de 3000 millones a nivel mundial y se estima que siga creciendo en los próximos años¹, de todo el espectro de personas que existen en el mundo, muchos de ellos no se interesan por realizar una lectura acabo de la descripción de la App, sino que las descargan sin dudar sobre su contenido ni de la forma en la que estas aplicaciones acceden y comparten a su información personal, mediante el uso de permisos [Ngobeni y Mhlongo, 2019], lo que, sin lugar a dudas, representa una vulnerabilidad y un riesgo potencial a la privacidad de los usuarios. En una encuesta realizada a 308 usuarios de Android a través de internet y a 25 usuarios realizando un estudio en un laboratorio, se probó la comprensión, la atención y el comportamiento de los usuarios al momento de instalar una aplicación, y obtuvo como resultado que solo el 20 % de las personas que realizaron el estudio en el laboratorio, tuvo conciencia y comprensión sobre los permisos de las aplicaciones que debía instalar [Felt *et al.*, 2012]. Esta falta de conciencia afecta directamente a los mismos usuarios, pues con esta actitud, inconscientemente, están haciendo entrega de su información personal a las aplicaciones y a quienes éstas les pertenezcan.

Los teléfonos con el SO de Android son más vulnerables frente a *malware* debido a la libertad que le da al usuario a la hora de instalar aplicaciones², y también a que es el S.O. más masivo³ y el que llega a más personas en el mundo, de manera que, ante una eventual vulnerabilidad en el SO o en las aplicaciones que allí se ejecutan, éstos podrían tener un mayor beneficio explotando estas debilidades. Un ejemplo de los que podría ocurrir si no se toman en cuenta los riesgos de la desinformación sobre los permisos de las aplicaciones, es el riesgo de comprometer la integridad de la información personal o que es confidencial del usuario, por ejemplo, la información bancaria o de contraseñas de correos electrónicos e incluso vulnerar la seguridad informática de una empresa a través de los descuidos de los usuarios que almacenen información delicada del negocio, en sus celulares [Ngobeni y Mhlongo, 2019].

Otro ejemplo de los riesgos a los que pueden estar expuestos los usuarios de smartphones, es entregar información personal y/o confidencial sin el consentimiento pleno de que se está haciendo. Esto podría llegar a ocurrir al utilizar una aplicación que contiene anuncios donde, por medio de estos, se recolecta la información para ser utilizada en la industria tanto publicitaria como de *tracking*. Este es un tipo diferente de riesgo, es más una falta de ética

¹<https://es.statista.com/estadisticas/636569/usuarios-de-telefonos-inteligentes-a-nivel-mundial>

²<https://www.xatakandroid.com/tutoriales/tres-permisos-que-abren-puerta-al-malware-tu-android-como-evitarlo>

³<https://es.statista.com/grafico/29620/sistema-operativo-movil-con-la-mayor-cuota-de-mercado-por-pais/>

por parte de las empresas, “robar” la información de las personas o, en otras palabras, no notificar o advertir de manera clara a los usuarios de estas aplicaciones que se les recolectara la información para los propósitos de terceros [Quermann y Degeling, 2020].

Un caso donde se evidencia el gran problema que se genera al tener usuarios que no son bien informados respecto a lo que se hará con su información, es el problema que tuvo en el 2018 Facebook con Cambridge Analytica. Para poner contexto respecto a este caso, Facebook realizó un test de personalidad a sus usuarios recolectando datos personales, los que luego fueron vendidos a Cambridge Analytica. Esto se reveló debido a que existieron reclamos que acusaban que gracias a esta información, se vieron influenciados los resultados de las elecciones presidenciales de EE.UU ⁴.

El problema de filtrado de información, en este caso, ocurre porque se tenía constancia que solo 270.000 personas realizaron el test de personalidad, pero Cambridge Analytica tenía datos de 50 millones de usuarios y estos fueron recolectados sin el consentimiento explícito de sus propietarios al ser extraídos a través de redes de amigos. Para contextualizar lo importante que son los permisos dentro de los sistemas Android, se tiene que los permisos de una app, son las que permiten acceder o en otro caso, vulnerar la privacidad del usuario, ya que éstos protegen el acceso a lo siguiente ⁵:

- **Datos restringidos**, como el estado del sistema y la información de contacto del usuario.
- **Acciones restringidas**, como conectarte a un dispositivo vinculado y grabar audio.

Con esto en consideración, es importante controlar y tener conocimiento, como usuario, de los permisos que se aceptan al momento de instalar las *apps*, que se descargan desde las tiendas de aplicaciones, ya que podrían vulnerar nuestra privacidad e incluso poner en riesgo nuestra seguridad.

Esta explicación de los permisos y su importancia, cobra más sentido tomando en cuenta que los dispositivos que utilizan el sistema operativo Android componen cerca del 71 % de todos los dispositivos actualmente ⁶.

Para una mejor visualización de la distribución actual de los distintos sistemas que utilizan actualmente los smartphones se tiene el gráfico de la figura 1.

Observando este gráfico, se puede identificar la distribución mundial de los dos sistemas operativos más utilizados en smartphones. Android se mantiene como líder a nivel mundial, cubriendo una cuota de mercado del 71 %, mientras que iOS representa tan solo el 28 %. En los países donde destaca iOS, se encuentra Estados Unidos, donde abarca el 57 %, superando

⁴<https://www.bbc.com/mundo/noticias-49093124>

⁵<https://developer.android.com/guide/topics/permissions/overview>

⁶<https://es.statista.com/grafico/29620/sistema-operativo-movil-con-la-mayor-cuota-de-mercado-por-pais/>



Figura 1: El mapa mundial de Android e iOS.
Fuente: es.statista.cl.

al 43 % de Android. El sistema operativo de Apple también domina en Canadá y en países europeos como Noruega, Suecia y el Reino Unido, además de tener presencia destacada en Japón, Australia y algunos otros países.

Al tener en cuenta estos números, es posible tener una idea de la magnitud del riesgo que representaría para los usuarios de Android una potencial fuga de información. Esto resalta la relevancia de concientizar a las personas sobre la seguridad de su información y, por ende, la importancia y comprensión que se debe tener al momento de instalar una aplicación móvil.

1.2. DEFINICIÓN DEL PROBLEMA

En la actualidad, en la tienda de aplicaciones de Google Play Store, se entrega una descripción sobre la seguridad de los datos, es decir, lo que la aplicación hará y no hará con los datos de los usuarios. Un ejemplo de la forma en cómo se presenta esta información, se presenta en

la Tabla 1, tomando como ejemplo la descripción de la aplicación *WhatsApp Messenger*:

Tabla 1: Descripción seguridad de los datos WhatsApp Messenger en PlayStore.

Fuente: Elaboración Propia.

WhatsApp Messenger	
Sección seguridad de los datos	
Compartición de datos	No se comparten datos a terceros
Datos que se recogen	Ubicación Información personal Información financiera Contactos Actividad en la aplicación Información y rendimiento de aplicaciones IDs de dispositivo o de otro tipo
Prácticas de seguridad	Los datos se cifran en tránsito Puedes solicitar que se eliminen los datos

En la sección de compartición de datos se indica que no se comparten los datos a terceros, esto quiere decir que la aplicación no comparte los datos personales a otras empresas u organizaciones. Luego se mencionan todos los datos que recolecta la aplicación, tales como la ubicación que tiene como finalidad ser una funcionalidad de la aplicación, análisis y prevención de fraudes, también de la información personal que recolecta el correo electrónico, IDs del usuario y el número de teléfono, la finalidad de la recolección de esta información, al menos según informa la empresa, se debe a la necesidad de brindar un mejor servicio por medio del mejoramiento de funcionalidades de la aplicación, prevención de fraudes y seguridad. De la información financiera la aplicación obtiene la información de pago del usuario y el historial de compras, estos datos se piden con la finalidad de utilizarse para la funcionalidad de la aplicación, prevención de fraudes, seguridad y cumplimiento. Los contactos se piden con la finalidad de ser una funcionalidad de la aplicación y prevención de fraudes, seguridad y cumplimiento. De la actividad en la aplicación se recolecta las interacciones de la aplicación y otros contenidos generados por el usuario, esto con la finalidad de ser una funcionalidad de la aplicación, análisis y prevención de fraudes, seguridad y cumplimiento. De la Información y rendimiento de aplicaciones se recolectan los registros de fallos, diagnósticos y otros datos de rendimiento de aplicaciones, estos se extraen para ser una funcionalidad de la aplicación y análisis. Los IDs de dispositivo o de otro tipo se recolectan, para ser usados en una funcionalidad de la aplicación, análisis, comunicaciones del desarrollador, publicidad o marketing y prevención de fraudes, seguridad y cumplimiento ⁷.

Toda esta información parece muy detallada y a primera vista se ve que los usuarios comunes quedarían muy informados sobre los datos e información que se entrega a la aplicación, en este caso *WhatsApp Messenger*, sin embargo, esta útil información la debe entregar el desarrollador de la aplicación y esto podría suponer un problema, ya que no es un requi-

⁷<https://play.google.com/store/apps/datasafety?id=com.whatsapp>

sito restrictivo para publicar aplicaciones en esta plataforma. De lo anterior se desprende que si los desarrolladores, no mencionan todo lo que hacen con los datos de las personas, esta información no podría estar a la mano del usuario, ya que Google Play no se hace cargo respecto a lo que hacen las aplicaciones con la información que éstas recolectan, sólo se preocupan de que las Apps cumplan con las políticas que la tienda les impone. Entonces queda a criterio del desarrollador si quiere o no anunciar que datos recolecta, cómo los protege, que acciones realiza con ellos y si es que los comparte o no con terceros ⁸.

Además la importancia de la coherencia entre la descripción y los permisos de una aplicación en el contexto de aplicaciones móviles Android es un tema crucial que ha sido objeto de estudio en la investigación académica. Un ejemplo destacado de este enfoque se encuentra en el *paper* titulado “AC-Net: Assessing the Consistency of Description and Permission in Android Apps”. Este estudio destaca cómo la relación entre lo que se describe en la información de una aplicación y los permisos que solicita es fundamental para la comprensión de los usuarios y la transparencia en cuanto a las capacidades y acceso que la aplicación tendrá en un dispositivo móvil.

La falta de coherencia entre la descripción y los permisos puede llevar a problemas de transparencia y confianza por parte de los usuarios. Cuando la descripción de una aplicación no refleja completamente los permisos necesarios, los usuarios pueden sentirse desconcertados o preocupados acerca de las intenciones de la aplicación. La consecuencia es que los usuarios podrían abstenerse de instalar aplicaciones o limitar su uso debido a la incertidumbre con respecto a la seguridad y privacidad de sus datos.

El enfoque propuesto en el *paper* mencionado busca abordar este problema mediante el uso de técnicas de aprendizaje automático, específicamente redes neuronales, para evaluar la coherencia entre las descripciones y los permisos de las aplicaciones. Esta evaluación se basa en un análisis de datos de entrenamiento, lo que permite al modelo determinar si la descripción de una aplicación es coherente con los permisos que solicita. El objetivo es proporcionar a los usuarios una evaluación más precisa y transparente de las capacidades y el acceso de las aplicaciones que están considerando instalar.

1.3. PROPUESTA DE SOLUCIÓN

Como propuesta para abordar el problema planteado, se propone el desarrollo de una herramienta informática basada en una máquina de aprendizaje automático. Esta herramienta utilizará datos provenientes de las aplicaciones disponibles en la tienda de Google Play Store, como sus descripciones. Utilizando esta información como entrada, la herramienta buscará identificar los permisos que una aplicación podría necesitar para su funcionamiento en un dispositivo móvil, y detectar discrepancias entre las funcionalidades declaradas en las descripciones de las aplicaciones y los permisos que realmente solicitan (coherencia entre

⁸<https://support.google.com/googleplay/android-developer/answer/10787469?hl=es>

descripción y permisos). La máquina de aprendizaje actuará como un clasificador de texto, utilizando las descripciones de las aplicaciones como datos de entrada y asignando etiquetas o clases que representen los permisos que cada aplicación podría utilizar. Posteriormente, se compararán los permisos predichos por el clasificador con los permisos reales obtenidos del archivo *manifest* de las aplicaciones de Android, para detectar posibles incongruencias.

Además, se creará una API con el objetivo de permitir a los usuarios obtener los permisos de una aplicación a través de un endpoint específico. Los usuarios podrán ingresar el ID de la aplicación y la API devolverá la lista de permisos asociados a dicha aplicación.

1.4. OBJETIVOS

1.4.1. Objetivo general

Diseñar e implementar una herramienta que permita identificar los permisos mínimos que necesita una aplicación, en función de su contexto y su descripción en la plataforma Google Play Store. Esta herramienta realizará un análisis de las variables previamente mencionadas, por medio del procesamiento de lenguaje natural.

1.4.2. Objetivos específicos

1. Evaluar métodos y librerías de procesamiento de lenguaje natural que, según sus características, puedan ser adaptados al problema a resolver.
2. Realizar un análisis de las aplicaciones disponibles y sus respectivas descripciones en Google Play Store, para determinar las *keywords* que especifican el contexto y los permisos que necesitará la aplicación para funcionar.
3. Construir un software que genere un listado de permisos mínimos necesarios para el funcionamiento de una aplicación, según el análisis de su descripción y contexto.
4. Generar un reporte con la lista de permisos que utilizará cada aplicación, señalando las que podrían traer un riesgo para el usuario.
5. Redactar un documento con la información obtenida de este proceso de investigación, que explique el proceso llevado a cabo para obtener esta herramienta, la explicación de su composición, banco de pruebas, análisis de resultados y conclusiones obtenidas.

1.5. Estructura del documento

El documento se divide en capítulos donde cada uno abarca una sección específica y a su vez estos están divididos en subsecciones. Cada uno de estos capítulos son:

1. Introducción: este trata sobre la motivación de resolver el problema y la definición del mismo, además de una breve explicación de la solución propuesta.
2. Marco conceptual: en este capítulo se explica el estado del arte y el marco teórico del problema a resolver.
3. Propuesta de solución: en este capítulo se desarrolla la solución, se profundiza en la propuesta de solución, se explica el *setup* tecnológico que se utilizó en el desarrollo de la solución. Además se explica como funciona la solución.
4. Validación de la solución: en este capítulo se validan los resultados obtenidos con la solución propuesta.
5. Conclusiones: en este capítulo se muestran las conclusiones, si se consiguieron resultados notables respecto al trabajo ya realizado por otros autores y sobre el trabajo futuro tomando como base este desarrollo.

CAPÍTULO 2

MARCO CONCEPTUAL

2.1. MACHINE LEARNING

El aprendizaje automático ha evolucionado a partir del campo del reconocimiento de patrones y, en particular, de la inteligencia artificial, que es una rama de la informática. El aprendizaje automático está relacionado con la estadística computacional y se especializa en hacer predicciones. La investigación actual en aprendizaje automático se centra en el procesamiento del lenguaje natural, la visión por ordenador, el reconocimiento de patrones, la computación cognitiva y la representación del conocimiento. Los métodos de aprendizaje automático pueden denominarse modelos predictivos cuando se utilizan en un contexto industrial . [N y Gupta, 2020]

El aprendizaje automático, en esencia, significa “enseñanza a las máquinas”. En este contexto, las máquinas se refieren a dispositivos informáticos capaces de ejecutar algoritmos de aprendizaje automático. Implica crear un marco de aprendizaje y proporcionar datos bien formateados, relevantes y limpios para educar a la máquina para lograr un objetivo específico. Estas características de los datos se refieren al estado de los datos: deben estar bien formateados en el sentido de ser aceptados por el modelo, ya sea como vectores de números o binarios. Para que sean relevantes y limpios, deben ser útiles en el proceso de aprendizaje del modelo. Por ejemplo, si se está creando un clasificador de texto, el texto ingresado al modelo no debe contener caracteres especiales ni puntuación. El término “modelo” se refiere a una representación matemática que evoluciona a medida que la máquina aprende de sus datos de entrenamiento. Un modelo de aprendizaje automático puede generalizar y adaptarse a casos novedosos, lo que lo hace competente para manejar diversas situaciones. El objetivo final es crear un sistema en el que el modelo mejore continuamente su desempeño.

Las GPUs son utilizadas para el entrenamiento de los modelos de Machine Learning debido a que realizan cálculos técnicos más rápido y con mayor eficiencia energética que las CPU. Esto significa que ofrecen un rendimiento líder para el entrenamiento y la inferencia de IA, así como ganancias en una amplia gama de aplicaciones que utilizan la computación acelerada.

⁹

2.1.1. DEEP LEARNING

Deep Learning es parte de Machine Learning, este es, en palabras simples, una red neuronal con tres o más capas que intenta emular el comportamiento del cerebro humano, no consigue igualarlo pero igualmente consigue “aprender” a partir de grandes cantidades de datos.

⁹<https://la.blogs.nvidia.com/2024/01/30/por-que-las-gpu-son-excelentes-para-la-ia/>

Ya con una sola capa, una red neuronal es capaz de realizar predicciones aproximadas, las capas ocultas adicionales ayudan a optimizar y refinar la precisión.

El deep learning y el machine learning se distinguen por el tipo de datos con los que trabajan, por ejemplo el machine learning necesita como datos de entrada datos estructurados y etiquetados para realizar las predicciones, este hecho no implica que no pueda utilizar datos no estructurados, lo que si realiza es un procesamiento previo para darles una estructura. En cambio los algoritmos de deep learning se les puede ingresar datos no estructurados ya que elimina parte del procesamiento previo de los datos que requiere el machine learning, como ejemplo de datos estructurados son tablas con datos numéricos y sus etiquetas correspondientes, en cambio los no estructurados son datos como líneas de texto o imágenes.¹⁰

2.1.2. RNN

Una red neuronal recurrente (RNN) es un tipo de modelo en el campo del aprendizaje profundo que se entrena para procesar y transformar datos que están organizados en una secuencia de entrada en una secuencia de salida específica. Estos datos secuenciales pueden ser palabras, oraciones o cualquier tipo de información que tenga un orden específico en el tiempo, donde las partes individuales interactúan siguiendo reglas complejas tanto semánticas como sintácticas. La RNN es una especie de programa informático compuesto por varios elementos conectados entre sí, diseñados para emular la manera en que los humanos realizan tareas que involucran la manipulación de datos secuenciales, como traducir texto de un idioma a otro. Sin embargo, actualmente las RNN están siendo reemplazadas en gran medida por modelos de inteligencia artificial basados en transformadores y modelos de lenguaje de gran escala, los cuales son considerablemente más efectivos en el procesamiento de datos organizados de manera secuencial.

