



**UNIVERSIDAD POLITÉCNICA SALESIANA**  
**SEDE CUENCA**  
**CARRERA DE COMPUTACIÓN**

DESARROLLO DE UN SISTEMA INTELIGENTE PARA LA GENERACIÓN,  
IDENTIFICACIÓN, CLASIFICACIÓN, REDACCIÓN Y RECOMENDACIONES DE  
NOTICIAS UTILIZANDO GPT-J

Trabajo de titulación previo a la obtención del título  
de Ingeniero en Ciencias de la Computación e  
Ingeniera en Ciencias de la Computación

AUTORES: DANIEL DIONICIO GUZMÁN CHUVA  
DIANA FERNANDA TIXI UYAGUARI  
TUTOR: ING. DIEGO FERNANDO QUISI PERALTA, MGS.

Cuenca - Ecuador  
2022

**CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE  
TITULACIÓN**

Nosotros, Daniel Dionicio Guzmán Chuva con documento de identificación N° 1401069131 y Diana Fernanda Tixi Uyaguari con documento de identificación N° 0302448451; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo de titulación.

Cuenca, 25 de julio del 2022

Atentamente,

---

Daniel Dionicio Guzmán Chuva

1401069131

---

Diana Fernanda Tixi Uyaguari

0302448451

## **CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Daniel Dionicio Guzmán Chuva con documento de identificación N° 1401069131 y Diana Fernanda Tixi Uyaguari con documento de identificación N° 0302448451, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud que somos autores del Proyecto Técnico: “Desarrollo de un Sistema Inteligente para la generación, identificación, clasificación, redacción y recomendaciones de noticias utilizando GPT-J”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Ciencias de la Computación e Ingeniera en Ciencias de la Computación, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, 25 de julio del 2022

Atentamente,

---

Daniel Dionicio Guzmán Chuva

1401069131

---

Diana Fernanda Tixi Uyaguari

0302448451

## **CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN**

Yo, Diego Fernando Quisi Peralta con documento de identificación N° 0104616461, docente de la Universidad Politécnica Salesiana, declaró que bajo mi tutoría fue desarrollado el trabajo de titulación: DESARROLLO DE UN SISTEMA INTELIGENTE PARA LA GENERACIÓN, IDENTIFICACIÓN, CLASIFICACIÓN, REDACCIÓN Y RECOMENDACIONES DE NOTICIAS UTILIZANDO GPT-J, realizado por Daniel Dionicio Guzmán Chuva con documento de identificación N° 1401069131 y por Diana Fernanda Tixi Uyaguari con documento de identificación N° 0302448451, obteniendo como resultado final el trabajo de titulación bajo la opción Proyecto Técnico que cumple con todos los requisitos determinado por la Universidad Politécnica Salesiana.

Cuenca, 25 de julio del 2022

Atentamente,

---

Ing. Diego Fernando Quisi Peralta

1401069131

## RESUMEN

El presente trabajo trata del desarrollo de un sistema inteligente utilizando un modelo de lenguaje autorregresivo (AR) que permite identificar, clasificar, generar y recomendar noticias ecuatorianas a través de tópicos o parámetros ingresados por el usuario, a más de realizar los diferentes servicios mencionados también se cuenta con creación de un repositorio de autoría propia el cual almacena noticias de 46 diarios más importantes del país, con el propósito de consolidar la información y a través de este repositorio se pueda utilizar en los diferentes servicios.

Los servicios que ofrece nuestro sistema inteligente se presentan a través de una aplicación web donde cualquier usuario puede ingresar con la URL o el nombre del dominio. Una de las ventajas de nuestro sistema es que realiza una síntesis del contenido de la noticia con el objetivo que el usuario puede comprender de manera rápida de tal forma se puede abordar más noticias en un mejor tiempo.

La implementación consiste en la reutilización y customización del modelo base de GPT-J el cual se implementó en una TPU de Google Cloud Platform, este modelo es una alternativa a modelos tradicionales de redes neuronales recurrentes (RNN) además de generar servicios y consumirlos a través de una API, se implementó una arquitectura de 3 capas, dentro de la cual se encuentra: capa de presentación, de negocios y datos. En la capa de datos se recolectan y almacenan noticias de diversas fuentes RSS, para la capa de negocios a más de servir los diferentes modelos generados, se realiza los ajustes y customización para la realización de tareas específicas que serán mencionadas en este documento, por último, la capa de presentación permite a los usuarios interactuar con los modelos por lo cual se crea una aplicación web.

**Palabras Clave:** Sistema Inteligente, Modelo de Lenguaje Autorregresivo, GPT-J, TPU, Aplicación Web, API

## ABSTRACT

This work deals with the development of an intelligent system using an autoregressive (AR) language model that allows to identify, classify, generate and recommend Ecuadorian news through topics or parameters entered by the user, in addition to performing the various services mentioned above, we also have created a repository of our own authorship which stores news from 46 major newspapers in the country, in order to consolidate the information and through this repository can be used in different services.

The services offered by our intelligent system are presented through a web application where any user can enter the URL or domain name. One of the advantages of our system is that it performs a synthesis of the content of the news with the objective that the user can understand quickly so that more news can be addressed in a better time.

The implementation consists of the reuse and customization of the GPT-J base model which was implemented in a Google Cloud Platform TPU, this model is an alternative to traditional models of recurrent neural networks (RNN) in addition to generating services and consuming them through an API, a 3-layer architecture was implemented, within which is: presentation, business and data layer. In the data layer, news from different RSS sources are collected and stored, for the business layer, in addition to serving the different models generated, adjustments and customization are made to perform specific tasks that will be mentioned in this document, finally, the presentation layer allows users to interact with the models for which a web application is created.

**Keywords:** Intelligent System, Autoregressive Language Model, GPT-J, TPU, Web Application, API.

# Indice contenido

RESUMEN.....	v
ABSTRACT .....	vi
1. Introducción .....	1
2. Problema .....	2
2.1 Antecedentes .....	2
2.2 Importancia y alcances: .....	3
2.2.1 Justificación .....	3
2.2.2 Grupo Objetivo .....	3
3. Objetivos generales y específicos .....	4
3.1 Objetivo general .....	4
3.2 Objetivos específicos.....	4
4. Revisión de la literatura .....	5
4.1 Procesamiento de Lenguaje Natural.....	5
4.1.1 Generación de Texto.....	5
4.1.2 Clasificación de Texto .....	5
4.2 Machine Learning (ML).....	5
4.3 Deep Learning .....	6
4.4 Redes Neuronales.....	6
4.5 Sistemas Recomendadores .....	6
4.5.1 Basado en Contenido.....	6
4.5.2 Filtrado Colaborativo.....	6
4.6 Generative Pretrained Transformer JAX (GPT-J) .....	7
4.7 Generative Pretrained Transformer 3 (GPT-3) .....	7
4.8 Recolección de Datos .....	7
4.8.1 Data Scraping .....	7
4.8.2 Data Crawler.....	7
4.8.3 Repositorio de Datos .....	8
4.9 RSS.....	8
4.10 Cliente Web.....	8
4.10.1 Aplicación de página única (SPA).....	8
4.10.2 Interfaz de programación de aplicaciones (API) .....	8
4.11 Framework .....	9
4.12 React.js .....	9
4.13 Arquitectura en capas .....	9
4.14 Metodología Scrum .....	10
5. Marco metodológico .....	12
5.1 Propuesta de solución.....	12
5.2 Metodologia Scrum .....	20
5.3 Elaboración de la lista de sprints.....	20
6. Resultados .....	25
6.1 Especificación de requerimientos.....	25
6.1.1 Requerimientos funcionales .....	25

6.1.2	Requerimientos de interfaz.....	26
6.2	Diseño detallado.....	30
6.2.1	Diagrama de actividades.....	30
6.3	Desarrollo.....	33
6.3.1	Desarrollo de las pruebas funcionales.....	33
6.3.2	Desarrollo del código fuente.....	37
6.3.3	Fase de aceptación.....	39
7.	Cronograma.....	46
8.	Presupuesto.....	49
9.	Conclusiones.....	50
10.	Recomendaciones.....	52
	Referencias Bibliográficas.....	53
	Anexos.....	56

## Indice de tablas

Tabla 1: Objetivo Especifico 1 .....	20
Tabla 2:Objetivo Especifico 2 .....	21
Tabla 3: Objetivo Especifico 3 .....	21
Tabla 4: Objetivo Especifico 4 .....	21
Tabla 5: Objetivo Especifico 5 .....	22
Tabla 6: Actividades a realizar en el sprint 1 .....	23
Tabla 7: Actividades a realizar en el sprint 2 .....	23
Tabla 8:Actividades a realizar en el sprint 3 .....	24
Tabla 9: Actividades a realizar en el sprint 4 .....	24
Tabla 10: Prototipo sección principal.....	26
Tabla 11: Prototipo de la sección recomendación .....	27
Tabla 12: Prototipo servicio de recomendar.....	28
Tabla 13: Prototipo servicio de generación de noticias.....	28
Tabla 14: Prototipo servicio de clasificación de noticias .....	29
Tabla 15: Conteo de respuesta - pregunta 1 .....	39
Tabla 16: Conteo de respuesta - pregunta 2 .....	39
Tabla 17: Conteo de respuesta - pregunta 3 .....	40
Tabla 18: Conteo de respuesta - pregunta 4 .....	41
Tabla 19: Conteo de respuesta - pregunta 5 .....	41
Tabla 20: Conteo de respuesta - pregunta 6 .....	42
Tabla 21: Conteo de respuesta - pregunta 7 .....	43
Tabla 22: Conteo de respuesta - pregunta 8 .....	43
Tabla 23: Conteo de respuesta - pregunta 9 .....	44

## Indice de figuras

Figure 1: Propuesta de solución.....	12
Figure 2: Diagrama de flujo para la recolección de datos .....	13
Figure 3: Formato del archivo.txt para entrenar el modelo de clasificación .....	15
Figure 4: Diagrama de flujo del modelo GPT-J Clasificación .....	15
Figure 5: Generación de Texto.....	15
Figure 6: Formato del archivo.txt para entrenar el modelo de generación de texto .....	16
Figure 7: Diagrama de flujo modelo GPT-J Generación.....	17
Figure 8: Similitud semántica .....	17
Figure 9: Diagrama de flujo para la generación de resumen .....	18
Figure 10: Formato del archivo.txt para entrenar el modelo de resumir .....	18
Figure 11: Diagrama de flujo para la generación del resumen .....	18
Figure 12: Conexión entre la aplicación y los servicios.....	19
Figure 13::Diagrama de actividades generar recomendación de noticias.....	30
Figure 14:Diagrama de actividades generar noticias.....	31
Figure 15:Diagrama de actividades clasificar noticias.....	32
Figure 16: Estructura básica del sistema inteligente.....	37
Figure 17:Topologia del docker-compose .....	38
Figure 18:Codigo Fuente .....	38
Figure 19:Resultado pregunta 1 .....	39
Figure 20:Resultado pregunta 2 .....	40
Figure 21:Resultado pregunta 3 .....	40
Figure 22:Resultado pregunta 4 .....	41
Figure 23:Resultado pregunta 5 .....	42
Figure 24:Resultado pregunta 6 .....	42
Figure 25:Resultado pregunta 7 .....	43
Figure 26:Resultado pregunta 8 .....	44
Figure 27:Resultado pregunta 9 .....	44

## **Indice de Anexos**

<b>Anexo 1: Topología recolección de datos .....</b>	<b>56</b>
<b>Anexo 2: Total de noticias recolectadas .....</b>	<b>56</b>
<b>Anexo 3:Parámetros de la red neuronal del modelo de clasificación .....</b>	<b>57</b>
<b>Anexo 4: Parámetros de la red neuronal del modelo de generación.....</b>	<b>58</b>
<b>Anexo 5: Parámetros de la red neuronal del modelo de resumen.....</b>	<b>59</b>
<b>Anexo 6: Resultados del entrenamiento del modelo de clasificación .....</b>	<b>60</b>
<b>Anexo 7: Resultados del entrenamiento del modelo de generación .....</b>	<b>60</b>
<b>Anexo 8:Resultados del entrenamiento del modelo de resumen .....</b>	<b>60</b>
<b>Anexo 9: Aplicación web sección principal .....</b>	<b>61</b>
<b>Anexo 10: Aplicación web sección guía de uso .....</b>	<b>61</b>
<b>Anexo 11: Aplicación web sección servicios.....</b>	<b>62</b>
<b>Anexo 12:Aplicacion web sección redacción de noticias.....</b>	<b>62</b>
<b>Anexo 13: Aplicación web sección clasificar noticia .....</b>	<b>63</b>

# 1. Introducción

Actualmente el mundo digital nos ha motivado a realizar cambios en nuestras actividades cotidianas, una de ellas es la forma en que nos informamos. La digitalización ha provocado que la prensa escrita quede en el olvido dando paso a los medios digitales o la prensa digital, puesto que nos permite garantizar eficiencia y rapidez al momento de dar a conocer las noticias que surgen en cualquier parte del mundo. Si bien es cierto la prensa digital o los medios digitales son de gran utilidad por contar con un sin número de noticias generadas al día, pero esto a su vez podría ser un problema para el usuario, debido que al contar con un número considerable de noticias es poco probable que se pueda abordar en su totalidad, provocando el desinterés de estar siempre informados debido a la sobrecarga de información que tiene.

Por lo cual se cuenta con los conocimientos, que nos permite plantear una solución tecnológica, la cual será de gran ayuda para los lectores. Al digitalizar las cosas, la mayor parte de la población cuenta con un dispositivo inteligente que es una herramienta de comunicación y transmisión de información, por lo tanto, la idea principal de este trabajo es crear un sistema inteligente que permite identificar, clasificar y generar una síntesis de lo más relevante de una noticia de tal forma se pueda abordar más noticias en un menor tiempo, además se plantea generar recomendaciones en base a un tema de interés, con el propósito de sugerir noticias de carácter similar o relacionadas al tema previamente ingresado por el usuario, para ello se ha diseñado una aplicación web la cual permite la interacción entre usuario y sistema inteligente. Cabe mencionar que para el desarrollo de este trabajo se obtuvieron las noticias de los diferentes canales, prensa y medios digitales más relevantes del país.

## 2. Problema

En el presente capítulo se va a tratar de la importancia de este proyecto, y de qué manera surge el problema planteado anteriormente.