Además existen más redes neuronales derivadas de RNN como GRU (unidad recurrente cerrada) esta es una red neuronal recurrente que permite la retención selectiva de la memoria. El modelo agrega una actualización y olvida la puerta a su capa oculta, que puede almacenar o eliminar información de la memoria.¹¹

2.1.3. ENTRENAMIENTO DE MODELOS

Durante el proceso de entrenamiento de las máquinas de aprendizaje, es fundamental que puedan generalizar conceptos. Por ejemplo, cuando una persona ve un perro labrador por primera vez, es probable que reconozca sus características como propias de un perro. Sin embargo, si esta misma persona luego observa 10 fotos de perros de distintas razas, aún así será capaz de reconocer las características comunes de los perros, como tener cuatro

¹⁰<https://www.ibm.com/es-es/topics/deep-learning>

¹¹<https://aws.amazon.com/es/what-is/recurrent-neural-network/>

patas y ser peludos, lo que le permite inferir que se trata de un perro, y además distinguirlos de otros animales con características similares. Este proceso de generalización es esencial para que las máquinas puedan realizar una correcta discriminación de los datos durante el entrenamiento, para así cuando a la máquina se le presente un dato nuevo desconocido sea capaz de sintetizarlo, comprenderlo y devolvernos un resultado fiable dada su capacidad de generalización.

Las máquinas de aprendizaje enfrentan un desafío con la generalización, ya que si la cantidad de datos de entrenamiento es muy reducida, pueden caer en el fenómeno conocido como *underfitting*. Siguiendo con el ejemplo de los perros, esto ocurriría si solo se presenta una imagen de un labrador durante el entrenamiento y luego se espera que la máquina discrimine correctamente entre varias imágenes de diferentes razas de perros. En este caso, la máquina sería incapaz de producir resultados fiables debido a la falta de datos de entrenamiento. Además, el *underfitting* también se manifiesta cuando la máquina generaliza en exceso, considerando que todo lo que ve es un perro, incluso si se le muestra una imagen de un gato o incluso de un árbol.

Sin embargo, si a la máquina se le enseña 10 fotos de perros de un solo color durante el entrenamiento y luego se enfrenta a una imagen de un perro de un color diferente, es posible que tampoco la clasifique como un perro. Esto se debe a que la máquina ha aprendido de manera excesiva las características específicas de los perros en las fotos de entrenamiento, sin poder generalizar a variaciones en sus características como el color del pelaje. Este fenómeno se conoce como *overfitting*.

El fenómeno más común es el *overfitting* ya que la máquina se ajusta a aprender los casos particulares que se les enseñe y no será capaz de reconocer nuevos datos de entrada.

Para abordar el problema del *overfitting* y el *underfitting*, existen diversas técnicas que pueden aplicarse. Una estrategia para combatir el *overfitting* es la implementación del *Early Stopping*, que implica monitorear el rendimiento del modelo en un conjunto de datos de validación durante el entrenamiento y detener el proceso de entrenamiento una vez que el rendimiento en este conjunto comienza a empeorar, lo que indica que el modelo está cayendo en *overfitting* con los datos de entrenamiento.

Para mitigar el *underfitting*, una técnica útil es el uso de *subsampling* o *oversampling*. Con el *subsampling*, se puede aumentar la cantidad de datos de entrenamiento al generar nuevas muestras a partir de los datos existentes, lo que puede ayudar al modelo a aprender mejor las características de los datos. Por otro lado, con el *oversampling*, se pueden equilibrar las clases desproporcionadas aumentando la frecuencia de las muestras de las clases minoritarias, lo que puede mejorar la capacidad del modelo para generalizar correctamente.¹²

¹²<https://www.aprendemachinelearning.com/que-es-overfitting-y-underfitting-y-como-solucionarlo/>

2.2. PROCESAMIENTO DE LENGUAJE NATURAL

2.2.1. VISIÓN GENERAL

El estado actual del Procesamiento del Lenguaje Natural (NLP) se caracteriza por su creciente importancia en la tecnología actual. NLP se ha convertido en un avance indispensable en los últimos años y tiene aplicaciones en una amplia gama de áreas, incluyendo la respuesta a preguntas, síntesis y reconocimiento de voz, recuperación de información, extracción de información y traducción automática. El NLP se enfoca en el análisis y comprensión de lenguaje humano utilizando diversas tecnologías y enriquece la comunicación entre humanos y máquinas, así como la comunicación entre seres humanos a través del procesamiento de texto y expresiones habladas.

Dentro del campo de NLP, el etiquetado de partes del discurso (*Part-Of-Speech tagging*) es una tarea esencial que consiste en asignar etiquetas sintácticas a cada palabra en un texto según el contexto en el que aparecen. Esta tarea es fundamental en aplicaciones de NLP como la traducción automática y la desambiguación de significados de palabras. La idea detrás del etiquetado de partes del discurso surge de la necesidad de interpretar las palabras correctamente, ya que una misma palabra puede tener diferentes significados según el contexto, por ejemplo, la palabra río que dependiendo del contexto se refiere a una corriente de agua o a una conjugación del verbo reír.

Los investigadores han explorado cómo combinar el Aprendizaje Profundo (*Deep Learning*, *DL*) y el Aprendizaje Automático (*Machine Learning*, *ML*) para crear etiquetadores de partes del discurso más sofisticados. Estas técnicas son populares en NLP, y la proliferación de potentes unidades de procesamiento gráfico (*GPUs*) ha contribuido a su crecimiento. El enfoque de ML se basa en la ingeniería de características para aprender de *corpus* de datos, mientras que los etiquetadores basados en DL son superiores en el aprendizaje de características complejas directamente de los datos sin necesidad de una ingeniería de características previa.

En cuanto a las aplicaciones de NLP, se destacan dos áreas: la Traducción Automática y la Categorización de Texto. La Traducción Automática es esencial para superar barreras lingüísticas en un mundo donde coexisten muchos idiomas. Se emplean técnicas de Aprendizaje Automático para calcular la probabilidad de que una palabra o frase en un idioma se relacione con una palabra o frase en otro idioma. La precisión de la Traducción Automática se evalúa mediante métricas como el *BLEU score*.

La Categorización de Texto se aplica en la clasificación de documentos y datos según índices predefinidos. Esto tiene aplicaciones en la automatización de la clasificación de informes, como noticias, informes de mercado y correos electrónicos. La detección de spam en correos electrónicos es un ejemplo de uso de la categorización de texto, donde los filtros de spam utilizan protocolos y reglas para distinguir correos no deseados de comunicaciones legítimas. [Kaushal y Chadha, 2022]

El BLEU score es significativo en tareas de traducción automática debido a su simplicidad y eficacia para evaluar la calidad de las traducciones generadas por máquinas en comparación con las traducciones de referencia. Sin embargo, tiene limitaciones, ya que se basa en gran medida en n-gramas y puede no capturar con precisión el significado general o la fluidez del texto traducido. También puede penalizar traducciones más largas que las de referencia, lo que puede ser injusto en algunos casos.¹³

2.2.2. CURVAS NLP

La investigación de NLP se puede dividir en 3 curvas, la curva sintáctica, la curva semántica y la curva pragmática, donde cada una se diferencia en el enfoque al realizar el análisis del texto. Estas curvas además representan los avances y cambios significativos en la investigación y desarrollo del procesamiento del lenguaje natural a lo largo del tiempo, es decir, representan las diferentes etapas o momentos clave en los que se producen avances significativos en el campo del NLP.

En las primeras etapas de la investigación del procesamiento de lenguaje natural, estaba centrado en la sintaxis, es decir, la curva sintáctica, la cual es utilizada para el análisis gramatical y la estructura de las oraciones. Sin embargo, con el crecimiento exponencial de la información en la era de la web y las redes sociales, la necesidad de comprender el significado detrás de las palabras se tornó imprescindible. Esta curva está agrupada en 3 categorías: detección de palabras claves, afinidad léxica y métodos estadísticos [Cambria y White, 2014]

1. La detección de palabras claves, es el acercamiento más ingenuo y el más popular debido a su accesibilidad y su economía. El texto es clasificado en categorías basado en la presencia de palabras poco ambiguas.
2. La afinidad léxica es más sofisticada que la detección de palabras claves, ya que es más que solo detectar palabras obvias, esta técnica asigna arbitrariamente una probabilidad de afinidad a ciertas palabras para una categoría en particular.
3. Los métodos estadísticos de NLP son la corriente común desde los 90s en su investigación. Este utiliza modelos lingüísticos basados en algoritmos populares de *machine learning*.

La semántica, o el estudio del significado del lenguaje, emergió como un enfoque crucial en NLP. La semántica permitió a las máquinas no solo reconocer palabras, sino también comprender conceptos y relaciones entre ellos. Éste, en lugar de analizar los textos, se enfoca en las palabras claves y las ocurrencias que estas tienen a lo largo del mismo. La técnica basada en semántica utiliza los conceptos de los textos que no están explícitamente escritos en

¹³<https://medium.com/@sthanikamsanthosh1994/understanding-bleu-and-rouge-score-for-nlp-evaluation-1ab334ecadcb>

ellos, lo que otorga al análisis información relevante, útil para el procesamiento de lenguaje natural.

Esta curva semántica de NLP se agrupa en dos categorías principales,

1. NLP taxonómico.
2. NLP endógeno.

El NLP taxonómico tiene iniciativas que apuntan a construir, como su nombre lo dice, taxonomías para encontrar la semántica subsuntiva o jerárquica asociada a expresiones de lenguaje natural. Estas taxonomías consisten en conceptos, instancias, atributos y valores, además de relaciones. Estas son utilizadas para crear representaciones basadas en relaciones que son extraídas mediante patrones sintácticos. En particular, las representaciones de conocimiento jerárquico se basan en relaciones “Es un/a” (IsA), que generalmente se extraen mediante patrones sintácticos para descubrir hiperónimos automáticamente. Por ejemplo, a partir de fragmentos de texto como “artistas como Pablo Picasso.” “Pablo Picasso y otros artistas”, se pueden inferir relaciones como <Pablo Picasso-Es un/a-artista>. Esto permite establecer que Pablo Picasso es una instancia de la categoría general “artista”. La construcción de tales relaciones jerárquicas es fundamental para comprender y organizar la información en el procesamiento del lenguaje natural y la representación del conocimiento.

El NLP endógeno involucra el uso de técnicas de *machine learning* para generar un análisis del cuerpo del texto, construyendo estructuras que se asemejan a los conceptos de un conjunto de documentos. Esto no significa que se debe conocer semánticamente los documentos a utilizar, más bien solo se basa en el conocimiento endógeno de estos. Algunas de las ventajas de este enfoque sobre conocimientos de ingeniería son la eficiencia y la no necesidad de tener mano experta, además de su portabilidad a distintos dominios.

La curva pragmática se enfoca en entender la narrativa y se refiere al uso del lenguaje en contextos reales y a la interpretación de mensajes basada en la situación y el conocimiento compartido. Este enfoque tiene como objetivo lograr una comprensión más profunda del lenguaje humano y sus matices, lo que permitiría a las máquinas no solo procesar el significado, sino también interactuar de manera más natural con los seres humanos. Este corresponde al siguiente paso, luego de la curva semántica, ya que una vez el NLP pueda comprender la semántica al nivel de procesamiento humano de textos, en ese momento será necesario dar un salto. Ya existen estudios que investigan esta interpretación de la narrativa, la más representativa es el trabajo de Patrick Winston sobre los modelos computacionales de narrativa que está basado en 5 hipótesis [Cambria y White, 2014],

- La hipótesis del lenguaje interno: todos tenemos un lenguaje simbólico interno que permite describir acontecimientos.

- La hipótesis de la historia consistente: podemos ensamblar partes de descripciones de eventos dentro de una historia.
- La hipótesis de la percepción dirigida: podemos dirigir nuestra facultad de percepción para responder preguntas utilizando situaciones reales e imaginarias.
- La hipótesis del animal social: tenemos una necesidad de expresar nuestro pensamiento de lenguaje interno en un lenguaje externo.
- La hipótesis de ingeniería exótica: nuestros cerebros no están diseñados como un sistema de ingeniería estándar de izquierda a derecha.

La noción de “saltar curvas” en NLP implica que la investigación se está moviendo más allá de los enfoques basados en sintaxis hacia la comprensión semántica y pragmática. Este avance promete revolucionar la capacidad de las máquinas para comprender el lenguaje humano de una manera más rica y natural. Además, este enfoque puede tener un impacto significativo en aplicaciones prácticas, como la traducción automática, la generación de texto y la interacción humano-máquina.

Las tres curvas antes descritas, son los pasos que ha tomado y tomará el procesamiento de lenguaje natural, el cual evolucionó de solo analizar el texto, considerando únicamente lo que está escrito explícitamente, a después analizar exhaustivamente, tomando en consideración el contexto y los conceptos que están detrás de las palabras obteniendo aún más información para interpretar lo que se escribe. Ya para la última etapa que está siendo investigada recientemente, se analiza el cómo se creó el texto para así entender el enfoque y profundizar más en el entendimiento del cerebro humano por parte de las inteligencias artificiales. [Cambria y White, 2014]

2.2.3. ETAPAS DEL PRE-PROCESAMIENTO PARA NLP

Antes de realizar un proceso de NLP, el texto debe pasar por un tratamiento previo de depuración. La Figura 2 presenta un diagrama del preprocesamiento que se debe realizar a un texto para posteriormente ser analizado y entendido por la máquina y los modelos de procesamiento de lenguaje natural.

Lo primero que se tiene al momento de iniciar el pre procesamiento de los datos es el texto en bruto sin ningún tratamiento para poder llegar a ser entendido por el computador. El primer paso es la segmentación de frases o *Sentence Segmentation*, esta consiste en dividir los párrafos en frases, ya que de este modo se reduce la complejidad y simplifica el proceso para conseguir mejores resultados. Este paso se realiza porque los computadores no interpretan el lenguaje como lo realizan los humanos y se puede facilitar este entendimiento dividiendo el texto en frases.

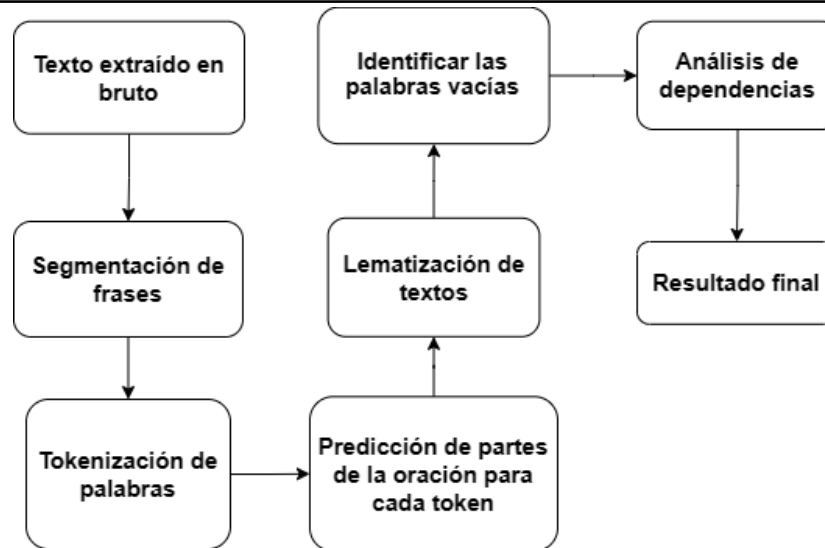


Figura 2: NLP pipeline.
Fuente: Elaboracion propia.

El siguiente paso es la *tokenización* de palabras o *Word Tokenization*, este consiste en el proceso de dividir una frase, oración, párrafo o documentos enteros en la unidad mínima, es decir, en palabras y términos. Estas pequeñas unidades son llamadas *tokens*. Estas pueden ser tanto palabras como números o puntuaciones. Este proceso es esencial para el preprocesamiento, pues el significado de una palabra se puede interpretar fácilmente analizando el resto que está presente en el texto, ya sea contando su cantidad, apariciones en una frase o su frecuencia dentro del texto completo. Para algunos modelos de aprendizaje automático como BERT existen tokens que deben tener los datos de entrada de estos modelos para ser interpretados, el tokenizador de BERT agrega los siguientes tokens,

- [CLS]: Este token se utiliza como el primer token de cada secuencia de entrada en BERT.
- [SEP]: Este token se utiliza para separar dos secuencias de entrada en BERT.
- [PAD]: Este token se utiliza para rellenar secuencias de entrada a la misma longitud en BERT.
- [UNK]: Este token se utiliza para representar palabras desconocidas o fuera del vocabulario durante el preprocesamiento de datos en BERT.

El tercer paso es la predicción de partes de la oración para cada *token* o *Parts of Speech Prediction For Each Token*, en este procedimiento se considera cada *token* para analizar y predecir si estos *tokens* pertenecen a un sustantivo, pronombre, verbo, adjetivo y así sucesivamente con todos.

El cuarto paso es la lematización del texto o *Text Lemmatization*, en este procesamiento las palabras o *tokens* son llevadas a su forma en infinitivo para mejorar la comprensión por parte de la máquina, esto debido a que el computador entiende que las mismas palabras se utilizan para las mismas situaciones y contextos.

El quinto paso es identificar las palabras vacías o *Identifying Stop Words*, en esta etapa se identifican todas las palabras. Esto se realiza, ya que en el Inglés existen muchas palabras que son de relleno y no son útiles para entender el contexto, no agregan valor al entendimiento de las frases y le quita significado a este.

Por último, el siguiente procedimiento finaliza con el análisis de dependencias o *Dependency Parsing*, este análisis se divide en 3 categorías, estas son el etiquetado de *Part-Of-Speech*, el análisis sintáctico de dependencias y la redacción de la circunscripción. El análisis de dependencia analiza la estructura gramatical de la frase. Además se analiza la relación gramatical entre las palabras de la oración mediante la identificación de las dependencias sintácticas entre ellas. Esto implica determinar cómo cada palabra se relaciona con las demás en términos de roles gramaticales, como sujeto, objeto, complemento, etc. Este análisis ayuda a comprender la estructura sintáctica y jerárquica de la oración. La redacción de circunscripción se refiere a la representación visual o estructural de las dependencias sintácticas identificadas en la oración. La redacción de la circunscripción suele presentarse en forma de árbol de dependencias, donde cada palabra se representa como un nodo y las relaciones entre ellas se muestran mediante enlaces direccionales que indican la dirección de la dependencia.