### 2.1 Antecedentes

Con el paso del tiempo la tecnología ha logrado grandes avances, uno de ellos en el área de la inteligencia artificial (IA) que se ha transformado en ser responsable de realizar tareas de procesamiento y generación de información, gracias a Machine Learning (ML) aplicada en el área de procesamiento de lenguaje natural (PLN), pretendiendo facilitar a los sistemas informáticos la capacidad de aprender (Beltrán & Rodríguez, 2021).

Es muy común escuchar sobre procesamiento del lenguaje natural puesto que es una derivación multidisciplinar de IA que se centra en entender, construir y generar tareas que estén relacionados al tratamiento de lenguaje y realice diferentes actividades como: procesamiento del lenguaje, análisis de sentimientos, sistemas de búsqueda de respuestas, traducción automática, generación de texto, clasificación de texto entre otras, con el propósito de desarrollar sistemas o aplicaciones capaces de generar contenido comprensible para el usuario final.

Implementar todas las tareas de PLN se puede presentar como un desafío, tal es el caso de las tareas de procesamiento complejo. Por ese motivo un grupo de investigación de inteligencia artificial EleutherAI ha considerado que GPT-J es la solución que facilita el análisis, desarrollo e implementación.

Generative Pre Trained Transformer (GPT-J) es un modelo de lenguaje autorregresivo de código abierto que ha sido desarrollado y entrenado para “generar texto” tomando líneas de entrada y prediciendo las palabras que continúan en base al contexto ingresado. (Forefront Team, 2021). GTP-J surgió en el 2021, con el objetivo de facilitar un sistema inteligente con enfoque general para PLN, obteniendo así un sistema de alto rendimiento y capaz de realizar las mismas tareas que el modelo GPT-3 Curie, con la única diferencia de que es código abierto. (Salinas, 2021).

La necesidad de crear una comunicación natural entre humano-máquina hace que GPT-J, pueda identificar temas, emociones, sentimientos, registros de chat a partir del lenguaje natural y generar resúmenes en segundos que permita brindar una mejor comprensión y satisfaga las necesidades de sus usuarios con respuestas no triviales (OpenAI, 2021).

Por otro lado, Según (Reuters Institute, 2020) informó que el 90% de la población optó por sitios web que proporcionen contenido informativo (noticias), no obstante, las redes sociales se han transformado en un medio estratégico para realizar la misma tarea. Estar informado de los acontecimientos políticos, económicos, educativos, deportivos, tecnológicos y de salud es un derecho que todas las personas tienen por ende surge la necesidad de buscar diferentes fuentes informativas. Hoy en día la tecnología y la internet son el boom de las cosas, desarrollando un sin número de plataformas y medios de comunicación digital.

## **2.2 Importancia y alcances:**

### **2.2.1 Justificación**

La evolución ha generado que los medios de comunicación impresos se conviertan en material obsoleto, dando paso a los medios de comunicación digital puesto que permite el intercambio de información por medio del internet, conectar diferentes tecnologías y técnicas de automatización que permiten generar contenido versátil. Existen diferentes estudios enfocados a mejorar los sistemas de clasificación, generación y recomendación de noticias que adopten nuevas técnicas para lograr una implementación eficiente; gracias a las investigaciones previas es posible crear un sistema inteligente con la capacidad de crear contenido en base a los intereses subjetivos.

Por otro lado, la revolución digital y la crisis sanitaria provocada por el COVID-19 ha incrementado el consumo del internet, según la INEC en 2020 se incrementó 11.5%, con una frecuencia de uso del 92.1% al día (INEC, 2021). Estos acontecimientos han provocado una migración hacia el uso del 100% de los medios digitales.

Por esta razón, se propone desarrollar un sistema inteligente para la generación, identificación, clasificación, redacción y recomendaciones de noticias utilizando algoritmos inteligentes y el modelo GPT-J, que será fundamental para mejorar la comunicación e interacción entre humano-sistema inteligente obteniendo como resultado un sistema que sea capaz sintetizar lo más relevante de una noticia con el propósito de que el usuario pueda abordar más noticias en un menor tiempo, además se propone desarrollar un sistema que sugiera diferentes noticias basadas en un tema de interés, donde todas las respuestas obtenidas sean enriquecedoras y cumplan con las expectativas del usuario.

### **2.2.2 Grupo Objetivo**

El sistema inteligente para la generación, identificación, clasificación, redacción y recomendación de noticias podrá ser utilizado para el público en general como también periodistas que tengan la intención de conocer los acontecimientos informativos del Ecuador, con la finalidad de fomentar la lectura para que los usuarios se puedan nutrir de nuevos conocimientos, de forma rápida y eficiente.

### **3. Objetivos generales y específicos**

En este capítulo detallaremos el objetivo general y los objetivos específicos de este proyecto.

#### **3.1 Objetivo general**

Desarrollar un sistema inteligente para la generación, identificación, clasificación, redacción y recomendaciones de noticias utilizando algoritmos inteligentes basados en clasificación y segmentación utilizando GPT-J

#### **3.2 Objetivos específicos**

- OE1: Revisar el estado del arte sobre Open IA y GPT-J para generar un modelo del lenguaje.
- OE2: Generar un repositorio de noticias del contexto local para entrenar al GPT-J.
- OE3: Analizar, diseñar, personalizar el contenido y respuesta del GPT-J basados en las noticias del contexto local.
- OE4: Diseñar e implementar un cliente que permita consumir los servicios del GPT-J.
- OE5: Generar una encuesta para validar y probar el funcionamiento del GPT-J

## **4. Revisión de la literatura**

En este capítulo analizaremos los diferentes conceptos, técnicas, y tecnologías utilizadas para el desarrollo e implementación del presente trabajo.

### **4.1 Procesamiento de Lenguaje Natural**

De acuerdo con lo que menciona (Kublik & Saboo, 2022) es “rama de la IA que se ocupa de construir computadoras que puedan comprender texto y palabras habladas al mismo tiempo, combina la lingüística computacional con modelos estadísticos de aprendizaje automático y aprendizaje profundo”.

#### **4.1.1 Generación de Texto**

Según (Karakaya, 2021) “la generación de texto es un subcampo del procesamiento del lenguaje natural (NLP). Aprovecha los conocimientos en lingüística computacional e inteligencia artificial para generar automáticamente textos en lenguaje natural, que pueden satisfacer ciertos requisitos comunicativos”.

#### **4.1.2 Clasificación de Texto**

Trata de asignar etiquetas a un texto dependiendo del contenido y la semántica tomando como base textos existentes los cuales permiten categorizar la información.

### **4.2 Machine Learning (ML)**

Según (Mahesh, 2018) establece que machine Learning es “la ciencia que hace que los ordenadores aprendan a partir de los datos”, es decir que un ordenador pueda aprender por sí mismo, sin estar programado paso a paso. Machine Learning cuenta con diferentes algoritmos de base para resolver problemas de datos, emplear algún tipo de algoritmo dependerá del problema a resolver.

### **4.3 Deep Learning**

Es una técnica de aprendizaje automático basado en redes neuronales, capaz de simular acciones del cerebro humano a través de capas de neuronas artificiales, estas capas se comprenden de forma jerárquica con el propósito de comprender los datos de capa. Deep Learning tiene como objetivo la extracción automática de las características más importantes de los datos, de tal forma que el modelo pueda convertir esa característica en una salida esperada (Pérez Borrero & Gegúndez Arias, 2021).

### **4.4 Redes Neuronales**

Según (Pino Díez, Gómez Gómez, & Abajo Martínez, 2001) una red neuronal puede definirse como “Un grupo de neuronas simuladas, que están muy interconectadas, al igual que las neuronas del cerebro, y que son capaces de aprender de la misma forma que los hacen las personas”. Las redes neuronales son modelos utilizados en Machine Learning debido a que puede realizar tareas similares a las de un humano.

### **4.5 Sistemas Recomendadores**

Los sistemas de recomendación (SR) son parte de la inteligencia artificial, se considera como una aplicación que sugiere ítems, productos o información que satisfagan los intereses del usuario a través de técnicas de Machine Learning y Deep Learning (Tamayo, 2012).

#### **4.5.1 Basado en Contenido**

Se recomienda en base a los temas de interés que se han seleccionado en el pasado. Proporciona resultados que tengan similitud con otros ítems que sean de interés del usuario.

#### **4.5.2 Filtrado Colaborativo**

Se recomienda desde un enfoque grupal, toma como referencia la puntuación de los elementos de un grupo de usuarios y busca la similitud de la calificación para determinar qué se debe recomendar.

## **4.6 Generative Pretrained Transformer JAX (GPT-J)**

JAX-bases (Mesh) Transformer LM (GPT-J) es una versión del modelo GPT-3 de OpenAI. GPT-J es un modelo orientado a realizar tareas de procesamiento de lenguaje, fue desarrollado por un grupo de investigadores de EleutherAI, Ben Wang y Aran Komatsuzaki. GPT-J es un modelo de generación de texto autorregresivo de 6000 millones de parámetros (Komatsuzaki, 2021).

JAX es una biblioteca de python utilizada para realizar tareas de aprendizaje automático tales como: generación de código, chatbot, escritura de cuentos, traducción de idiomas, búsqueda de información entre otras.

## **4.7 Generative Pretrained Transformer 3 (GPT-3)**

Es la tercera versión del modelo GPT de OpenAI, se centra en predecir el texto que viene a continuación en función del texto de entrada. Su enfoque es general por ello puede realizar cualquier tarea de carácter lingüístico, el acceso es a través de la API de OpenAI (Tingiris, 2021).

## **4.8 Recolección de Datos**

Consiste en recopilar información de diferentes fuentes con el propósito de obtener un perspectiva completa y precisa del tema de interés.

### **4.8.1 Data Scraping**

Técnica que permite extraer datos de fuentes no estructuradas como, sitios web y aplicaciones, con la finalidad de limpiar, estructurar y procesar para que sean utilizables por los sistemas de aprendizaje automático

### **4.8.2 Data Crawler**

Técnica utilizada para leer, copiar y almacenar el contenido de los sitios web con fines de archivo o indexación. Su objetivo principal es elaborar una base de

datos con toda la información extraída de los diferentes hipervínculos del sitio web de interés.

#### **4.8.3 Repositorio de Datos**

Colección central de datos y datasets, necesaria para alimentar, entrenar y testear la efectividad de nuestro modelo.

### **4.9 RSS**

Really Simple Syndication (RSS) se define como una herramienta de recolección de datos en sitios, web, noticieros en línea, publicaciones, etc. RSS obtiene la información a través del código XML, detectando la información actualizada y envía la información a los suscriptores. Se utiliza para guardar y conservar los datos actualizados (Singh & Sahu, 2015).

### **4.10 Cliente Web**

El desarrollo web cambia de manera rápida y constante, a tal punto que en la actualidad el diseño de una página web no solo prioriza el diseño de visualización, sino también el rendimiento de velocidad para presentar la información solicitada, es por ello que se adoptan técnicas de aplicación de página única.

#### **4.10.1 Aplicación de página única (SPA)**

Aplicación web en donde todas las interfaces solicitadas se muestran en una página según lo solicitado y sin necesidad de recargar el navegador. Tiene como propósito generar una experiencia de usuario más rápida.

#### **4.10.2 Interfaz de programación de aplicaciones (API)**

Interfaz de comunicación entre aplicaciones, permite la comunicación entre productos y servicios sin necesidad de conocer la estructura de implementación. Tiene como propósito de integrar sistemas y facilitar el intercambio de datos entre diferentes softwares.

#### 4.11 Framework

Es un ambiente de trabajo predefinido que se utiliza como base para la creación de aplicaciones, el cual cuenta con diferentes herramientas y funcionalidad que ayudan a minimizar el riesgo de cometer errores en el proceso de desarrollo, obteniendo como resultado una aplicación de alta calidad.

#### 4.12 React.js

Es una librería de JavaScript de código abierto diseñada para crear interfaces de usuario interactivas. Su enfoque principal es facilitar el desarrollo de SPA, como también se encarga de actualizar y renderizar los componentes (Berbel, 2018).

#### 4.13 Arquitectura en capas

La arquitectura en capas tiene como objetivo distribuir en partes la aplicación (aplicación web, aplicación móvil) con el propósito de que cada capa cumpla un rol específico. Se detalla cada una de las capas a continuación.

- **Capa de presentación:** esta capa representa la interfaz gráfica de la aplicación, es decir lo que el usuario ve e interactúa. Mantiene una comunicación directa con la capa de negocio.
- **Capa de negocio:** la capa de negocio contiene todo el funcionamiento del sistema, se encarga de recibir y enviar la información para posteriormente procesarla. Se conecta con la capa inferior y superior.
- **Capa de datos:** administra los datos sobre los que se está operando, esta capa recibe peticiones que permiten almacenar y recuperar la información por parte de la capa de negocios.

## 4.14 Metodología Scrum

Scrum es un proceso que sirve para gestionar proyectos mediante un conjunto de tareas regulares y de manera colaborativa. Esta metodología es una de las más utilizada en el ámbito de desarrollo de software por su gestión y desarrollo ágil, para lograr su objetivo incluye reuniones y herramientas que de forma colaborativa que ayudan a los desarrolladores a estructurar y organizar de mejor manera las diversas fases de desarrollo (Gallego, 2012).

La metodología Scrum esta con compuesta de diferentes fases que a continuación se menciona.

- **El quién y qué:** determina los roles y las actividades para cada persona del grupo.
- **El dónde y cuándo:** representa el sprint
- **El porqué y el como:** herramienta utilizadas por el equipo SCRUM

**Roles Scrum quien y que:** actores que cumplen tareas específicas en el desarrollo del proyecto

- **Propietario del producto (Product Owner):** responsable de gestionar las actividades en el backlog, denominado como “voz del cliente”.
- **Líder (Scrum Master):** responsable del equipo de trabajo, su objetivo es garantizar el cumplimiento de los eventos acorde a los planificado.
- **Miembros del Equipo de Desarrollo (Development team Members):** miembros del grupo de trabajo que se encargan de desarrollar los sprints.
- 

**El sprint: dónde y cuándo:** contenedor para los eventos, tiene una duración fija de un mes o menos para crear congruencia.

- **Planificación de Sprint (Sprint Planning):** asignación de tareas, tiempos y métodos que se debe cumplir el equipo de trabajo.
- **Scrum diario (Daily Scrum):** evento de 15 min por día que verifica el progreso hacia una meta de sprint.