2.2.4. MACHINE LEARNING EN NLP

El aprendizaje automático para el procesamiento de lenguaje natural y la analítica de texto es un campo crítico que implica el uso de algoritmos de aprendizaje automático y la inteligencia artificial “estrecha” para comprender el significado de documentos de texto. Estos documentos pueden abarcar una amplia gama de fuentes de texto, como comentarios en redes sociales, reseñas en línea, respuestas a encuestas, documentos financieros, médicos, legales y regulatorios. El objetivo central de utilizar el aprendizaje automático y la IA en NLP y la analítica de texto es mejorar, acelerar y automatizar los procesos esenciales de análisis de texto y las características de NLP que transforman el texto no estructurado en datos e información valiosa. Un ejemplo de estos textos no estructurados son los comentarios en redes sociales; al no contener una estructura definida, no pueden ser utilizados para NLP.

El aprendizaje automático para NLP y la analítica de texto emplean un conjunto de técnicas estadísticas para identificar partes del discurso, entidades, sentimientos y otros aspectos del texto. Estas técnicas pueden representarse como modelos que luego se aplican para analizar texto sin etiquetar (aprendizaje automático supervisado) o como algoritmos que funcionan en conjuntos extensos de datos para extraer significado (aprendizaje automático no supervisado).

El aprendizaje automático supervisado para NLP implica etiquetar o anotar documentos de texto con ejemplos de lo que la máquina debe buscar y cómo debe interpretar ese aspecto. Estos documentos etiquetados se utilizan para entrenar un modelo estadístico, que luego se utiliza para analizar texto no etiquetado. Esto permite que la máquina aprenda y entienda aspectos específicos del texto, como el análisis de sentimiento, el etiquetado de partes del discurso, el reconocimiento de entidades con nombre y la categorización. Algunos algoritmos comunes de aprendizaje automático NLP supervisado incluyen Máquinas de Soporte Vectorial, Redes Bayesianas, Campo Aleatorio Condicional y Redes Neuronales/Aprendizaje Profundo.

Las máquinas de soporte vectorial son un algoritmo de aprendizaje automático supervisado que puede utilizarse para problemas de clasificación o regresión. Sin embargo, su uso más habitual es la clasificación. Dadas 2 o más clases de datos etiquetados, actúa como un clasificador discriminativo, definido formalmente por un hiperplano óptimo que separa todas las clases. Los nuevos ejemplos, que se mapean en el mismo espacio, pueden clasificarse según el lado de la brecha en el que se encuentren.¹⁴

El nodo de redes bayesianas permite construir un modelo de probabilidad combinando pruebas observadas y registradas con conocimientos del mundo real “de sentido común” para determinar la probabilidad de instancias utilizando atributos aparentemente no relacionados. El nodo se centra en las redes Naïve Bayes aumentadas por árbol (TAN) y Markov Mantle Naïve Bayes, que se utilizan principalmente para la clasificación.¹⁵

Los modelos de Campo Aleatorio Condicional se basan en que dada una secuencia de datos este modelo asigna una etiqueta para cada elemento. Aunque presenta similitudes con los modelos ocultos de Márkov, estos son modelos generativos que modelan conjuntamente la distribución de probabilidad de las etiquetas (o estados) y las observaciones, mientras que los campos aleatorios condicionales modelan la probabilidad de la secuencia correcta de etiquetas condicionada por las observaciones, es decir, son modelos discriminativos.¹⁶

Los modelos de Aprendizaje profundo son un subconjunto de machine learning, que es básicamente una red neuronal con más de 3 capas. Estas redes neuronales intentan emular el comportamiento del cerebro humano, aunque están lejos de igualar su capacidad, igualmente les permiten “aprender” a partir de grandes cantidades de datos.¹⁷

Por otro lado, el aprendizaje automático no supervisado implica entrenar un modelo sin etiquetas o anotaciones previas. El Índice Semántico Latente (LSI) y la Factorización de Matrices son ejemplos de técnicas de aprendizaje no supervisado utilizadas en NLP. Estas técnicas ayudan a identificar patrones y relaciones dentro de conjuntos de datos de texto grandes y pueden ser valiosas para tareas como la categorización, la extracción de temas y la búsqueda

¹⁴<https://medium.com/@csarchiquerodriguez/maquina-de-soporte-vectorial-svm-92e9f1b1b1ac>

¹⁵<https://www.ibm.com/docs/es/spss-modeler/saas?topic=models-bayesian-network-node>

¹⁶https://www.wikiwand.com/es/Campo_aleatorio_condicional

¹⁷<https://www.ibm.com/es-es/topics/deep-learning#:~:text=el%20siguiente%20paso-,%C2%BFQu%C3%A9%20es%20Deep%20Learning%3F,de%20grandes%20cantidades%20de%20datos.>

de relaciones semánticas.¹⁸

El Índice Semántico Latente identifica palabras y frases que se repiten con frecuencia. Los científicos de datos utilizan LSI para búsquedas facetadas, o para devolver resultados de búsqueda que no son el término de búsqueda exacto.

La factorización de matrices utiliza “factores latentes” para descomponer una matriz grande en la combinación de dos matrices más pequeñas. Los factores latentes son similitudes entre los ítems, por ejemplo en la frase: “Tiré la pelota por encima de la montaña”. Es más probable que la palabra “lancé” se asocie con “pelota” que con “montaña”.

2.2.5. MÉTRICAS DE EVALUACIÓN

Para medir la calidad de los modelos de aprendizaje automático, se utilizan métricas que su objetivo es estimar la precisión de la generalización de un modelo sobre los datos futuros.¹⁹

Una matriz de confusión es una representación matricial de los resultados de las predicciones de cualquier prueba binaria que se utiliza a menudo para describir el rendimiento del modelo clasificador sobre un conjunto de datos de prueba cuyos valores reales se conocen. La matriz de confusión es relativamente sencilla de comprender, pero la terminología relacionada puede ser confusa. En la figura 3 se muestra una representación de la matriz de confusión de un problema de clasificación:

- Verdaderos positivos (TP): Datos que fueron clasificados por el modelo correctamente, verificados con los valores reales.
- Falso positivo (FP): Datos que fueron clasificados erróneamente como positivos, siendo que el valor real es negativo.
- Falso negativo (FN): Datos que fueron clasificados como negativos pero realmente son positivos.
- Verdadero negativo (TN): Datos que fueron clasificados correctamente como negativos.

El siguiente es un listado de las métricas que se utilizan comúnmente en clasificadores:¹⁹

- Exactitud (Accuracy): La exactitud mide cuántas predicciones son correctas en relación con el número total de predicciones. Es una métrica útil para problemas de clasifica-

¹⁸<https://www.lexalytics.com/blog/machine-learning-natural-language-processing/>

¹⁹<https://www.datasource.ai/es/data-science-articles/metricas-de-evaluacion-de-modelos-en-el-aprendizaje-automatico>

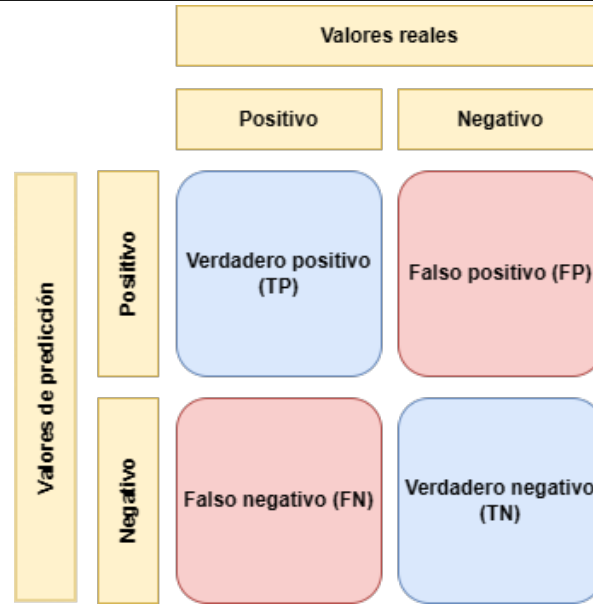


Figura 3: Matriz de confusión.
Fuente: Elaboración propia.

ción equilibrados, pero puede no ser adecuada cuando las clases están desequilibradas.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

- **Precisión (Precision):** La precisión mide cuántas predicciones positivas son realmente positivas. Se calcula dividiendo el número de verdaderos positivos por el número total de predicciones positivas. Es importante para evaluar cuán precisas son las predicciones positivas.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

- **Exhaustividad (recall):** La exhaustividad mide cuántas instancias positivas se identifican correctamente. Se calcula dividiendo el número de verdaderos positivos por el número total de instancias positivas. Es relevante para evaluar si el modelo captura la totalidad de las instancias positivas.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

- **Puntuación F1 (F1 score):** La puntuación F1 es una medida que combina precisión y exhaustividad en una sola métrica. Es especialmente útil cuando las clases están desequilibradas, ya que proporciona un equilibrio entre ambas. Un puntaje de F1 alto es sobre 0.7 y bajo esto el rendimiento del modelo no sería suficiente para considerarlo confiable.

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \quad (4)$$

Las métricas utilizadas en problemas de clasificación binaria pueden adaptarse para abordar problemas de clasificación multi-clase mediante técnicas de agregación. Hay dos enfoques principales para esta agregación: el promedio macro y el promedio micro.

El promedio macro calcula las métricas para cada clase individualmente y luego toma el promedio de estos valores. Este enfoque trata todas las clases de manera igual, sin importar su frecuencia o importancia relativa.

Por otro lado, el promedio micro construye una única matriz de confusión para todas las clases y calcula las métricas a partir de esta matriz combinada. Aquí, cada predicción y cada etiqueta tienen el mismo peso en el cálculo de las métricas, lo que puede ser útil cuando las clases tienen diferentes tamaños o importancias relativas.

2.3. HIPERPARÁMETROS

Los hiperparámetros son variables de configuración externa utilizadas para gestionar el entrenamiento de modelos de *machine learning*. Estos se diferencian de los parámetros, que son elementos internos derivados automáticamente durante el proceso de aprendizaje y no se configuran manualmente. Además determinan características clave como la arquitectura del modelo, la tasa de aprendizaje y la complejidad del modelo.

Los hiperparámetros influyen directamente en la estructura, funciones y rendimiento de los modelos. El ajuste de estos permite modificar el rendimiento del modelo para lograr mejores resultados. Este proceso es una parte fundamental del *machine learning*, y la elección de los valores de hiperparámetros apropiados es esencial para el éxito del entrenamiento.

Por ejemplo, si se considera la tasa de aprendizaje como un hiperparámetro, si el valor es demasiado alto, es posible que el modelo converja demasiado rápido con resultados subóptimos. Por el contrario, si el valor es muy bajo, el entrenamiento puede demorarse demasiado y los resultados pueden no converger.²⁰

Estos son algunos ejemplos de hiperparámetros comúnmente utilizados en el entrenamiento de modelos de machine learning:

- La tasa de aprendizaje es la frecuencia con la que un algoritmo actualiza las estimaciones. El decaimiento de la tasa de aprendizaje es una reducción gradual de dicha tasa a medida que avanza el tiempo para acelerar el aprendizaje.
- Los nodos de la red neuronal se refieren al número de nodos en cada capa oculta.
- Las capas de la red neuronal hacen referencia al número de capas ocultas en una red neuronal.

²⁰<https://aws.amazon.com/es/what-is/hyperparameter-tuning/>

- El tamaño del lote o *Batch size* es el tamaño del lote de datos de entrenamiento.
- Las capas de *dropout* que se agregan a la estructura del modelo para evitar el *overfitting* eliminando valores aleatorios de los vectores en cada *epoch*, así evitando que el modelo se concentre en una parte del conjunto de datos de entrenamiento.
- La cantidad de *epochs* o épocas son la cantidad de ciclos que un modelo entrena con cada ejemplo de entrada.

2.4. BERT

BERT, acrónimo de “Bidirectional Encoder Representations from Transformers”, es un modelo de lenguaje desarrollado por Google que ha demostrado ser un punto de referencia en el procesamiento del lenguaje natural. La innovación clave detrás de BERT radica en su capacidad para capturar el contexto y el significado de las palabras en el lenguaje de manera excepcional. Antes de BERT, los modelos de lenguaje pre-entrenados solían entrenarse de manera unidireccional, es decir, sólo consideraban el contexto a la izquierda o a la derecha de una palabra. BERT rompe esta limitación al aprender de manera bidireccional, comprendiendo todo el contexto de la oración.

El proceso de pre-entrenamiento de BERT implica exponer el modelo a enormes cantidades de texto en línea. Durante este proceso, BERT aprende a predecir palabras ocultas en una oración utilizando el contexto completo de la oración. A medida que BERT se vuelve experto en esta tarea, se convierte en un modelo de lenguaje altamente sofisticado que “comprende” el lenguaje en un nivel profundo. Este conocimiento pre-entrenado se almacena en los pesos del modelo, que luego pueden ajustarse para tareas de procesamiento del lenguaje natural específicas.

La capacidad de adaptación de BERT a tareas específicas es lo que lo hace tan valioso. Los investigadores y desarrolladores pueden utilizar BERT como punto de partida y ajustar sus pesos para resolver problemas de NLP, como la clasificación de texto, la generación de lenguaje natural, la traducción automática y la extracción de información. Esta versatilidad ha llevado a la proliferación de modelos BERT ajustados y adaptados a diversas aplicaciones de NLP.

Los embeddings en BERT son representaciones vectoriales de palabras que capturan la semántica contextual de cada palabra en una oración. Estos embeddings son fundamentales en el proceso de pre-entrenamiento de BERT y son utilizados como entrada para el modelo durante la etapa de fine-tuning en tareas específicas de procesamiento de lenguaje natural.

En términos de rendimiento, BERT ha superado con creces a modelos de lenguaje pre-entrenados anteriores y ha establecido un estándar muy alto en tareas de procesamiento del lenguaje natural, como la comprensión de preguntas, la traducción automática y la clasificación de texto. Su arquitectura basada en transformers y su enfoque bidireccional lo han

convertido en un modelo fundamental en la investigación y desarrollo de NLP.

Los modelos a los que supera BERT son OpenAI Sota, ELMo y OpenAI GPT, basándose en GLUE, The General Language Understanding Evaluation, que es una colección de diversas tareas de comprensión del lenguaje natural. Según este test, BERT obtuvo un puntaje de 82,1, siendo superior a OpenAI Sota, ELMo y OpenAI GPT, con puntajes de 74,0, 71,0 y 75,1 respectivamente [Jacob *et al.*, 2018].

2.4.1. FINE-TUNNING

Fine-tuning es una técnica de aprendizaje automático en la que un modelo pre entrenado se adapta a una tarea específica ajustando sus parámetros. Esto se logra al entrenar el modelo con un conjunto de datos más pequeño y específico para la tarea en cuestión. La técnica de *fine-tuning* se utiliza comúnmente en el entrenamiento de modelos de procesamiento del lenguaje natural (NLP), como clasificación de texto, respuesta a preguntas y traducción automática. Esto se debe a que los modelos pre entrenados de NLP, como BERT y RoBERTa, han sido entrenados en grandes conjuntos de datos que incluyen texto y código, permitiéndoles desarrollar una comprensión general del lenguaje útil para diversas tareas de NLP.

Para llevar a cabo el *fine-tuning* en un modelo pre entrenado, es necesario cargar el modelo y definir una nueva capa de clasificación. Esta capa de clasificación es una red neuronal simple que predice la etiqueta de clase para cada entrada de texto. Posteriormente, se entrena el modelo con un conjunto de datos más pequeño que contenga texto y etiquetas específicas para la tarea.

Una vez que el modelo ha sido entrenado, se puede utilizar para predecir la etiqueta de clase de nuevas entradas de texto. El *fine-tuning* resulta ser una manera efectiva de entrenar modelos de aprendizaje automático para tareas de NLP, ya que permite aprovechar el conocimiento previamente adquirido por un modelo pre entrenado.

Los beneficios de este proceso de entrenamiento incluyen su rapidez y simplicidad en comparación con el entrenamiento desde cero, así como la mejora del rendimiento, especialmente en conjuntos de datos pequeños. Además, permite adaptar el modelo pre entrenado a nuevas tareas sin perder el conocimiento previo. Sin embargo, presenta desafíos, como la elección adecuada de hiperparámetros durante el entrenamiento, la dificultad para evaluar el rendimiento con conjuntos de datos pequeños y la interpretación complicada de los resultados finales del modelo después del entrenamiento ²¹.

²¹<https://medium.com/@danushidk507/fine-tune-bert-for-text-classification-with-tensorflow-7175c4ab5475>

2.4.2. TOKENIZADOR DE BERT

Para poder utilizar BERT los datos de entrada deben ser *tokenizados* de una forma específica para que el modelo pre entrenado lo comprenda. Los *embeddings* para BERT son entrenados para dos tareas específicas, de clasificación de clases o etiquetas y para predicción de la siguiente sentencia, esta última consiste en averiguar si una oración es posterior a otra. Para las tareas de clasificación, un vector representa toda la oración de entrada necesaria para la tarea, pero en BERT se toma la decisión de que el estado oculto del primer token represente toda la frase, para lograrlo se adiciona manualmente el token [CLS] con este propósito.

Para las tareas de predicción de la siguiente oración, se necesita un método que informe al modelo donde termina la primera frase, y donde comienza la siguiente, para esto el token [SEP] se adiciona artificialmente. Además para las clasificaciones se agrega este mismo token al final de cada entrada de texto para demarcar su fin.

Al momento de utilizar BERT se le entrega una largo de las frases determinado, si las oraciones de entrada son más cortas que este parámetro se agrega relleno o tokens vacíos hasta completar el largo determinado. Para la implementación original el token [PAD] es usado para representar el relleno dentro de una frase.

Cuando el modelo original de BERT fue entrenado a cada token se le asigna una ID único, por lo tanto cuando se utiliza como modelo pre entrenado lo primero que se debe hacer es transformar las frases de entrada en sus IDs únicos correspondientes.

Un punto importante a tener en cuenta es que al utilizar un modelo pre entrenado como BERT, pueden existir tokens que no existan dentro del vocabulario del modelo, ya que este fue determinado dentro de su entrenamiento original, por lo que estos tokens son reemplazados por el token [UNK], conocido como el token desconocido o *unknown token*. Para evitar que una gran parte de los datos de entrada sean reemplazados por [UNK], BERT utiliza el algoritmo *WordPiece* que divide las palabras, y estas subpalabras pueden estar representadas dentro del vocabulario original de BERT ²².

2.4.3. RoBERTa

Existen ciertas características que diferencian a BERT de RoBERTa, una de ellas es la máscara dinámica, de la arquitectura de BERT recordamos que durante el pre entrenamiento, este realiza el modelado lingüístico intentando predecir un determinado porcentaje de tokens enmascarados. El problema con la implementación original es el hecho de que los tokens elegidos para el enmascaramiento de una secuencia de texto dada en diferentes lotes son a veces los mismos. Más concretamente, el conjunto de datos de entrenamiento se duplica 10 veces, por lo que cada secuencia se enmascara sólo de 10 formas distintas. Teniendo

²²<https://albertauyeung.github.io/2020/06/19/bert-tokenization.html>

en cuenta que BERT ejecuta 40 épocas de entrenamiento, cada secuencia con el mismo enmascaramiento se pasa a BERT cuatro veces. Según los investigadores, es ligeramente mejor utilizar enmascaramiento dinámico, lo que significa que el enmascaramiento se genera de forma única cada vez que se pasa una secuencia al BERT. En general, esto se traduce en menos datos duplicados durante el entrenamiento, lo que da al modelo la oportunidad de trabajar con datos y patrones de enmascaramiento más variados [Liu *et al.*, 2019]

2.5. SISTEMA DE PERMISOS DE ANDROID

Los permisos de las aplicaciones en Android tienen por objetivo mantener segura la privacidad del usuario, ya que protegen el acceso a datos restringidos y acciones restringidas. Algunos ejemplos de datos restringidos son tu información personal, contraseñas almacenadas en el dispositivo o la información de los contactos de tu agenda, en cuanto a los ejemplos de acciones restringidas se encuentran grabar audio o video, realizar llamadas o conectarse a un dispositivo vinculado ²³.

2.5.1. ESTRUCTURA SISTEMA DE PERMISOS ANDROID

La estructura del sistema de permisos de Android tiene los siguientes componentes [Almomani y Khayer, 2020]:

- *Archivo Manifest*: Todas las aplicaciones tienen este archivo que contiene la declaración de todos los permisos que utilizará la aplicación. Además este archivo contiene todos los componentes de la aplicación y a cada uno de estos se le declaran propiedades básicas, incluyendo los permisos que utilizará.
- *Niveles de protección*: Estos niveles de protección se refieren al uso previsto que se le dará a un permiso, los permisos estándar de android tienen un nivel predefinido. Estos niveles se clasifican en 3:
 1. *Nivel de protección normal*: Los permisos que pertenecen a este nivel presentan un nivel bajo de riesgo y estos no son aprobados explícitamente por el usuario, el sistema los aprueba automáticamente. Un ejemplo de esto es el permiso que accede al estado del Wi-fi.
 2. *Nivel de protección riesgosa*: Los permisos que pertenecen a este nivel dan acceso a información privada de los usuarios y a características del dispositivo. Estos a diferencia de los que pertenecen al nivel de protección normal deben ser solicitados al usuario y este aceptarlos. Un ejemplo de esto es el permiso que accede a la localización del teléfono.

²³<https://developer.android.com/guide/topics/permissions/overview?hl=es-419>

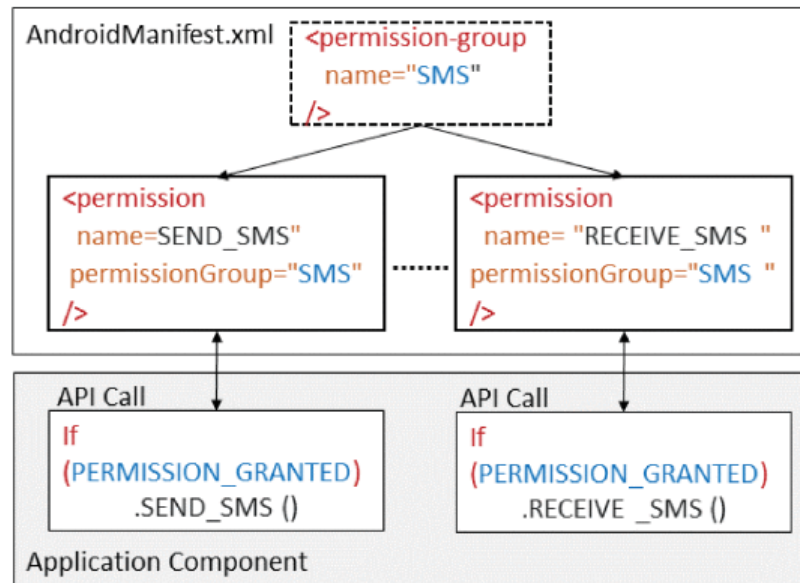


Figura 4: Ejemplo ilustrativo de agrupamiento de permisos.
Fuente: [Almomani y Khayer, 2020]

3. Nivel de protección de firma: Los permisos que pertenecen a este nivel para ser autorizados a las aplicaciones necesitan que la aplicación solicitante y la aplicación declarante comparen sus certificados. Estos permisos serán aprobados si ambos certificados están firmados por la misma clave privada. Esto significa que dos aplicaciones que están firmadas con diferentes certificados no podrán acceder a los mismos permisos, incluso si tienen el mismo nombre de paquete. Este mecanismo ayuda a garantizar que las aplicaciones se originen de una fuente confiable y evita que otras aplicaciones intenten imitar a las aplicaciones legítimas en el mismo dispositivo. Un ejemplo de este nivel es el permiso que accede a las estadísticas de la batería.
- Indicadores de permisos: Estos indicadores se utilizan para controlar de manera más detallada los permisos que son utilizados por las aplicaciones. Los indicadores incluyen el nivel de protección y los permisos. Los indicadores de nivel de protección tienen un rango que va desde cero hasta varios niveles, y si el nivel de protección no está definido en el archivo *Manifest*, se asume que tiene el nivel de protección normal.
- Grupos de permisos: Estos grupos consisten en una clasificación lógica de los permisos, en la figura 4 se muestra un ejemplo de una clasificación del permiso de SMS,

En la figura 4 se muestra como se agrupan los permisos `SEND_SMS` y `RECEIVE_SMS` en el grupo de permisos `SMS`, este se define en el archivo *Manifest*.

Dentro de la estructura del sistema de permisos de Android, estos también se clasifican según el nivel de API (*Level API*) de cada aplicación y de cada dispositivo Android, ya que cada

dispositivo funciona en un nivel de API específico. Cada uno de estos niveles de API es único para cada versión de la plataforma Android. El nivel de API identifica con precisión la versión del conjunto de interfaces de programación de aplicaciones (API) con la que su aplicación puede interactuar. Esto incluye elementos en el archivo de *Manifest*, permisos y otros aspectos con los que los desarrolladores codifican las aplicaciones. El sistema de niveles de API de Android ayuda a determinar si una aplicación es compatible con una imagen del sistema Android antes de instalarla en un dispositivo.

Cuando se construye una aplicación, esta incluye la siguiente información sobre el nivel de API:

- El nivel de API de Android que es el objetivo para el cual la aplicación está diseñada para funcionar.
- El nivel de API mínimo de Android que un dispositivo Android debe tener para ejecutar su aplicación.

Estas configuraciones se utilizan para garantizar que la funcionalidad necesaria para ejecutar la aplicación correctamente esté disponible en el dispositivo Android en el momento de la instalación. Si no está disponible, la aplicación no se ejecutará en ese dispositivo. Por ejemplo, si el nivel de API de un dispositivo Android es inferior al nivel de API mínimo que usted especifica para su aplicación, el dispositivo Android evitará que el usuario instale su aplicación ²⁴.

En la tabla 2 se tienen los niveles de API y a la versión de Android a la cual está asociada, además del año de lanzamiento.

²⁴<https://learn.microsoft.com/en-us/xamarin/android/app-fundamentals/android-api-levels?tabs=windows>

Tabla 2: API Levels y sus versiones de Android.

Fuente: apilevels.com

Versión	SDK / API Level	Nombre código	Año
Android 14	Level 34	Upside Down Cake	2023
Android 13	Level 33	Tiramisú	2022
Android 12	Level 32: Android 12L Level 31: Android 12	Snow Cone	
Android 11	Level 30	Red Velvet Cake	2020
Android 10	Level 29	Quince Tart	2019
Android 9	Level 28	Pie	2018
Android 8	Level 27: Android 8.1 Level 26: Android 8.0	Oreo	2017
Android 7	Level 25: Android 7.1 Level 24: Android 7.0	Nougat	2016
Android 6	Level 23	Marshmallow	2015
Android 5	Level 22: Android 5.1 Level 21: Android 5.0	Lollipop	
Android 4	Level 20: Android 4.4W	KitKat	2013
	Level 19: Android 4.4 Level 18: Android 4.3	Jelly Bean	
	Level 17: Android 4.2 Level 16: Android 4.1	Ice Cream Sandwich	2012
	Level 15: Android 4.0.3 - 4.0.4 Level 14: Android 4.0.1 - 4.0.2		
Android 3	Level 13: Android 3.2 Level 12: Android 3.1 Level 11: Android 3.0	Honeycomb	2011
Android 2	Level 10: Android 2.3.3 - 2.3.7 Level 9: Android 2.3.0 - 2.3.2	Gingerbread	2010
	Level 8: Android 2.2	Froyo	
	Level 7: Android 2.1	Eclair	
	Level 6: Android 2.0.1 Level 5: Android 2.0		
Android 1	Level 4: Android 1.6	Donut	2009
	Level 3: Android 1.5	Cupcake	2008
	Level 2: Android 1.1	Petit Four	
	Level 1: Android 1.0	None	

El total de permisos, definidos por Android, son 314, al día 28-03-2024 ²⁵, y todos están clasificados según el nivel de protección antes mencionado, además algunos están agrupados, en la tabla 3 se muestra un extracto de esto junto a estas características.

²⁵<https://developer.android.com/reference/android/Manifest.permission>

Tabla 3: Clasificación permisos Android.

Fuente: Elaboración propia

Permiso	Grupo	Niveles de protección
ACCEPT_HANDOVER	Phone	Dangerous
ACCESS_BACKGROUND_LOCATION	Location	Dangerous
ACCESS_COARSE_LOCATION	Location	Dangerous
ACCESS_FINE_LOCATION	Location	Dangerous
ACCESS_MEDIA_LOCATION	-	Dangerous
ACTIVITY_RECOGNITION	ACTIVITYRECOGNITION	Dangerous
ADD_VOICEMAIL	Phone	Dangerous
ANSWER_PHONE_CALLS	Phone	Dangerous

2.5.2. PERMISOS ANDROID RIESGOSOS

El sistema de permisos de Android agrupa los permisos peligrosos o riesgosos. En la imagen 5 se muestran los grupos y los permisos que estos contienen.

- Árboles de permisos: Un árbol de permisos consiste en un *namespace* que define una familia de permisos. La aplicación que define el nombre base del árbol tiene acceso como propietario a todos los permisos declarados que están en la jerarquía. Este sistema de árboles de permisos permite organizar y gestionar los permisos de manera estructurada y controlada, lo que facilita el control de los permisos. Un ejemplo de este árbol de permisos se puede tomar con los permisos relacionados con la cámara. Árbol de Permisos de Cámara:

- android.permission.CAMERA
- android.permission.CAMERA:takePhoto (permiso para tomar fotos)
- android.permission.CAMERA:recordVideo (permiso para grabar videos)
- android.permission.CAMERA:accessStorage (permiso para acceder al almacenamiento)

En este ejemplo, el árbol de permisos de cámara define un permiso base android.permission.CAMERA. Bajo este permiso base, están los tres permisos secundarios que son más específicos.

La aplicación que establece este árbol de permisos como propietaria, tendrá acceso a todos estos permisos relacionados con la cámara. Esto facilita la gestión y control de los permisos necesarios para las funcionalidades de la cámara en la aplicación.

- Cumplimiento de los permisos: Este punto refiere a que si una aplicación solicita un permiso, primero se chequea su nivel de API y la versión del SDK de destino de la aplicación. Si el SDK de destino es inferior al 23 todos los permisos que solicite la aplicación

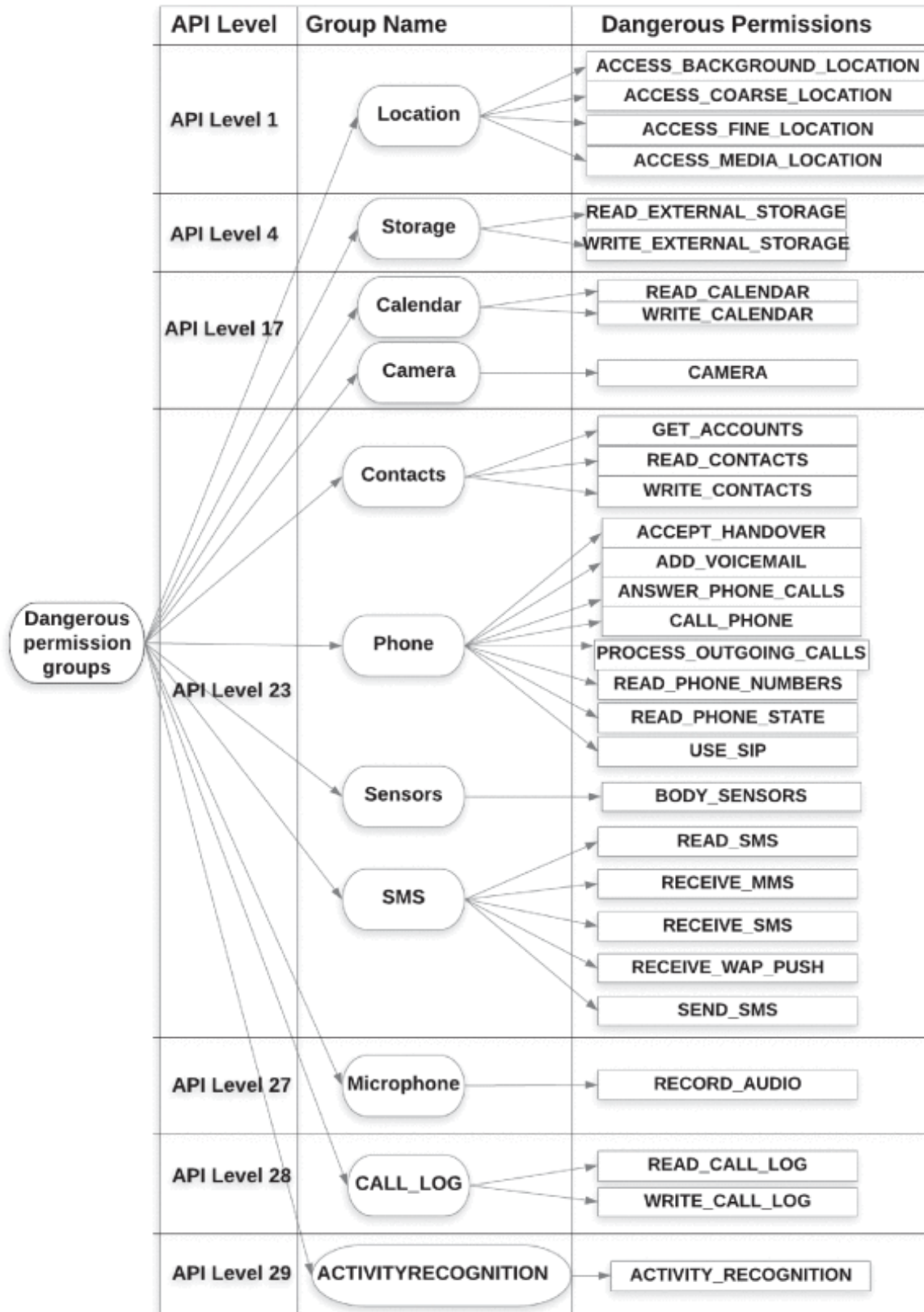


Figura 5: Grupos de permisos riesgosos.

Fuente: [Almomani y Khayer, 2020]

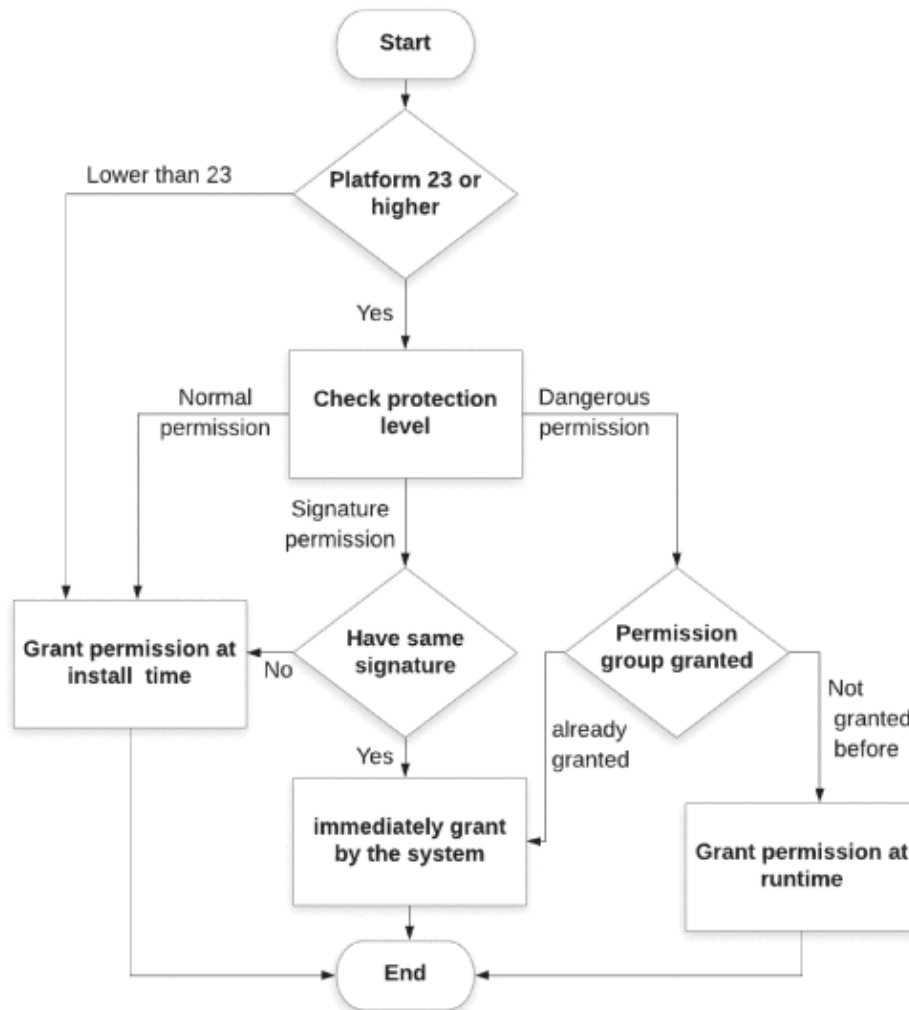


Figura 6: Permission granting based on protection level.
Fuente: [Almomani y Khayer, 2020]

serían autorizados al momento de la instalación, sin notificarle al usuario. Este proceso se ilustra en la figura 6,

2.6. ESTADO DEL ARTE

Las soluciones encontradas en la literatura se basan en la premisa que al existir una gran popularidad de los *smartphones*, existen una gran cantidad de aplicaciones hechas por terceros en las tiendas de *apps*, especialmente en Android.