- **Refinamiento del Backlog (Backlog Refinement):** el propietario del producto revisa cada elemento en el Product Backlog para aclarar cualquier duda que quede en el equipo.
- **Revisión del Sprint (Sprint Review):** revisa los resultados de cada sprint e identifica diferentes medidas de adaptación para el futuro.
- **Retrospectiva (Sprint Retrospective):** planificación de para elevar la calidad y la eficiencia del producto.

**Herramientas:** diseñado para mejorar la transparencia de la información,

- **Backlog de Producto:** facilita una breve informacion sobre el proyecto y, en ocasiones, esta información no se considera una aplicación oficial.
- **Historias de usuarios:** historias que ofrecen información del comportamiento de las solicitudes.
- **Backlog de Sprint:** el conjunto de elementos que priorizan, miden y aceptan durante la planificación de Sprint.
- **El panel de tareas:** panel de asignacion de tareas.
- **Definición de listo:** cada equipo competente y ágil tiene ciertos acuerdos que deben realizarse antes de que se complete el proyecto. Entre estos acuerdos, se han completado todas las tareas y se han realizado revisiones y pruebas de código.

## 5. Marco metodológico

En esta sección vamos a detallar la propuesta de solución para este proyecto.

### 5.1 Propuesta de solución

En el siguiente gráfico se presenta un modelo de la estructura del sistema inteligente y a continuación se define cada fase de desarrollo del proyecto.

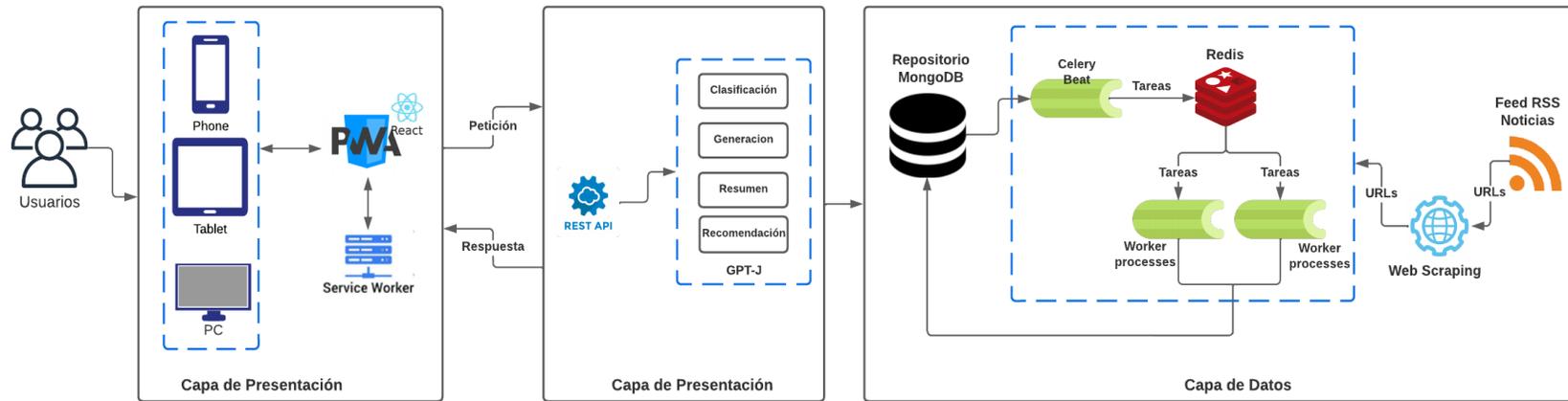


Figure 1: Propuesta de solución

El proyecto va a constar de 6 fases, la **primera fase** es estudiar los diferentes métodos de recolección de datos, fundamentos de procesamiento de lenguaje natural, algoritmos de Machine Learning, Deep Learning para la clasificación, identificación, generación de texto, configuración del ambiente de desarrollo y finalmente estudiar los fundamentos para el desarrollo del sistema inteligente utilizando React.js, con el objetivo de tener un conocimiento solido de lo que debemos aplicar en cada fase de desarrollo.

La **segunda fase** del proyecto consiste en recolectar los datos (noticias), en esta etapa lo que se busca es capturas las ultimas noticias de 46 sitios web los cuales serán la fuente de información. El proceso de recolección de datos está compuesto de siete servicios. A continuación, se presenta una figura que ilustra los pasos a realizar para la recopilación de datos.

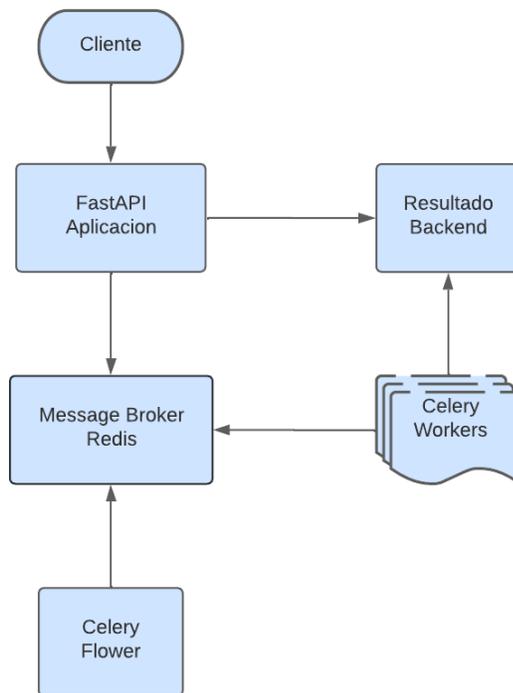


Figure 2: Diagrama de flujo para la recolección de datos

En primera instancia se utiliza RSS Feed proveniente de cada diario digital para obtener los links asociados a cada noticias, mediante una petición al analizador de fuentes RSS (*graphql-rss-parser*) el cual consta de 3 aplicaciones nos devuelve como respuesta un JSON de la fuente a analizar, posteriormente pasa a una cola de tareas asíncronas (*Celery Worker*) en donde se extrae la información a través del metadata, en caso de que el metadata no contenga información en específico por ejemplo la categoría, se procede a realizar una extracción de información en el componente *yoast -schema-graph*, este componente está diseñado para obtener la mejor información sobre el sitio web por ende es enriquecedor en cuanto a información específica pero cabe recalcar que puede existir la posibilidad de que no se encuentra la información ni en el metadata , ni el componente por tal motivo se realiza una búsqueda exhaustiva a través etiqueta y la posición de la misma, con el propósito de obtener la información específica.

Después de realizar la búsqueda y la adquisición de la información se utiliza la librería *Yake* para la extracción de palabras clave en base al contenido de la noticia, se realiza este proceso para obtener una información completa la cual será de gran utilidad al momento de implementar la fase de diseño y entrenamiento del modelo. Una vez realizado el proceso de adquisición de palabras clave se procede a almacenar la información en una base de datos.

La **tercera fase** del proyecto es crear un repositorio, se considera de suma importancia debido a que nos permite centralizar toda la información recolectada y procesada dentro de un repositorio. Una vez almacenada la información en la base de datos se utiliza MongoDB Atlas, para la administración de los datos en la nube proporcionados una API con la cual se puede realizar una petición GET para obtener las noticias.