2.6.1. WHYPER

En el artículo “WHYPER: Towards Automating Risk Assessment of Mobile Applications” [Rahul Pandita y Xie, 2013], se presenta el primer paso para abordar el reto de mantener el *malware* fuera de los mercados de aplicaciones. En concreto, se centra en los permisos para una aplicación determinada y se examina si la descripción de la aplicación proporciona alguna indicación de por qué la aplicación necesita un permiso. Para lograr esto se desarrolla WHYPER, un framework que utiliza técnicas de NLP para identificar las frases que describen la necesidad de un permiso determinado en la descripción de una aplicación. WHYPER logra una precisión media del 82,8 % y una recuperación media del 81,5 % para tres permisos (libreta de direcciones, calendario y grabación de audio) que protegen recursos sensibles a la seguridad y la privacidad de uso frecuente. Estos resultados son muy prometedores en el uso de técnicas de NLP para salvar la brecha semántica entre las expectativas del usuario y la funcionalidad de la aplicación, lo que contribuye a la evaluación de riesgos de las aplicaciones móviles [Rahul Pandita y Xie, 2013].

2.6.2. AUTOCOG

Google Play, la tienda de aplicaciones de Android, ofrece una descripción que entrega una idea sobre la seguridad de las aplicaciones, además provee una API con los permisos que pueden solicitar los desarrolladores en sus apps para acceder a la seguridad y privacidad de los usuarios. AutoCog es un sistema que utiliza esta información para analizar las descripciones de las aplicaciones y así analizar si cumplen con el término que ellos acuñaron, fidelidad entre descripción o permiso o en inglés, *description-to-permission fidelity*, este utiliza técnicas de NLP y sus propios algoritmos para relacionar las descripciones con los permisos. Al realizar sus pruebas con 11 permisos tuvieron una precisión media de 89.8 % y un *recall* medio de 89.7 %. El término que crearon, *description-to-permission fidelity* se puede entender con el siguiente ejemplo, si una aplicación se describe como una red social, esto probablemente significa que tendría los permisos de los contactos, o también si es un juego como Candy Crush, no debería pedir permisos sobre la ubicación del dispositivo. Este abuso excesivo de permisos innecesarios podría ser un indicio de *malware* o que es una app que invade la privacidad [Zhengyang Qu, 2014].

2.6.3. AC-NET

El framework presentado en el *paper* titulado “AC-Net: Assessing the Consistency of Description and Permission in Android Apps” [Feng *et al.*, 2019] aborda el problema de que, en ocasiones, la descripción de una aplicación no refleja completamente los permisos requeridos por la misma. Esto puede resultar en una falta de transparencia y comprensión por parte de los usuarios sobre las capacidades y el acceso que la aplicación puede tener.

Como solución a este problema, el artículo propone un framework denominado AC-Net (Assessing Consistency based on Neural Network), que utiliza técnicas de aprendizaje automático y se aprovecha del procesamiento de lenguaje natural. AC-Net se ha desarrollado con el propósito de evaluar la coherencia entre la descripción de las aplicaciones y los permisos que solicitan mediante el uso de redes neuronales. Este framework toma la descripción de una aplicación como entrada y devuelve una medida de coherencia con los permisos solicitados por la aplicación para su funcionamiento. Además, en el mismo artículo se introduce un nuevo modelo de aprendizaje para analizar las representaciones semánticas de las descripciones de las aplicaciones, conocido como *TextGRU*. Este modelo es una variante de la arquitectura de RNN con GRU. La entrada de la red consiste en una de las oraciones, que es una secuencia de índices de palabras. La salida de la red incluye distribuciones de probabilidad binarias para cada permiso. El modelo se desglosa en cuatro capas, y proporcionamos detalles específicos sobre cada una de ellas en la implementación.

Sin embargo, esta herramienta presenta limitaciones en su implementación. Puede generar falsas alertas en los casos en que las aplicaciones tengan descripciones poco explicativas o superficiales, es decir, que no detallen todos los aspectos de la aplicación. Además, se realiza una comparación con otras herramientas presentadas en papers de años anteriores, como AutoCog y Whyper. Esta comparación se centra en la eficiencia al entrenar el modelo y al utilizarlo para predecir resultados. AC-Net requiere más tiempo en comparación con las otras opciones al momento de entrenar debido a que es una red neuronal profunda, pero al momento de predecir el tiempo es similar a AutoCog y Whyper.

A diferencia de AutoCog y Whyper, este artículo se enfoca en evaluar los permisos en descripciones de granularidad más fina, es decir, las múltiples correspondencias entre oraciones de descripción y permisos. Además, representa el primer intento de extender técnicas de aprendizaje profundo para predecir permisos de aplicaciones basados en oraciones de descripción. Los resultados experimentales demuestran la efectividad de esta herramienta [Feng *et al.*, 2019].

2.6.4. FCDP

En el *paper* “FCDP: Fidelity Calculation for Description-to-Permissions in Android Apps” [Wu *et al.*, 2021b] se propone un enfoque denominado *Fidelity Calculation for Description-to-Permissions (FCDP)*. Su objetivo es ayudar a los auditores del mercado de aplicaciones a evaluar si la descripción de una aplicación indica toda la información relacionada con los permisos peligrosos utilizando una fidelidad cuantificada para proporcionar una descripción de alta calidad a los usuarios móviles. Esta fidelidad cuantificada es un indicador que mide qué tan precisa y detallada es una descripción de una aplicación con respecto a los permisos que esa aplicación podría solicitar o utilizar. En este contexto, la fidelidad cuantificada busca evaluar si la descripción proporciona una representación precisa de los permisos peligrosos o delicados que la aplicación puede requerir. Cuanto mayor sea la fidelidad cuantificada, más precisa será la descripción en relación con los permisos de la aplicación. Además, se inves-

tigó experimentalmente el efecto de diferentes factores en FCDP, y se demostró que FCDP supera al método de vanguardia como AC-NET, en más de un 3,65 % en la predicción de *description-to-permissions*. Mediante el uso de 64.265 descripciones de Android rastreadas desde Google Play, este análisis en profundidad indica además que la mayoría de las descripciones de aplicaciones no revelan completamente la semántica de los permisos peligrosos para los usuarios de móviles [Wu *et al.*, 2021b].

2.6.5. SMARTPI

Otra de las soluciones que han sido propuestas es SmartPI, es un framework que automáticamente identifica las funcionalidades relevantes desde las *reviews*, así infiriendo los permisos implicados en estos, brindando una perspectiva diferente a las funcionalidades de la aplicación, evidenciando una diferencia entre lo que realmente hace la aplicación y lo que los desarrolladores dicen que hace. Con este *framework* se analizaron más de 20.000 apps y 4.247.769 reseñas de los usuarios. Los detectores de *malwares* obtienen sus resultados realizando un análisis de código estático y dinámico, enfocándose en lo que realiza la aplicación, sin tomar en consideración las funcionalidades que dicen los desarrolladores que tiene. SmartPI se diferencia de AutoCog en que este último solo se enfoca en las descripciones de las apps y la documentación de la API de Android para realizar el reconocimiento de los permisos que utilizaba la aplicación, en cambio, SmartPI utiliza tanto las descripciones de las apps, la documentación de la API, la documentación de los permisos que tiene disponible Android, todo esto junto a las reseñas de los usuarios logrando el reconocimiento eficaz de los permisos. Los resultados de este estudio se ven resumidos dentro de la figura 7 que se compara directamente con AutoCog.

En resumen, SmartPI supera a AutoCog en términos de rendimiento en general. SmartPI logró una precisión del 91.8 %, un *recall* del 90.6 % y un *accuracy* de 92.4 %. En comparación, AutoCog obtuvo una precisión del 89.8 %, un *recall* del 89.7 % y un *accuracy* de 90.5 %. Aunque la diferencia es relativamente pequeña, sigue siendo significativa, lo que refleja un avance notable en estas herramientas [Wang *et al.*, 2020].

2.6.6. FIDEDROID

El *paper* “Enhancing Fidelity of Description in Android Apps With Category-Based Common Permissions” [Wu *et al.*, 2021a] aborda el problema común de las descripciones incoherentes en las aplicaciones de Android en relación con los permisos requeridos. Para solucionar este problema, se propone el framework FideDroid, que utiliza permisos comunes basados en categorías para mejorar la fidelidad de las descripciones de las aplicaciones. Este enfoque se basa en la idea de que muchas aplicaciones comparten funcionalidades esenciales que se pueden agrupar en categorías de permisos comunes.

System	Permission	TP	FP	FN	TN	Prec(%)	Rec(%)	F(%)	Accu(%)
<i>SmartPI</i>	SEND_SMS	51	6	6	83	89.5%	89.5%	89.5%	91.8%
	READ_SMS	48	4	5	82	92.3%	90.6%	91.4%	93.5%
	CALL_PHONE	46	3	6	95	93.9%	88.5%	91.1%	94%
	ACCESS_COARSE_LOCATION	38	6	8	110	86.4%	82.6%	84.4%	91.4%
	ACCESS_FINE_LOCATION	56	7	3	92	88.9%	94.9%	91.8%	93.7%
	READ_CONTACTS	58	7	9	75	89.2%	86.6%	87.9%	89.3%
	RECORD_AUDIO	89	7	11	83	89.2%	86.6%	87.9%	89.3%
	BODY_SENSORS	119	9	8	67	93%	93.7%	93.3%	91.6%
	CAMERA	63	6	7	78	91.3%	90%	90.6%	91.6%
	READ_PHONE_STATE	69	2	3	88	97.2%	95.8%	96.5%	96.9%
	Total	637	57	66	853	91.8%	90.6%	91.2%	92.4%
<i>AutoCog</i>	SEND_SMS	51	7	4	76	87.9%	92.7%	90.3%	92%
	READ_SMS	56	6	7	86	90%	88.9%	89.6%	91.6%
	CALL_PHONE	21	3	8	34	87.5%	72.4%	79.2%	83.3%
	ACCESS_COARSE_LOCATION	36	5	9	87	87.8%	80%	83.7%	89.8%
	ACCESS_FINE_LOCATION	48	5	7	93	90.6%	87.3%	88.9%	92.2%
	READ_CONTACTS	69	8	6	81	89.6%	92%	90.8%	91.5%
	RECORD_AUDIO	107	8	6	83	93%	94.7%	93.9%	93.1%
	BODY_SENSORS	67	7	15	60	90.5%	81.7%	85.9%	85.2%
	CAMERA	62	8	6	78	88.6%	91.2%	89.6%	90.9%
	READ_PHONE_STATE	63	9	6	79	87.5%	91.3%	89.4%	90.4%
	Total	580	66	74	757	89.8%	88.7%	89.2%	90.5%

Figura 7: Results of Review to Permission Inference.

Fuente: [Wang et al., 2020]

El framework FideDroid realiza una evaluación de la coherencia entre los permisos inferidos y los utilizados por la aplicación, revelando los permisos sospechosos e innecesarios y ayudando a los desarrolladores a refinar las descripciones de las aplicaciones y a optimizar el uso de los permisos. Los experimentos realizados en la investigación del *paper* se llevaron a cabo sobre aplicaciones del mundo real y demostraron que los permisos comunes basados en categorías pueden cubrir funcionalidades no mencionadas en la descripción de la aplicación, sin tener que considerar todos los permisos posibles.

Además, el *paper* identifica tres factores que contribuyen a la incoherencia entre las descripciones y los usos de los permisos:

- **Intervenciones humanas en la redacción de la descripción:** Los desarrolladores escriben las descripciones de las aplicaciones para promocionarlas, lo que provoca que las descripciones de las aplicaciones no sean coherentes con su comportamiento real.
- **Malas prácticas en el uso de los permisos:** En el estudio, se examinó si las *API/URIs* se utilizaban en el código fuente. Dentro de estos grupos de permisos, se observó que los permisos relacionados con Calendario y Sensores tenían tasas de uso muy bajas, sólo un 2 % y un 7 %, respectivamente. En términos generales, se encontró que más del 50 % de los permisos (67,7 %) se destinaban a bibliotecas de terceros y permisos innecesarios.
- **Desarrolladores prolíficos:** En un estudio del *paper* con 12 desarrolladores prolíficos se reveló que siete de los 12 desarrolladores utilizaban el mismo archivo *Android Manifest* para sus aplicaciones, con la única diferencia siendo el nombre del paquete. Cuatro desarrolladores utilizaban archivos *Android Manifest* diferentes para sus aplicaciones. Se seleccionaron dos de estos desarrolladores al azar y se inspeccionaron manualmente sus aplicaciones. Se descubrió que utilizaban permisos de manera diferente para las mismas partes de la descripción de la aplicación, lo que sugiere que los desarrolladores prolíficos tienden a reutilizar su código durante el desarrollo de aplicaciones.

Estos hallazgos brindan un conocimiento valioso para los desarrolladores, permitiéndoles mejorar y perfeccionar las descripciones de las aplicaciones y optimizar el uso de los permisos, lo que a su vez mejora la experiencia del usuario y la transparencia en el acceso y las capacidades de la aplicación [Wu *et al.*, 2021a].

CAPÍTULO 3

PROPUESTA DE SOLUCIÓN

Como solución al problema planteado, se emplean técnicas de *Machine Learning* que implican la creación de un modelo de clasificación de textos multi-etiqueta. Este modelo utiliza las descripciones de las aplicaciones en Play Store para predecir los permisos que cada aplicación utilizará. Posteriormente, estos permisos predichos se comparan con los permisos que las aplicaciones declaran en el archivo Manifest, lo que permite calcular un delta de error.

En los siguientes subcapítulos se detallarán las librerías utilizadas para la creación del modelo y la estructura del conjunto de datos, además de una sección que muestra el stack tecnológico utilizado para la solución.

3.1. ESTRUCTURA DE LA SOLUCIÓN

En la figura 8 se muestra la estructura general de la solución propuesta.

En la figura 9 se muestra el diagrama de flujo del funcionamiento de la herramienta.

3.2. STACK TECNOLÓGICO

En la tabla 4 se muestra el stack tecnológico utilizado para el desarrollo de la solución propuesta.

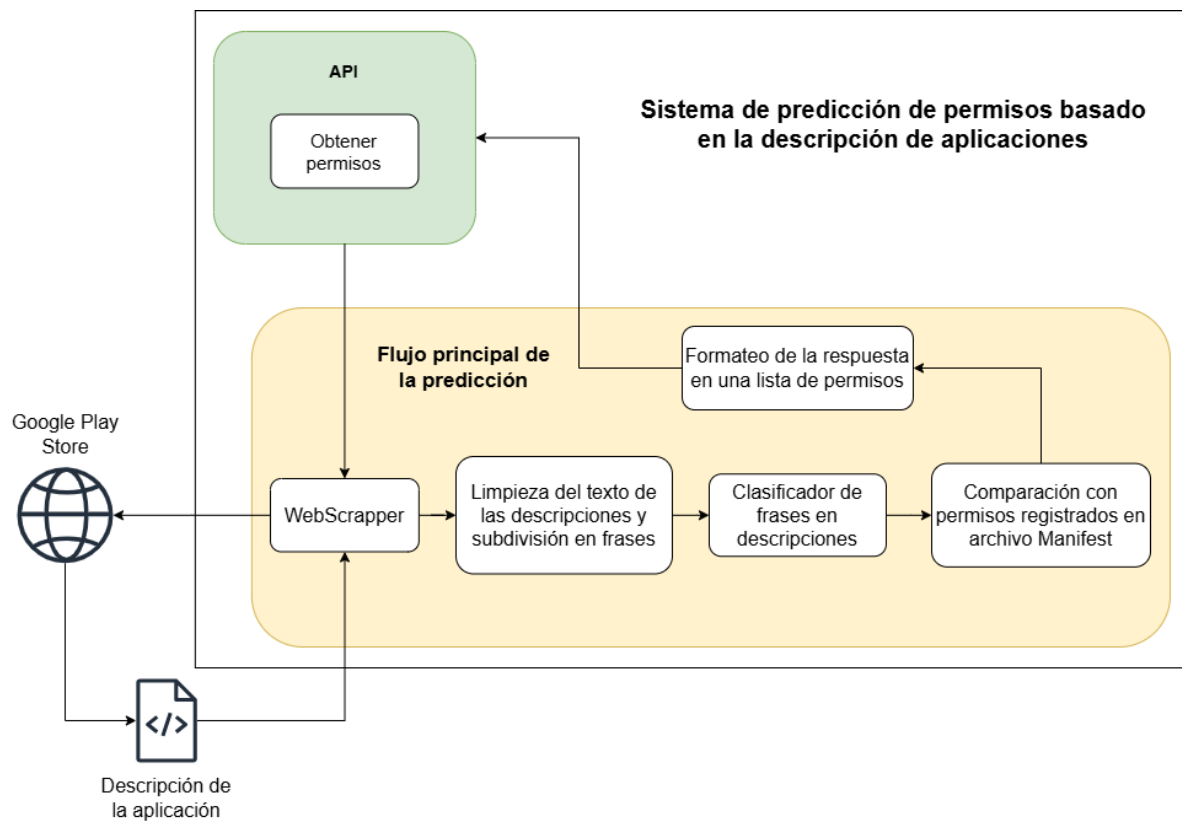


Figura 8: Esquema de la estructura de la solución.
Fuente: Elaboración propia.

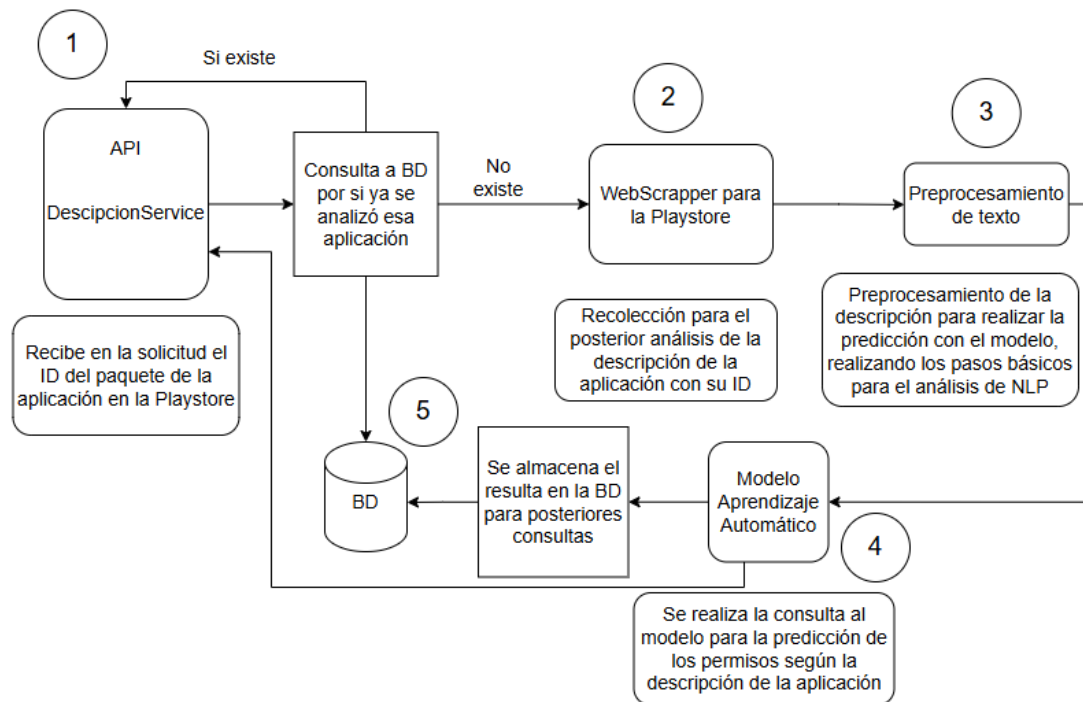


Figura 9: Diagrama de flujo de la solución.
Fuente: Elaboración propia.

Tabla 4: Stack tecnológico.
Fuente: Elaboración propia

Nombre	Versión	Tipo	Utilidad
Python	3.10	Lenguaje de programación	Desarrollo de las técnicas, API y análisis de datos.
TensorFlow	2.16.1	Librería	Se utiliza para construir modelos de aprendizaje automático y redes neuronales.
PyTorch	2.2.1+cu121	Librería	Se utiliza para construir, entrenar y desplegar modelos de aprendizaje profundo.
Scikit-learn	1.2.2	Librería	Se utiliza para preprocesamiento de datos, evaluación de modelos y selección de características.
PyTorch Lightning	2.2.1	Librería	Se utiliza para entrenar modelos de aprendizaje profundo con PyTorch.
Transformers	4.38.2	Librería	Se utiliza para trabajar con modelos de lenguaje pre-entrenados basados en la arquitectura Transformer, como BERT, GPT o RoBERTa.

3.3. CONJUNTO DE DATOS

En primer lugar, para implementar la solución, se utilizó un conjunto de datos o *dataset* en el entrenamiento del modelo de aprendizaje automático que originalmente es del artículo *AC-Net: Assessing the Consistency of Description and Permission in Android Apps* [Feng et al., 2019]. Este conjunto de datos consiste en una colección de ejemplos de aplicaciones de Android, donde cada ejemplo está compuesto por una oración de descripción y una lista de permisos asociados solo a esta; es decir, para cada aplicación, existen múltiples filas dentro del *dataset*.

La columna de descripción contiene texto que describe las funcionalidades y características de la aplicación, mientras que las columnas representan los diferentes permisos que la aplicación solicita para acceder a determinados recursos del dispositivo, designadas por el sistema Android. Estos permisos actúan como etiquetas o clases en el proceso de clasificación, donde el objetivo es predecir estos basándose en la descripción de la aplicación.

Durante el proceso de entrenamiento, se emplean técnicas de procesamiento de lenguaje natural para llevar a cabo el preprocesamiento y la transformación de las descripciones de

las aplicaciones en representaciones numéricas adecuadas para el modelo de aprendizaje automático. En este caso, el conjunto de datos se procesa utilizando el *tokenizador* de BERT. En la figura 10 se muestra la estructura del conjunto de datos, el cual está compuesto por una columna que contiene las descripciones de las aplicaciones, mientras que el resto de las columnas contienen valores binarios que representan si dichas descripciones incluyen el permiso correspondiente a esa columna.

description	STORAGE	CONTACTS	LOCATION	CAMERA	MICROPHONE	SMS	CALL_LOG	PHONE	CALENDAR	SETTINGS	TASKS
ROOT is REQUIRED for automatic synchronization...	0	0	0	0	0	0	0	0	0	0	0
I'm looking for the solution at the moment	0	0	0	0	0	0	0	0	0	0	0
Please contact me if you know how to fix it	0	0	0	0	0	0	0	0	0	0	0
ClockSync synchronizes device system clock wit...	0	0	0	0	0	0	0	0	0	0	0
Useful if provider doesn't support NITZ# sends...	0	0	0	0	0	0	0	0	0	0	0

Figura 10: Extracto del dataset.

Fuente: Elaboración propia.

Para una descripción más detallada, se tiene lo siguiente: la columna “description” contiene las frases en las que se divide la descripción completa de la aplicación. Esto implica que, por cada aplicación, se pueden tener una o más filas. Las siguientes columnas determinan si hay una relación con los permisos correspondientes, utilizando un 0 si no hay relación y un 1 en caso contrario.

- **STORAGE:** Indica si la frase está relacionada con los permisos de almacenamiento del dispositivo.
- **CONTACTS:** Indica si la frase está relacionada con los permisos para acceder a los contactos guardados en el dispositivo.
- **LOCATION:** Indica si la frase está relacionada con los permisos de localización, ya sea aproximada o precisa, del dispositivo.
- **CAMERA:** Indica si la frase está relacionada con el uso de la cámara del dispositivo, tanto para tomar fotos como para grabar videos.
- **MICROPHONE:** Indica si la frase está relacionada con el uso del micrófono del dispositivo, por ejemplo, para grabar audios.
- **SMS:** Indica si la frase está relacionada con el registro y envío de mensajes de texto.
- **CALL_LOG:** Indica si la frase está relacionada con el acceso al registro de llamadas.
- **PHONE:** Indica si la frase está relacionada con la realización de llamadas desde el dispositivo.

- CALENDAR: Indica si la frase está relacionada con el acceso al calendario del dispositivo.
- SETTINGS: Indica si la frase está relacionada con el acceso a las configuraciones del dispositivo.
- TASKS: Indica si la frase está relacionada, por ejemplo, con la finalización de procesos.

3.4. MODELO DE APRENDIZAJE AUTOMÁTICO

Para la creación del modelo de clasificación, se empleó el modelo pre entrenado BERT. Este al ser un modelo de lenguaje basado en *transformers*, es utilizado para tareas de procesamiento de lenguaje natural. Este modelo fue utilizado en conjunto con una red neuronal creada con la librería PyTorch y su framework Lightning y otra red neuronal con Tensorflow y Keras, los cuales son ampliamente utilizados para tareas de clasificación de texto. Con BERT se realiza un sobreentrenamiento o *fine-tuning* con el *dataset* y así adaptar el modelo al problema que se necesita resolver, en este caso un clasificador multi-etiqueta.

Lo que se espera de esta solución con el clasificador multi-etiqueta es que al utilizar BERT se puedan reconocer los patrones que utilizan los desarrolladores al momento de crear una descripción en la Playstore y sean relacionados a los permisos necesarios para ejecutar las características o funciones que se describen en dicho texto. Un ejemplo de estos posibles patrones se puede encontrar en la descripción que tiene la aplicación Instagram:

Little moments lead to big friendships. Share yours on Instagram. —From Meta

Connect with friends, find other fans, and see what people around you are up to and into. Explore your interests and post what's going on, from your daily moments to life's highlights. Share what you're up to and into.

- *Keep up with friends on the fly with Stories and Notes that disappear after 24 hours.*
- *Start group chats and share unfiltered moments with your Close Friends.*
- *Share memories from recent events or trips in Feed.*
- *Turn your life into a movie and discover short, entertaining videos on Instagram with Reels.*
- *Customize your posts with exclusive templates, music, stickers and filters*

Dive into your interests.

- *Watch videos from your favorite Creators and discover new content that's personalized to your interests.*

- *Get inspired by photos and videos from new accounts in Explore.*

- *Discover brands and small businesses, and shop products that are relevant to your personal style.*

Some Instagram features may not be available in your country or region.

De la frase * *Share memories from recent events or trips in Feed...* se espera que el modelo encuentre el patrón de texto que asocie este texto con los permisos relacionados con la cámara tanto para tomar fotos y grabar videos al utilizar la expresión compartir recuerdos.

CAPÍTULO 4

VALIDACIÓN DE LA SOLUCIÓN

Para el desarrollo de la solución, se utilizó el lenguaje de programación Python en su versión 3.10. Se empleó la plataforma de Google Colab en su versión gratuita para entrenar el modelo de aprendizaje automático. El entorno de Colab contaba con una CPU de 1 núcleo y 2 hilos, 12.7 GB de RAM y una GPU tipo T4 con 15 GB de RAM. Además en conjunto se utilizó la plataforma Kaggle, similar a Google Colab, ofrece un servicio de notebooks donde se trabaja en su mayoría en problemas de aprendizaje automático. Las características del hardware de esta plataforma son una CPU Intel(R) Xeon CPU @ 2.00 GHz, con 13 GB de RAM y una GPU tipo T4 similar a la que se utiliza en Google Colab.

Se utilizaron dos plataformas ya que al utilizar sus versiones gratuitas las horas que se podían utilizar las GPUs era limitado semanalmente.

La elección de Google Colab y Kaggle para el desarrollo y entrenamiento del modelo de clasificación se basó en su practicidad. En primer lugar, ofrecen de forma gratuita un entorno de ejecución especializado para el desarrollo de soluciones con Machine Learning. Esto incluye acceso a recursos computacionales como una GPU dedicada para el procesamiento de algoritmos de aprendizaje automático, así como la posibilidad de utilizar una TPU (Tensor Processing Unit) que acelera el entrenamiento de modelos.

Además, tienen preinstaladas las librerías y herramientas más utilizadas en el campo del aprendizaje automático, como TensorFlow, PyTorch y Scikit-learn. Esto facilita enormemente el proceso de importar y utilizar estas librerías sin tener que instalarlas manualmente.

Una ventaja única de utilizar Google Colab es su integración con otros servicios de Google, como Google Drive. Esto permite almacenar y acceder fácilmente a conjuntos de datos y modelos entrenados, así como compartir y colaborar en proyectos con otros usuarios. Kaggle tiene su propio almacenamiento, para los dataset y así poder utilizarlos en diversos proyectos con facilidad.

4.1. EXPERIMENTOS

Como se mencionó anteriormente se realizaron los experimentos con los modelos de aprendizaje automático utilizando las librerías Tensorflow y Pytorch Lightning, esto con el fin de poder desarrollar el modelo que mejor desempeño pudiera obtener con el *dataset* escogido, este para el entrenamiento fue dividido en 15 % para evaluación durante el entrenamiento, otro 15 % para pruebas después del entrenamiento y el resto, un 70 % para el entrenamiento. Basados en las métricas de evaluación de un modelo de aprendizaje automático, las principales con las que se designó el modelo con mejor desempeño es la pérdida y el *F1 score*.

Se utilizan estas primero, porque la pérdida mide cuán bien el modelo se ajusta a los datos de entrenamiento, es decir, indica qué tan equivocadas son las predicciones del modelo durante el entrenamiento respecto a los datos reales y el segundo debido a que se utiliza para modelos de clasificación en que las clases o etiquetas estén desequilibradas el cual es el caso, además un *F1 score* alto indica un buen equilibrio entre la capacidad del modelo para predecir correctamente las instancias positivas y evitar falsos negativos y falsos positivos, lo cual para el objetivo de la solución se quiere disminuir lo más posible así evitando que una aplicación pueda ser designada con algún permiso por error. En los experimentos que se realizaron la mayor dificultad era prevenir el *overfitting*, ya que el *dataset* que se tenía, las clases o etiquetas estaban desequilibradas, además existe una gran mayoría de las filas dentro del conjunto de datos que no tenían etiquetas provocando un sobreajuste y que tuviera un mal desempeño. Para evitarlo primero se realizó un *subsampling* en el *dataset*, eliminando todos los datos que no tuvieran etiquetas para clasificar, así evitando que el modelo los tome en cuenta al momento del entrenamiento, en cada experimento se fueron ajustando los siguientes hiperparámetros para encontrar la combinación que entregue el mejor desempeño,

- Tamaño máximo de los tokens al momento de utilizar el *tokenizador* de BERT
- Tasa de aprendizaje
- Tamaño de los lotes de datos
- Cantidad de epochs
- Capas de *Dropout*

Además, para prevenir el *overfitting*, se incorpora una técnica llamada *Early Stopping*. Esta estrategia monitorea una de las métricas del modelo en cada época y, si durante un número predefinido de épocas dicha métrica muestra un comportamiento no esperado, el entrenamiento se detiene.

4.2. RESULTADOS

Los resultados de los experimentos realizados, están resumidos en tablas, estos estarán divididos en 2, donde la tabla 6 corresponde a los realizados utilizando Tensorflow con *subsampling* de las etiquetas vacías, y la tabla 5 con el otro experimento utilizando Pytorch Lightning con *subsampling* de las etiquetas vacías.

En todos los experimentos finales el largo de las secuencias de los textos es de 30 y la cantidad máxima de *epochs* es 30, solo se varía tanto la tasa de aprendizaje como el tamaño de los lotes y ambos tienen incorporada la técnica *Early Stopping* que monitorea la métrica de la pérdida en el proceso de validación del entrenamiento, con una paciencia de 5 *epochs*.

Tabla 5: Resultados del experimento con Pytorch Lightning.

Fuente: Elaboración propia.

Pytorch Lightning				
Experimentos	Accuracy	F1 Score	Loss	Epochs alcanzados
BS=64, LR=2e-4	0.6787	0.7728	0.1614	13
BS=16, LR=2e-5	0.6667	0.7880	0.1429	24
BS=32, LR=2e-5	0.686	0.7703	0.1594	27
BS=64, LR=2e-5	0.7	0.7652	0.1555	21
BS=64, LR=2e-6	0.594	0.6991	0.1930	30

Tabla 6: Resultados del experimento con Tensorflow.

Fuente: Elaboración propia.

Tensorflow				
Experimentos	Accuracy	F1 Score	Loss	Epochs alcanzados
BS=64, LR=2e-4	0.7271	0.6974	0.1846	12
BS=16, LR=2e-5	0.7467	0.7235	0.1647	12
BS=32, LR=2e-5	0.7303	0.6973	0.1469	12
BS=64, LR=2e-5	0.7459	0.7218	0.1399	12
BS=64, LR=2e-6	0.7459	0.7217	0.1399	12

Los experimentos realizados con la librería Pytorch Lightning entregaron mejores resultados en cuanto a la clasificación de los permisos, esto utilizando de base la métrica *F1 Score*, que indica la relación entre la precisión y la exhaustividad. Un *F1 score* alto indica un buen equilibrio entre la capacidad del modelo para predecir correctamente las instancias positivas y evitar falsos negativos y falsos positivos. Con esta métrica en cuenta el modelo que entrega el experimento que cuenta con un *Batch size* de 16 y una tasa de aprendizaje de 2e-5, es el que entrega un mejor rendimiento, logrando resultados cercanos a los reales.

Estos valores escogidos para los hiperparámetros se ven resumidos en la tabla 7,

Tabla 7: Hiperparámetros utilizados.

Fuente: Elaboración propia

Hiperparámetro	Valor utilizado
Tamaño máximo de los tokens	30
Tasa de aprendizaje	2e-5
Tamaño de los lotes de datos	16
Cantidad de epochs	30

Se utilizó este modelo para clasificar las descripciones de 10 aplicaciones que se encuentran en PlayStore, mostrando las diferencias entre los permisos que la aplicación tiene en su archivo *Manifest* y los predichos por el modelo de clasificación.

4.2.1. Resultados clasificación de aplicaciones

Aplicación: Instagram

Grupos de permisos que utiliza: 'STORAGE', 'CONTACTS', 'LOCATION', 'CAMERA', 'MICROPHONE', 'PHONE'.

Grupos de permisos predichos del modelo: 'CAMERA', 'MICROPHONE', 'SETTINGS'.

En estas oraciones el modelo extrajo que la aplicación podría utilizar el grupo de permisos de 'CAMERA':

- share yours on instagram
- turn your life into a movie and discover short entertaining videos on instagram with reels
- get inspired by photos and videos from new accounts in explore

En esta aplicación la descripción es explícita en cuanto al uso de la cámara por lo que el modelo pudo clasificarlo.