La **cuarta fase** del proyecto tiene como objetivo configurar el ambiente de desarrollo en base a los requisitos previos para implementar el modelo GPT-J. El hardware más común es un T4, V100 o TPU. En este caso se implementará en una TPU de Google Cloud se tomará como base la guía de configuración de Mesh Transformer JAX, donde establece los conceptos básicos para utilizar el modelo, como también nos proporciona los pesos pre entrenados para ajustar el modelo en base a nuestros requerimientos (Wang & Ben, 2021).

La **quinta fase** del proyecto es diseñar los modelos de clasificación, generación y recomendación para lo cual se tomará como base fundamental el modelo GPT-J dado que permite comprender diferentes dominios y se acopla a los requerimientos del sistema. La creación del modelo dependerá del caso de uso, como formato general para implementar GPT-J se debe contar con un conjunto de ejemplos formateados en un archivo.txt el cual contiene un ejemplo de entrada y su salida específica, a más de la entrada y salida se debe usar un separador al final de cada solicitud y la finalización en el conjunto de datos de tal forma el modelo pueda interpretar donde empieza y donde termina el ejemplo, GPT-J utiliza para todos los casos el separador `<|endoftext|`, cabe recalcar que el mensaje junto con la finalización y el separador no puede exceder de los 2048 tokens (Forefront, 2021).

A continuación, se detalla los diferentes modelos y parámetros que cada uno requiere para el desarrollo del sistema inteligente.

- **Modelo Clasificación:** este modelo consiste en clasificar el texto, cada entrada se clasifica en una clase previamente definida. En este caso se estableció 16 clases o categorías, el modelo se entrenó con 21126 artículos y se validó con 2348 artículos durante 10 épocas, los diferentes parámetros de configuración del modelo se adjuntan en la sección anexos. Cabe destacar que para entrenar el modelo se debe cumplir con el formato pre establecido, a continuación, se presenta una figura que permite ilustrar el formato:

```

<|endoftext|>

    Titulo:

    Contexto:

    Categoria:

<|endoftext|>

```

Figure 3: Formato del archivo.txt para entrenar el modelo de clasificación

A continuación, se presenta una figura que ilustra de mejor manera el comportamiento del modelo de clasificación.

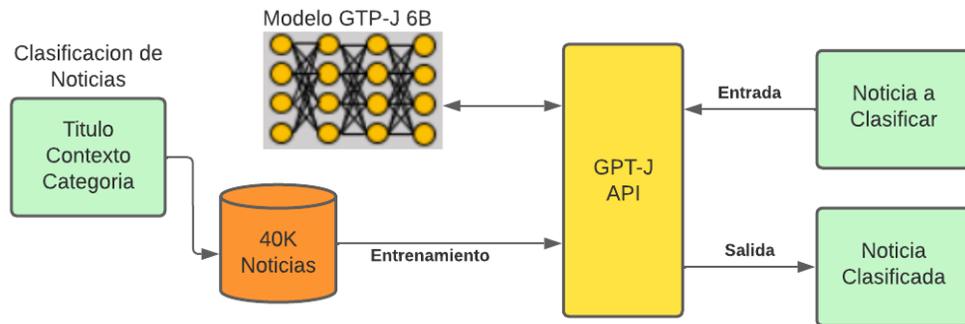


Figure 4: Diagrama de flujo del modelo GPT-J Clasificación

- **Modelo Generación:** el objetivo es crear un modelo capaz de predecir la palabra o secuencia de palabras dado un texto de entrada. A continuación, se presenta una figura que ilustra el proceso de generación de texto

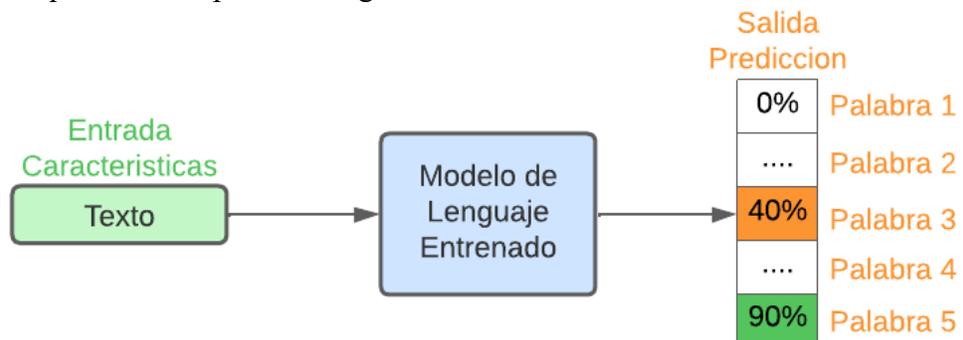


Figure 5: Generación de Texto

Como se puede apreciar en la figura anterior, se toma un texto de entrada, el modelo procesa esta entrada en base a la distribución de probabilidad sobre el conjunto de palabras que conoce, de tal forma que elige la palabra que continua en base al texto de entrada y al conjunto de datos con el cual se entrenó el modelo.

El artículo titulado “THE CURIOUS CASE OF NEURAL TEXT DeGENERATION” establece que utilizar *Nucleus Samplig* en la generación de texto es una buena solución para evitar incoherencias y palabras repetidas, Su principal objetivo es utilizar la distribución de probabilidad para determinar el conjunto de palabras del cual se tomara la muestra. El muestreo Top-K toma las muestras de las distribuciones neuronales en cada paso de tiempo se expone los top-k de probabilidad más relevante. Dada la distribución  $P(x|x_{1:i-1})$  se define el vocabulario top-k  $V^{(k)} \subset V$  como el conjunto de tamaño  $k$  que maximiza  $\sum_{x \in V^{(k)}} P(x|x_{1:i-1})$ . Pero  $p' = \sum_{x \in V^{(k)}} P(x|x_{1:i-1})$ . Cabe mencionar que declarar el valor constante de  $k$  es 5 (Holtzman, Buys, Du, Forbes, & Choi, 2020).

Tomando como base los mencionado anteriormente, en nuestro caso para generar una noticia se toma como texto de entrada el título y las palabras clave, dado que, al proporcionar más información al modelo, este generara un mejor resultado. El modelo se entrenó con 30641 artículos y se validó con 7661 artículos durante 10 épocas, a más de configurar las épocas se configuro diferentes parámetros de la red los cuales se detalla en la sección de anexos. De la misma forma para entrenar el modelo se debe contar con un formato específico, el cual se especifica a continuación.

```
<|endoftext|>  
  
Titulo:  
  
Palabras_Clave:  
  
Contexto:  
  
<|endoftext|>
```

Figure 6: Formato del archivo.txt para entrenar el modelo de generación de texto

A continuación, se presenta una figura que ilustra de mejor manera el comportamiento del modelo de generación de noticias