Para el permiso 'MICROPHONE' lo obtuvo con las siguientes frases:

- from meta connect with friends find other fans and see what people around you are up to and into
- share memories from recent events or trips in feed
- discover brands and small businesses and shop products that are relevant to your personal style
- some instagram features may not be available in your country or region
- terms and policies <https://help.instagram.com>

Las frases no indican de manera explícita que la aplicación emplea la captura de audio. Sin embargo, al analizar el contexto, se infiere que a partir de la primera frase "from meta connect with friends..." y "share memories from recent events or trips in feed", se entiende que, al conectarse con amigos y compartir recuerdos de eventos o viajes en la página de Instagram, se podría utilizar el micrófono para capturar audio. El resto de las frases son clasificadas como falsos positivos, ya que el modelo arroja buenos resultados, pero existe un margen de error.

Para el grupo de permisos 'SETTINGS' el modelo clasifica las siguientes frases:

- share what you're up to and into on insta
- customize your posts with exclusive templates music stickers and filters
- watch videos from your favorite creators and discover new content that's personalized to your interests

Ninguna frase tiene relación con el grupo señalado por lo que es un falso positivo que tuvo el modelo durante la clasificación.

Aplicación: Facebook

Grupos de permisos que utiliza: 'STORAGE', 'CONTACTS', 'LOCATION', 'CAMERA', 'MICROPHONE', 'PHONE', 'CALENDAR', 'TASKS'.

Grupos de permisos predichos del modelo: Ninguno.

En este caso, el modelo no logró reconocer ningún permiso dentro de la descripción de la aplicación. Esto se debe a que en todas las oraciones no se especifica ninguna funcionalidad, como la captura de fotos, videos o audio, así como el almacenamiento de estos elementos.

Por ejemplo, en el fragmento de la descripción de la aplicación proporcionado a continuación, se observa que el vocabulario utilizado es específico de la aplicación, evitando términos generales como "realizar videos". En su lugar, se emplea el término "reel", que, dentro del contexto de la aplicación, se refiere a videos cortos. Debido a esta especificidad, el modelo puede no tener suficiente contexto para reconocer que, a partir de la frase "Effortlessly create reels from trending templates ...", la aplicación utilizará los permisos de 'CÁMARA' y 'MICRÓFONO'.

**Effortlessly create reels from trending templates, or let your creativity shine with a full suite of editing tools*

**Customize your profile to choose how you show up and who you share your posts with*

**Turn your hobby into a side hustle by becoming a creator or selling things on Marketplace*

**Celebrate everyday, candid moments with stories, which disappear in 24 hours*

Este es un extracto de la descripción de la aplicación que se encuentra en la Play Store al día del análisis. La utilización de un vocabulario específico de la aplicación en lugar de términos generales puede haber contribuido a la falta de reconocimiento por parte del modelo en relación con los permisos asociados a las funcionalidades específicas mencionadas.

Aplicación: Slack

Grupo de permisos que utiliza: 'CONTACTS', 'CAMERA', 'MICROPHONE'

Grupos de permisos predichos del modelo: 'CAMERA', 'MICROPHONE', 'CALL_LOG', 'SETTINGS'

Para el permiso 'MICROPHONE' el modelo obtuvo la siguiente frase:

- slack brings team communication and collaboration into one place so you can get more work done whether you belong to a large enterprise or a small business

Al utilizar la frase 'slack bring team communication ...' El modelo interpretó la comunicación de equipo a que la aplicación tendrá una función para hablar entre los miembros del equipo en Slack.

La siguiente frase el modelo la interpretó a que la aplicación utilizará el permiso 'CALL_LOG':

- use slack to communicate with your team and organize your conversations by topics projects or anything else that matters to your work message or call any person or group within your team share and edit documents and collaborate with the right people all in slack integrate into your workflow the tools and services you already use including google drive salesforce dropbox asana twitter zendesk and more easily search a central knowledge base that automatically indexes and archives your team s past conversations and files customize your notifications so you stay focused on what matters scientifically proven or at least rumored to make your working life simpler more pleasant and more productive

Esto se debe a que la descripción especifica que en la aplicación puedes mensajear o llamar a otras personas que la utilicen, esto conlleva implícitamente que se llevará un registro de estas llamadas. Este es un falso positivo ya que la llamadas que se realizan dentro de la aplicación no son las mismas que se realizan con el dispositivo, por lo que realmente la aplicación no usa los registros de llamadas del dispositivo.

Para los permisos 'CAMERA' y 'SETTINGS' clasifico las siguientes frases respectivamente:

- check off your to do list and move your projects forward by bringing the right people conversations tools and information you need together
- slack is available on any device so you can find and access your team and your work whether you re at your desk or on the go

En el primer caso la aplicación si utiliza el permiso de 'CAMERA' pero la frase que se clasificó en ese permiso no tiene relación directa, se podría interpretar que clasifica esta frase en 'CAMERA' debido a que menciona herramientas en conversaciones y el tomar fotos pueden se toma como una de estas herramientas.

En el segundo caso la aplicación no utiliza el permiso 'SETTINGS' por lo que la frase fue clasificada erróneamente.

Aplicación: Zoom

Grupo de permisos utilizados: 'STORAGE', 'CONTACTS', 'LOCATION', 'CAMERA', 'MICROPHONE', 'PHONE', 'CALENDAR'.

Grupo de permisos predichos por el modelo: 'CAMERA', 'MICROPHONE', 'CALL_LOG', 'CALENDAR'

El modelo clasificó la siguiente frase dentro del permiso 'CALL_LOG' y 'CALENDAR':

- communicate easily with anyone chat with internal and external contacts place and accept calls or send sms text messages schedule a video meeting and join with a single click keep projects moving chats and channels for messaging file sharing and more share and annotate content during meetings brainstorm on virtual whiteboards share them in chat and access them later work safely on the go stay focused on the road with android auto support hey google voice access commands for hands free control sso support and okta integration bounce between locations move a call or meeting between devices with a single click without interruption use the mobile app to start a zoom rooms meeting and share content multi task on your android phone with picture in picture or tablet with split screen usage information any free or paid license can be used with the mobile app a paid zoom one subscription or add on license is required to use certain product features

Se interpreta que el modelo tomó la sección 'communicate easily with anyone chat with internal and external contacts place and accept calls or send sms text messages ...' para clasificarlo dentro del primer permiso señalado, ya que al realizar llamadas, se infiere que se tendrá acceso al registro de las llamadas realizadas. Este es un falso positivo ya que la llamadas que se realizan dentro de la aplicación no son las mismas que se realizan con el dispositivo, por lo que el problema es el contexto del modelo.

Para el segundo permiso de 'CALENDAR' dentro de esta frase el modelo tomó la sección de agendar una reunión de videollamada, ya que para agendar necesita tener acceso al calendario del sistema.

Con los permisos 'CAMERA' y 'MICROPHONE' el modelo clasificó las siguientes frases respec-

tivamente:

- contact us at <http://support.zoom.us>
- works over wi fi g g lte and g networks support for android phones and tablets follow us on social zoom
- terms of service <https://explore.zoom.us/terms> privacy statement <https://explore.zoom.us/privacy> have a question

La tres frases no tiene relación alguna con el permiso donde fueron clasificadas, esto a causa que al modelo le falta contexto respecto a la aplicación, ya que en la descripción completa la única frase que contiene una señal que se utilizara la cámara y el micrófono es 'Schedule a video meeting ...', al tener la frase de agendar una videoconferencia o videollamada, el modelo no tiene el contexto de esto, interpretó la acción como agendar una videoconferencia, más no realizarla dentro de la aplicación.

Aplicación: OneDrive

Grupo de permisos utilizados: 'STORAGE', 'CONTACTS', 'CAMERA', 'SETTINGS'.

Grupo de permisos predichos por el modelo: 'CAMERA', 'SETTINGS'

El modelo clasifica las siguientes frases para el permiso de 'SETTINGS':

- microsoft onedrive gives you more storage space for your photos and files
- onedrive s cloud storage will secure photos videos files and keep them safe
- keep your files protected synced and accessible on all your devices
- the onedrive app lets you view and share photos files photos and videos with friends and family for safe and free storage
- plus you can use the app to automatically back up your phone s photos and videos
- start with gb of free storage space or upgrade to a microsoft subscription to get up to tb or gb of cloud storage
- microsoft onedrive offers the following features back up photos videos more storage for all of your important files

- upload photos videos documents more automatic photo backup secure photo storage when you turn on camera upload find photos in the photo locker easily with automatic tagging view share photos on your phone computer online free storage photo locker will secure photos keep them safe upload videos keep them in secure photo storage bedtime backup is a dedicated backup state for seamless photo backup while you re snoozing file sharing and access secure photo storage for all of your photos videos albums share files photos videos albums with friends family share photos upload videos easily get notifications when a shared document is edited secure folder settings offer password protected or expiring sharing links access selected onedrive files on the app without being online security all onedrive files are encrypted at rest in transit personal vault protect important files with identity verification in secure folder storage secure photos upload videos keep them safe with secure photo storage restore files with version history stay protected with ransomware detection recovery collaboration with microsoft share files across platforms share photos in the photo locker use microsoft office apps to edit collaborate in real time on word excel powerpoint onenote files stored in onedrive back up view save office documents document scanning scan sign markup send docs right from the onedrive mobile app keep documents safe in a secure folder search search photos by what s in them i e
- search docs by name or content the onedrive app for android offers gb of free cloud storage to sync photos and files across your devices share photos and docs and keep your digital life backed up in the cloud
- microsoft personal family subscription subscriptions begin at a month in the us and can vary by region more storage with tb per person for up people with the family subscription onedrive premium features accessible for everyone in the plan share files folders and photos for specific time windows for added security protect your passwords with password protected sharing links secure file sharing app with added ransomware detection and recovery security features file restore recover files up to days after malicious attacks file corruption or accidental edits or deletions share up to x more content a day with friends and family access premium versions of word excel powerpoint onenote outlook and onedrive microsoft subscriptions and onedrive standalone subscriptions purchased from the app will be charged to your google play store account and will automatically renew within hours prior to the end of the current subscription period unless auto renewal is disabled beforehand
- a subscription cannot be canceled or refunded during the active subscription period

Para este caso la frases en su mayoría mencionan el almacenamiento de archivos, fotos y videos que no tiene relación directa con los permisos de 'SETTINGS', que tratan sobre la modificación de las configuraciones del dispositivo. La frase 'bedtime backup is a dedicated backup state for seamless photo backup while you re snoozing' tiene una relación directa frente al grupo de permisos señalados ya que agrega una funcionalidad que se realiza en momentos donde el teléfono está en reposo, implicaría modificar alguna configuración para que se pueda llevar a cabo.

Para el permiso 'CAMERA' el modelo clasifica las siguientes frases:

- the onedrive app lets you view and share photos files photos and videos with friends and family for safe and free storage
- start with gb of free storage space or upgrade to a microsoft subscription to get up to tb or gb of cloud storage
- search docs by name or content the onedrive app for android offers gb of free cloud storage to sync photos and files across your devices share photos and docs and keep your digital life backed up in the cloud
- microsoft personal family subscription subscriptions begin at a month in the us and can vary by region more storage with tb per person for up people with the family subscription onedrive premium features accessible for everyone in the plan share files folders and photos for specific time windows for added security protect your passwords with password protected sharing links secure file sharing app with added ransomware detection and recovery security features file restore recover files up to days after malicious attacks file corruption or accidental edits or deletions share up to x more content a day with friends and family access premium versions of word excel powerpoint onenote outlook and onedrive microsoft subscriptions and onedrive standalone subscriptions purchased from the app will be charged to your google play store account and will automatically renew within hours prior to the end of the current subscription period unless auto renewal is disabled beforehand
- to manage your subscriptions or to disable auto renewal after purchase go to your google play store account settings
- using this onedrive app for work or school for you to sign into your work or school account on onedrive your organization needs to have a qualifying onedrive sharepoint online or microsoft business subscription plan

La única frase que tiene una relación con los permisos de 'CAMERA' es la primera y la tercera, ya que tratan sobre ver y compartir fotos y videos, el resto no tienen una relación directa con el grupo señalado, tratan sobre suscripción de servicios y almacenamiento, lo que las convierte en falsos positivos.

El resto de permisos que si utiliza la aplicación, 'STORAGE' y 'CONTACTS' se mencionan dentro de la descripción pero no existen frases clasificadas con estos grupos por lo que existen falsos negativos que entregó el modelo.

Aplicación: Mercado Pago

Grupo de permisos utilizados: 'STORAGE', 'CONTACTS', 'LOCATION', 'CAMERA', 'MICROPHONE'

Grupo de permisos predichos por el modelo: 'CAMERA', 'SETTINGS'

Para el grupo de permisos de 'CAMERA' el modelo clasifica las siguientes frases con este:

- with the mercado pago digital account you can always keep your money safe in addition to having a line of credit make payments and transfers online immediately from your cell phone
- enjoy the benefits of a digital account in mercado pago ask for a free card and without annuity
- you can also get a line of credit to pay for services recharge your cell phone and buy in mercado libre
- know the advantages of mercado pago daily returns generate daily returns on your account it s simple and you activate it through the app
- transfer money with a debit card or money in an account and cash in stores
- digital account with the mercado pago digital account it is much easier to pay bills recharge your cell phone and make transfers to any bank or to whomever you want receive payments online and take care of your finances
- take care of your personal finances the digital account works like a virtual wallet that allows you to make payments avoiding paperwork and saving time
- have the facility to pay services with qr code or payment link and transfer money between accounts
- you can use many alternatives to pay for your purchases with a top up balance in your mercado pago account credit debit or rechargeable card cash with transfer bank deposit or request a line of credit
- make purchases online or pay in up to fixed monthly installments with mercado credito without using a credit card
- with your digital account you can also obtain a line of credit to make payments with qr recharges pay for services and purchases in mercado libre using your money as you need
- receive payments however you want payments with qr code payment link through social networks or web payments with the point card reader and start selling online and in person
- charge with qr code or payment link for whatsapp social networks chats and more
- offers the customer the possibility of transferring money and paying through the app to your digital account

- you order it online without formalities or paperwork and the money is deposited instantly in your digital account
- start receiving payments with point or via transfer directly into your online account
- choose the point card reader for your business by going to www.mercadopago.com.mx

De estas frases 17 solo 4 tienen una relación con el grupo señalado estas son:

- have the facility to pay services with qr code or payment link and transfer money
- with your digital account you can also obtain a line of credit to make payments with qr recharges pay for services and purchases in mercado libre using your money as you need
- receive payments however you want payments with qr code payment link through social networks or web payments with the point card reader and start selling online and in person
- charge with qr code or payment link for whatsapp social networks chats and more

Esto debido a que tiene la funcionalidad de utilizar códigos QR por contexto las dos primeras frases utilizarían el grupo de permisos 'CAMERA' ya que se especifican funcionalidades que leen códigos QR y para que el dispositivo lo haga necesita tener acceso a la cámara.

Para el grupo de permisos 'SETTINGS' el modelo clasifica las siguientes frases:

- use the virtual card to make purchases online and in physical stores
- if you also want a physical card request it through the app and we will send it to your home for free
- make all your transfers from one place
- request a physical card or a free virtual card without paying an annuity or issuance
- acquire the point card reader to sell collect and receive payments
- get ready to sell more with mercado pago
- you can sell online or in person and charge the way you want
- product available only for the metropolitan area of cdmx

Ninguna de las frases que se clasificaron dentro de este grupo tienen una relación con este, por lo que es un falso positivo, además que la aplicación no lo utilizó por lo que es un error del modelo en la clasificación.

El modelo no clasificó frases para el grupo de permisos 'CONTACTS', y la descripción de la aplicación hace alusión a la utilización de este, ya que dentro de sus funcionalidades está el compartir información con tus amigos o familia. Por lo tanto el modelo obtuvo un falso negativo en las frases de la descripción.

Aplicación: Google Assistant

Grupo de permisos utilizados: Ninguno

Grupo de permisos predichos por el modelo: 'CAMERA', 'MICROPHONE', 'PHONE', 'CALENDAR', 'SETTINGS'

Para el grupo de permisos 'CAMERA' el modelo clasifica las siguientes frases:

- open youtube show me my photos from hawaii turn on flashlight stay in touch with hands free calls texts and emails stay connected to those who matter most
- make calls send text messages and look up emails to your contacts
- read my unread texts call carly text sam on my way show me my emails from michelle stay productive while on the go get things done using just your voice while on the go set reminders and alarms manage your schedule and tasks

La primera frase hace alusión a las fotos que están guardadas dentro del dispositivo en la siguiente frase 'show me my photos from hawaii ...' por lo que el modelo tuvo un falso positivo porque el grupo de permisos 'CAMERA' es para utilizar la cámara del dispositivo. El resto de frases no tienen relación directa al grupo de permisos por lo que son falsos positivos del modelo.

El grupo de permisos 'MICROPHONE' es clasificado en las siguientes frases:

- it can help you set reminders and alarms manage your schedule look up answers navigate and control smart home devices while away from home and much more
- look up answers and get help with directions and local information
- try saying remind me to buy milk at pm set an alarm for am set a timer for minutes add eggs to my shopping list what s my first meeting tomorrow

- gas stations near me navigate home helps you stay one step ahead get proactive information and contextual reminders to help with just what you need right at the moment you need it
- good morning good night let s go home control your smart home even when away from home your phone is the remote control to your smart home

Las frases 'try saying remind me to buy milk ...' y 'good morning good night let s go home control your smart home ...' tienen relación con el grupo señalado en este punto. En el primero menciona una funcionalidad que da una instrucción al usuario como 'trata de decir' que se puede inferir que la aplicación tiene una función que utiliza el micrófono. La segunda frase la clasifica dentro de este grupo por el contexto de esta, al estar las oraciones 'buenos días' y 'buenas noches', se entiende que el usuario tendrá que decirlas al dispositivo para ejecutar las funciones que menciona la aplicación.

El modelo clasifica las siguientes frases dentro del grupo de permisos 'PHONE':

- compatible devices required an easy way to use your phone and apps quickly open your favorite app navigate your phone and manage your phone settings easily turn on do not disturb adjust your bluetooth and airplane mode settings and turn the flashlight on all using just your voice
- try saying remind me to buy milk at pm set an alarm for am set a timer for minutes add eggs to my shopping list what s my first meeting tomorrow
- adjust the temperature lighting and control your smart appliances even when you re not at home
- try saying turn the lights off broadcast i m on my way to the bedroom speakers compatible devices required

Ninguna de estas frases tiene una relación directa con alguna funcionalidad que pueda llegar a utilizar el grupo de permisos señalado en este punto, por lo tanto es un falso positivo del modelo.