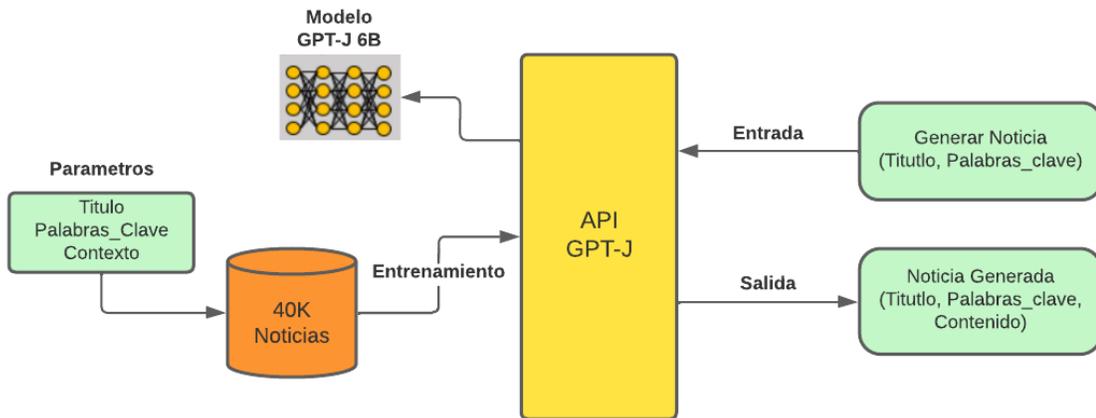


Figure 7: Diagrama de flujo modelo GPT-J Generación

- Modelo Recomendación:** El desarrollo de este modelo se comprende en dos procesos, en primera instancia se realiza el proceso de similitud semántica, consiste en determinar si dos textos son similares, para ello toma un texto de entrada y lo convierte en un vector, de igual forma convierte en vectores los textos a comparar, el objetivo es buscar las mejores respuestas de un grupo de datos dado una consulta (Muennighoff, 2022). A continuación, se presenta una figura que ilustra proceso detallado.

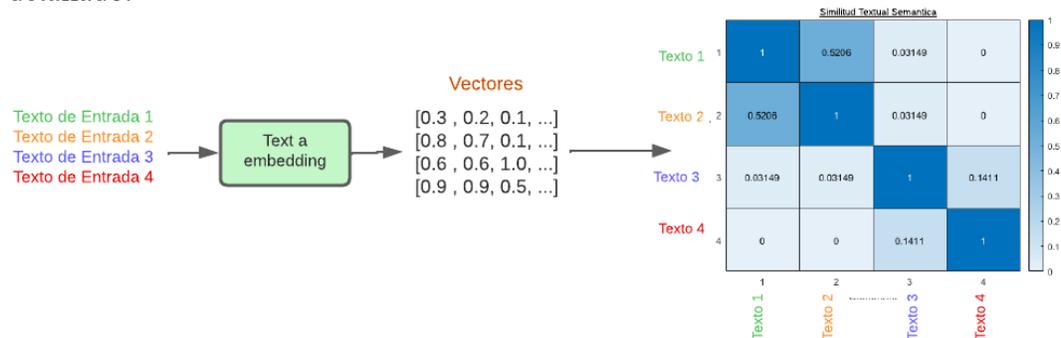


Figure 8: Similitud semántica

Una vez obtenida la noticia con mayor similitud se realiza el segundo proceso, el cual consiste en realizar un resumen del contenido de la noticia, previa a generar el resumen con el modelo GPT-J, se realizó el proceso de generar una nueva data.

Para crear la nueva data se utilizó un modelo pre entrenado de generación de resumen, puesto que GPT-J solo aprende en base a los datos proporcionados. Por tal razón se realizó el resumen con los datos del repositorio, para el entrenamiento se utilizó título y contenido como parámetros de entrada. Se presenta una figura que permite abstraer de mejor manera los conocimientos previamente mencionado.

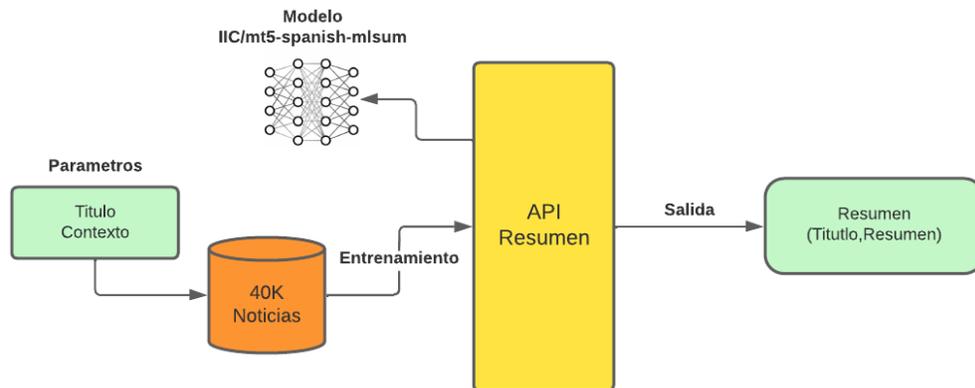


Figure 9: Diagrama de flujo para la generación de resumen

Una vez entrenado los datos con el modelo “IIC/mt5-spanish-mlsum” se creó una nueva data con el formato pre establecido por el modelo GPT-J

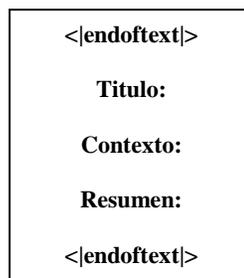


Figure 10: Formato del archivo.txt para entrenar el modelo de resumir

Después de obtener la data con el formato requerido, se procede a entrenar con el modelo GPT-J, en nuestro caso entrenamos con 30000 artículos y se validó con 3000 artículos durante 10 épocas, a más de configurar las épocas se configuró diferentes parámetros los cuales se detalla en la sección de anexos.

La siguiente figura representa el proceso de generación de resumen con el modelo GPT-J

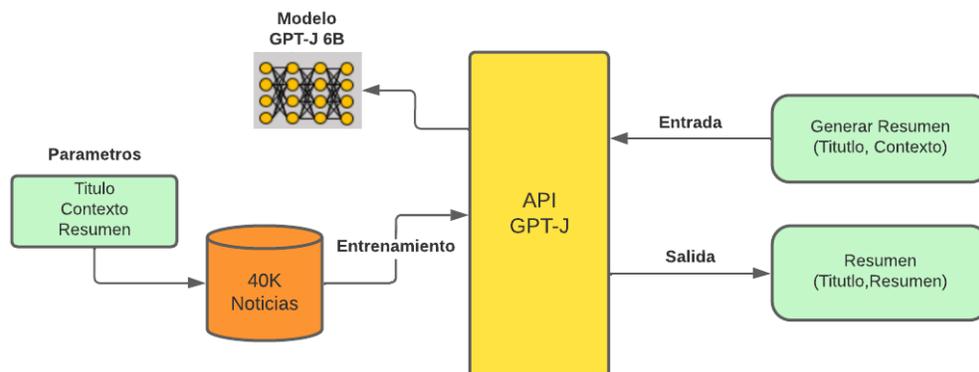


Figure 11: Diagrama de flujo para la generación del resumen

Finalmente, se obtiene como resultado una noticia similar al tema de interés del usuario y un resumen del contenido.