Para el grupo 'CALENDAR' el modelo clasificó las siguientes frases dentro de este:

- make calls send text messages and look up emails to your contacts
- read my unread texts call carly text sam on my way show me my emails from michelle stay productive while on the go get things done using just your voice while on the go set reminders and alarms manage your schedule and tasks

La segunda frase menciona la funcionalidad de 'administrar tu agenda y tareas' por lo que la aplicación necesitaría utilizar el grupo de permisos 'CALENDAR', debido a que dentro del contexto de la aplicación la agenda y las tareas se trabajan con el calendario del dispositivo. La primera frase no menciona una funcionalidad que haga alusión al grupo de este punto por lo que es un falso positivo del modelo.

Por último para el grupo 'SETTINGS' el modelo clasificó la siguiente frase dentro de este:

- and for the things you do regularly you can create automatic routines to streamline your day even more

Esta no especifica ninguna funcionalidad directa que pueda utilizar este grupo de permisos, solo se menciona que 'se pueden crear rutinas automáticas' dentro de la aplicación, pero estas no modifican los ajustes del teléfono o al menos no se puede inferir esto a partir de esta única frase.

Aplicación: Gmail

Grupo de permisos utilizados: 'STORAGE', 'CONTACTS', 'CAMERA', 'MICROPHONE', 'PHONE', 'CALENDAR'

Grupo de permisos predichos por el modelo: 'STORAGE', 'CAMERA'

Para el grupo de permisos 'STORAGE' el modelo clasifica la siguiente oración dentro de este:

- you can connect with coworkers via google meet or google chat send an invite in calendar add an action to your task list and more without leaving gmail use suggested actions like smart reply smart compose grammar suggestions and nudges to help you stay on top of work and take care of simple tasks so you can be more efficient with your time stay safe

Esta frase no tiene una relación directa con el grupo de este punto, ya que este permite si una aplicación tiene permitido escribir en un dispositivo de almacenamiento externo, y en el texto no se especifica ninguna funcionalidad que utilice este grupo de permisos.

El modelo clasifica las siguientes frases dentro del grupo de permisos 'CAMERA':

- the official gmail app brings the best of gmail to your android phone or tablet with robust security real time notifications multiple account support and search that works across all your mail

- gmail is also available on wear os so you can stay productive and manage emails right from your wrist
- with the gmail app you can automatically block more than percent of spam phishing malware and dangerous links from ever reaching your inbox undo send to prevent embarrassing mistakes turn on google chat to connect create and collaborate with others get more done as a group in spaces a dedicated place for organizing people topics and projects enjoy high quality video calling with google meet respond to emails quickly with smart reply suggestions switch between multiple accounts attach and share files with ease get notified of new mail fast with notification center badge and lock screen options search your mail faster with instant results predictions as you type and spelling suggestions organize your mail by labeling starring deleting and reporting spam swipe to archive delete to quickly clear out your inbox read your mail with threaded conversations auto complete contact names as you type from your google contacts or your phone respond to google calendar invites right from the app add a gmail complication and tile on your wear os watch to get a quick overview of your emails gmail is part of google workspace allowing you and your team to easily connect create and collaborate
- our machine learning models block more than of spam phishing and malware from reaching our users learn more about google workspace <https://workspace.google.com/products/gmail> follow us for more [twitter https://twitter.com/googleworkspace](https://twitter.com/googleworkspace) [linkedin https://www.linkedin.com/showcase/googleworkspace](https://www.linkedin.com/showcase/googleworkspace) [facebook https://www.facebook.com/googleworkspace](https://www.facebook.com/googleworkspace)

De estas frases la tercera menciona una funcionalidad relacionada con utilizar la cámara del dispositivo, en esta sección del texto menciona dicha función de la aplicación 'projects enjoy high quality video calling with google meet ...'. El resto de frases no tienen relación con este grupo de permisos por lo que el modelo tuvo falsos positivos con estas.

Aplicación: Google Play Services for AR

Grupo de permisos utilizados: Ninguno

Grupo de permisos predichos por el modelo: 'CAMERA', 'SETTINGS'

Para el grupo de permisos 'CAMERA' clasifica las siguientes oraciones dentro de este:

- <https://developers.google.com/ar/arcore/open-source/licenses> by using this service you agree to be bound by our google terms of service [google tos https://www.google.com/accounts/tos](https://www.google.com/accounts/tos) and google s general privacy policy <https://www.google.com/intl/en/policies/privacy>

- this service is a service as defined in the google tos and the terms regarding software in our services apply to your use of this service

De estas frases, no es posible extraer una funcionalidad explícita de la aplicación. Sin embargo, en la primera frase se hace mención de la tecnología de Realidad Aumentada (AR, por sus siglas en inglés, Augmented Reality). En resumen, la tecnología de Realidad Aumentada busca proporcionar una experiencia mejorada e interactiva del entorno en el mundo real, lograda mediante elementos audiovisuales digitales y otros estímulos sensoriales a través de tecnología holográfica. Para mostrar estos elementos visuales holográficos en el entorno real, el dispositivo requiere de una cámara que los proyecte en conjunto con el entorno, por lo que es necesario contar con el permiso de acceso a la cámara.

En el caso del grupo de permisos 'SETTINGS' el modelo clasifica las siguientes oraciones dentro de este:

- google play services for ar is automatically installed and updated on supported devices
- automatic updates ensure that apps with ar functionality work without requiring an additional download
- install this service and unlock new ways to shop learn create and experience the world together

La segunda frase menciona una funcionalidad respecto a que la aplicación tiene actualizaciones automáticas por lo que se puede inferir que utilizará el permiso relacionado al grupo 'SETTINGS'. El resto de frases no tienen relación con el grupo por lo que son falsos positivos del modelo.

Aplicación: Samsung Members

Grupo de permisos utilizados: 'LOCATION', 'CAMERA', 'MICROPHONE', 'SETTINGS'

Grupo de permisos predichos por el modelo: Ninguno

En el caso de esta aplicación, el modelo no logró clasificar ninguna frase de la descripción en los grupos de permisos que se emplearon como etiquetas durante el entrenamiento del modelo. A continuación, se presenta la descripción de la aplicación tal como se encuentra en la tienda Play Store:

- Discover your possibilities with Samsung Members.

- In here, you'll discover just what you want, plus a whole world of other things you never knew you needed - until now :
- ? Get insider tips to unlock your device's full potential
- ? Easily review device diagnostics
- ? Access exclusive events and benefits
- ? Pose questions directly to the experts and get 1-on-1 customer support
- ? Connect and share with others in the community (Availability and services offered above may vary by country and region)
- ? Note: This application requires network connection (Wi-Fi, 3G/LTE and so on). Additional charges may apply for network connection.

En esta descripción, no se menciona explícitamente ninguna funcionalidad que pueda hacer uso de uno de los grupos de permisos clasificados por el modelo. El problema en este caso radica en que la descripción no especifica de manera clara los permisos que la aplicación utilizará. Los únicos permisos que se describen de manera clara son los relacionados con Wi-Fi y conexión a internet, aunque el modelo no los tiene en cuenta durante la clasificación.

4.2.2. Resumen de resultados

Después del análisis de los resultados, comparando con las descripciones que están en Playstore de las aplicaciones se tuvieron 50 % de falsos positivos, estos resultados se resumen en la tabla 8.

Tabla 8: Resumen análisis aplicaciones.

Fuente: Elaboración propia.

Aplicación	Permisos predichos	Estado	Permisos originales
Instagram	CAMERA MICROPHONE SETTINGS	VERDADERO VERDADERO FALSO POSITIVO	STORAGE CONTACTS LOCATION CAMERA MICROPHONE PHONE
Facebook	NONE		STORAGE CONTACTS LOCATION CAMERA MICROPHONE PHONE

SHERLOCK: A TOOL FOR DETECTING THE MINIMUM ACCESS PERMISSIONS THAT AN APP REQUIRES USING HIS PLAY STORE DESCRIPTION AND ITS CONTEXT AS AN INPUT AND NATURAL LANGUAGE PROCESSING AS A TECHNIQUE

			CALENDAR TASKS
Slack	CAMERA MICROPHONE CALL_LOG SETTINGS	FALSO POSITIVO VERDADERO FALSO POSITIVO FALSO POSITIVO	CONTACTS CAMERA MICROPHONE
Zoom	CAMERA MICROPHONE CALL_LOG CALENDAR	FALSO POSITIVO FALSO POSITIVO FALSO POSITIVO VERDADERO	STORAGE CONTACTS LOCATION CAMERA MICROPHONE PHONE CALENDAR
OneDrive	CAMERA SETTINGS	VERDADERO VERDADERO	STORAGE CONTACTS CAMERA SETTINGS
Mercado Pago	CAMERA SETTINGS	VERDADERO FALSO POSITIVO	STORAGE CONTACTS LOCATION CAMERA MICROPHONE
Google Assistant	CAMERA MICROPHONE PHONE CALENDAR SETTINGS	FALSO POSITIVO VERDADERO FALSO POSITIVO VERDADERO FALSO POSITIVO	NONE
Gmail	STORAGE CAMERA	FALSO POSITIVO VERDADERO	STORAGE CONTACTS CAMERA MICROPHONE PHONE CALENDAR
Google AR	CAMERA SETTINGS	VERDADERO VERDADERO	NONE
Samsung Members	NONE		LOCATION CAMERA MICROPHONE SETTINGS

De esta tabla, destaca el caso de la aplicación de Facebook, ya que el modelo no logró predecir ningún grupo de permisos. Esto se debe a que la descripción de la aplicación no presenta funcionalidades explícitas como otras aplicaciones; en su lugar, utiliza palabras que se entien-

den dentro del contexto de la aplicación, como por ejemplo “reel”, que dentro del contexto se refiere a un video corto creado con la aplicación.

Por otro lado, la aplicación Google AR no tiene originalmente ningún permiso asociado, pero el modelo logró predecir los permisos de la cámara y la modificación de las configuraciones del dispositivo, lo cual es un acierto para resolver el problema planteado.

En el caso de la aplicación Samsung Members, se trata de un error directo en la descripción, ya que no menciona en ninguna frase alguna funcionalidad que haga referencia a la utilización de los permisos mencionados en su archivo *Manifest*, en estos casos el modelo no es efectivo.

Existen casos como el de Zoom, donde la descripción sí menciona funcionalidades relacionadas con los permisos declarados, y el modelo predice algunos permisos, pero clasifica incorrectamente las oraciones. Este error se debe a que durante la etapa de entrenamiento, el modelo no tuvo suficientes ejemplos de prueba para reconocer ese tipo de frases y clasificarlas correctamente.

CAPÍTULO 5

CONCLUSIONES

En la actualidad, donde el uso generalizado de smartphones o dispositivos inteligentes es una realidad, es necesario que la población que hace uso de estos dispositivos cuente con herramientas e información suficientes para evitar riesgos asociados con ciertas aplicaciones. Estos riesgos pueden variar desde el robo de información personal, suplantación de identidad hasta el peligro de caer en estafas. En anteriores capítulos del documento se especifica que los dispositivos Android son considerados más vulnerables en términos de seguridad para el usuario. Esto se debe a que ocupan un puesto destacado a nivel mundial en cuanto a la utilización de este sistema operativo, lo que aumenta la probabilidad de la existencia de usuarios vulnerables a estos peligros.

Las aplicaciones que se encuentran en la tienda de aplicaciones de Android, Play Store, están obligadas a proporcionar una descripción que menciona sus funcionalidades. Sin embargo, no existe un estándar para la redacción de estas descripciones. La manera en que los desarrolladores presentan sus aplicaciones varía significativamente. Algunos utilizan un vocabulario común y detallan las funcionalidades clave de la aplicación, mientras que otros emplean un lenguaje que solo se puede comprender dentro del contexto de la aplicación. Esta diversidad en la forma de describir las aplicaciones complica la comprensión por parte de los usuarios y no deja claro qué datos podría llegar a recolectar o qué periféricos del dispositivo utilizará la aplicación al momento de instalarla, generando así un gran riesgo de desinformación.

La herramienta desarrollada, descrita a lo largo del documento, intenta identificar los permisos que se encuentran dentro de las funcionalidades descritas en las descripciones de las aplicaciones, esta herramienta solo permite automatizar un proceso de comprensión y análisis de coherencia entre las descripciones y los permisos realmente solicitados, como una estrategia para evaluar el riesgo de una aplicación para el cliente. Sin embargo, esta presenta limitaciones al ser desarrollada utilizando un modelo con redes neuronales y BERT, principalmente debido a dos factores. En primer lugar, el modelo en sí mismo, arrojó buenos resultados en este desarrollo, pero no alcanzó la optimización deseada. Esto se debe a la dificultad del problema y a la escasa cantidad de datos de prueba para el entrenamiento del modelo. Al tratarse de un problema de clasificación de texto multi-etiqueta con 11 etiquetas, resulta complicado para el modelo determinar cuáles son las más relevantes para cada una de las frases dentro de las descripciones. Asimismo, la cantidad de etiquetas para el modelo se redujo, ya que para la clasificación se estaban considerando todos los permisos de Android con un nivel de protección peligroso, que son aproximadamente 60 permisos, con esta reducción se logró así mejorar las métricas en los resultados del modelo. El segundo factor que limita la precisión en los resultados del modelo es la calidad de las descripciones proporcionadas por los desarrolladores. Como se mencionó anteriormente, estas descripciones no detallan todas las funcionalidades asociadas a los permisos que utilizarán las aplicaciones. Como resultado, el modelo no cuenta con suficiente contenido textual para identificar todos

los permisos que utilizarían las aplicaciones.

Una sugerencia para mejorar esta situación es que los desarrolladores proporcionen descripciones más claras y completas de las funcionalidades de la aplicación. Estas descripciones deberían utilizar términos comunes que no dependan de un contexto específico de la aplicación. Por ejemplo, en lugar de utilizar términos específicos como “reel” para describir una función de vídeo de corta duración en Facebook, podrían utilizar un lenguaje más genérico y descriptivo que sea comprensible para cualquier usuario. Esto ayudaría al modelo a identificar con mayor precisión los permisos necesarios para cada aplicación.

Para concluir, a partir de los antecedentes entregados, se puede afirmar que la herramienta desarrollada es un acertado primer acercamiento a una futura aplicación que pueda informar a los usuarios y que se convierta en una herramienta útil para aquellos preocupados por la posibilidad de que las aplicaciones que utilizan diariamente no entreguen toda la información respecto a los permisos que utilizan estas. Para el trabajo futuro que perfeccione la herramienta, es necesario recolectar más datos para el entrenamiento del modelo y así conseguir que este contenga más contexto respecto a las posibles aplicaciones y se disminuya la tasa de errores, tanto de falsos positivos como de falsos negativos. Además, se debe entregar más contexto al modelo. Existen artículos donde se ha tratado este problema y las soluciones que desarrollaron incluyen, además de la descripción de la aplicación, las reseñas de los usuarios que entregan aún más información respecto a las funcionalidades relacionadas con los permisos de las aplicaciones.

Para finalizar, se invita a que se sigan realizando investigaciones y desarrollando soluciones que den solución a este problema ya que en los tiempos actuales donde todo el mundo utiliza smartphones y sus aplicaciones es necesario que estén informados sobre los riesgos de estos mismos.

REFERENCIAS BIBLIOGRÁFICAS

- [Almomani y Khayer, 2020] Almomani, I. M. y Khayer, A. A. (2020). A comprehensive analysis of the android permissions system. *IEEE Access*, 8:216671–216688.
- [Cambria y White, 2014] Cambria, E. y White, B. (2014). Jumping nlp curves: A review of natural language processing research [review article]. *IEEE Computational Intelligence Magazine*, 9(2):48–57.
- [Felt et al., 2012] Felt, A. P., Ha, E., Egelman, S., Haney, A., Chin, E., y Wagner, D. (2012). Android permissions: user attention, comprehension, and behavior.
- [Feng et al., 2019] Feng, Y., Chen, L., Zheng, A., Gao, C., y Zheng, Z. (2019). Ac-net: Assessing the consistency of description and permission in android apps. *IEEE Access*, 7:57829–57842.
- [Jacob et al., 2018] Jacob, D., Ming-Wei, C., Kenton, L., y Kristina, T. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding.
- [Kaushal y Chadha, 2022] Kaushal, R. y Chadha, R. (2022). State of the art, recent developments, and future directions in applying deep learning to part of speech tagging in nlp. pp. 38–41.
- [Liu et al., 2019] Liu, Y., Ott, M., NamanGoyal, Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., y Stoyanov, V. (2019). Roberta: Arobustly optimized bert pretraining approach.
- [N y Gupta, 2020] N, T. R. y Gupta, R. (2020). A survey on machine learning approaches and its techniques:. pp. 1–6.
- [Ngobeni y Mhlono, 2019] Ngobeni, A. y Mhlono, S. (2019). Towards enhancing security in android operating systems – android permissions user unawareness. pp. 1–6.
- [Quermann y Degeling, 2020] Quermann, N. y Degeling, M. (2020). Data sharing in mobile apps — user privacy expectations in europe. pp. 107–119.
- [Rahul Pandita y Xie, 2013] Rahul Pandita, Xusheng Xiao, W. Y. W. E. y Xie, T. (2013). Whyper: Towards automating risk assessment of mobile applications.
- [Wang et al., 2020] Wang, R., Wang, Z., Tang, B., Zhao, L., y Wang, L. (2020). Smartpi: Understanding permission implications of android apps from user reviews. *IEEE Transactions on Mobile Computing*, 19(12):2933–2945.
- [Wu et al., 2021a] Wu, Z., Chen, X., Khan, M. U., y Lee, S. U.-J. (2021a). Enhancing fidelity of description in android apps with category-based common permissions. *IEEE Access*, 9:105493–105505.

[Wu *et al.*, 2021b] Wu, Z., Chen, X., y Lee, S. U.-J. (2021b). Fcdp: Fidelity calculation for description-to-permissions in android apps. *IEEE Access*, 9:1062–1075.

[Zhengyang Qu, 2014] Zhengyang Qu, Vaibhav Rastogi, X. Z. Y. C. T. Z. Z. C. (2014). Autocog: Measuring the description-to-permission fidelity in android applications.