La **sexta fase** del proyecto trata del diseño y desarrollo la aplicación web para lo cual se utiliza la librería React.js, la cual permite interar a los usuarios con el sistema inteligente. Dentro de esta fase también se realiza la conexión con los diferentes modulos generados en la fase anterior, por ello se utiliza una API que se comporta como un intermediario entre la aplicación web y los servicios (modelos) que están alojadas en Google Cloud. A continuación, se presenta una ilustración que permite comprender de mejor forma la conexión entre la aplicación y los servicios

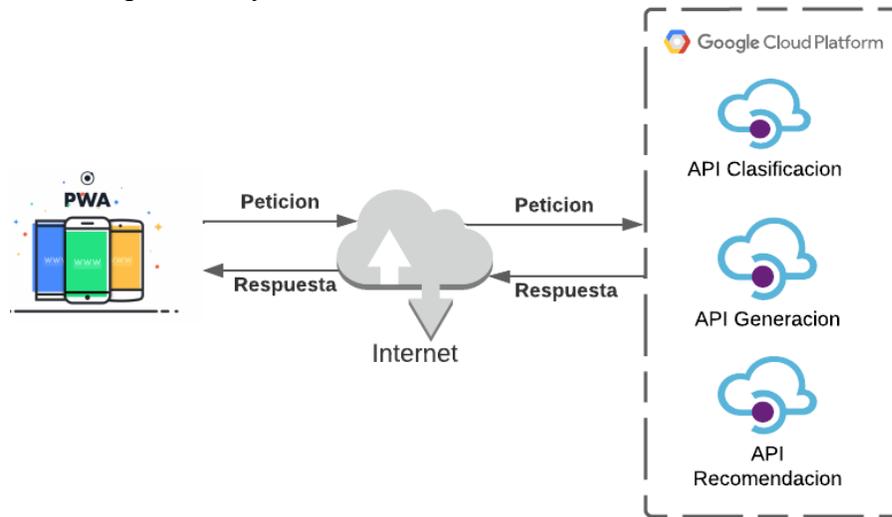


Figure 12: Conexión entre la aplicación y los servicios

Finalmente, como **séptima fase** se realizará una evaluación del funcionamiento con el objetivo de medir el grado de aceptación. Se desarrollará una encuesta que permita validar si el sistema funciona correctamente, esta encuesta se realiza a 30 personas (público en general y periodistas) con el fin de tener una retroalimentación si cumple con sus expectativas o requiere un análisis para posibles mejoras.

## 5.2 Metodología Scrum

La metodología que se va a utilizar en el desarrollo del sistema inteligente es Scrum, en este caso el Product Owner – Stakeholder son las personas encargadas de enumerar los requisitos del sistema inteligente de tal forma se pueda conformar los sprints, el Scrum Master cumple el papel de líder del proyecto, el cual se encarga de asignar las tareas al Scrum Team, como también de realizar reuniones periódicas para conocer los avances e inquietudes que surge durante el desarrollo, se establece que cada sprint tendrá una duración máxima de dos a tres semanas dependiendo la complejidad de la actividad a realizar. Cabe mencionar que en la sección de cronograma de actividades se detallara a todos los sprints planteados para alcanzar los objetivos de manera rápida y eficiente.

### Roles Scrum:

**Product Owner:** Ing. Diego Quisi P (RH)

**Scrum Máster:** Daniel Dionicio Guzmán Chuva (DG)

### Scrum Team:

- **Desarrollador N°1:** Daniel Dionicio Guzmán Chuva (DG)
- **Desarrollador N°2:** Diana Fernanda Tixi Uyaguari. (DT)

**Stakeholders:** Usuarios que realizan la encuesta.

## 5.3 Elaboración de la lista de sprints

Para la correcta elaboración de la lista de sprints debemos tener una lista base de actividades que se deben realizar para el proyecto, a continuación, se muestra las actividades que se deben realizar en base a los objetivos específicos que ya fueron previamente establecidos.

- **Lista de actividades:**

**OE1:** Revisar el estado del arte sobre Open IA y GPT-J para generar un modelo del lenguaje

No.	Actividad
1	Realizar el estado del arte de Open IA, GPT-J y tecnologías relacionadas.
2	Estudiar la arquitectura y usos de GPT-J.
3	Identificar los modelos y técnicas más efectivas para la implementación y ajustes de GPT-J.
4	Realizar una documentación de la información relevante para la implementación de GPT-J.

Tabla 1: Objetivo Especifico 1

**OE2:** Generar un repositorio de noticias del contexto local para entrenar al GPTJ.

No.	Actividad
1	Recolectar noticias de ámbito nacional haciendo uso de la técnica Data Scraping y RSS para abordar los diferentes medios de comunicación digital de carácter informativo.
2	Limpiar y estructurar las noticias recolectadas.
3	Generar repositorio de noticias.

*Tabla 2: Objetivo Especifico 2*

**OE3:** Analizar, diseñar, personalizar el contenido y respuesta del GPT-J basados en las noticias del contexto local.

No.	Actividad
1	Realizar el estado del arte de Open IA, GPT-J y tecnologías relacionadas.
2	Estudiar la arquitectura y usos de GPT-J.
3	Identificar los modelos y técnicas más efectivas para la implementación y ajustes de GPT-J.
4	Realizar una documentación de la información relevante para la implementación de GPT-J.

*Tabla 3: Objetivo Especifico 3*

**OE4:** Diseñar e implementar un cliente que permita consumir los servicios del GPT-J.

No.	Actividad
1	Definir requerimientos para una PWA.
2	Desarrollar los requerimientos para el desarrollo de la PWA.
3	Generar los servicios REST para el consumo de GPT-J.
4	Diseño y generación de pruebas unitarias.
5	Implementar una de la PWA en base a los requerimientos definidos.
6	Realizar pruebas de funcionalidad y rendimiento.
7	Realizar documentación sobre la funcionalidad y uso de la PWA.
8	Desplegar el sistema en la nube.

*Tabla 4: Objetivo Especifico 4*

**OE5:** Generar una encuesta para validar y probar el funcionamiento del GPT-J

<b>No.</b>	<b>Actividad</b>
1	Generar encuestas para probar y validar el correcto funcionamiento de GPT-J.
2	Analizar la retroalimentación obtenida de los usuarios.
3	Realizar el registro de los resultados obtenidos.

*Tabla 5: Objetivo Especifico 5*

Tomando como base los objetivos específicos se planea desarrollar 4 sprints, los cuales están conformados por diferentes actividades que se encuentran listadas en los objetivos. Al concluir cada sprint se realiza un análisis para verificar su cumplimiento con el fin de conocer los avances y dificultades que permitan mejorar el producto funcional, dando paso al nuevo sprint creando nuevas funcionalidades con el desarrollo de los requerimientos. A continuación, se detalla cada uno de los sprints.

<b>Sprint: N°1</b>
<b>Objetivo específico N°1</b>
<b>Actividad N°1:</b> Realizar el estado del arte de Open IA, GPT-J y tecnologías relacionadas.
<b>Actividad N°2:</b> Estudiar la arquitectura y usos de GPT-J.
<b>Actividad N°3:</b> Identificar los modelos y técnicas más efectivas para la implementación y ajustes de GPT-J.
<b>Actividad N°4:</b> Realizar una documentación de la información relevante para la implementación de GPT-J.
<b>Objetivo específico N°2</b>
<b>Actividad N°1:</b> Recolectar noticias de ámbito nacional haciendo uso de la técnica Data Scraping y RSS para abordar los diferentes medios de comunicación digital de carácter informativo.
<b>Actividad N°2:</b> Limpiar y estructurar las noticias recolectadas.
<b>Actividad N°3:</b> Generar repositorio de noticias.

*Tabla 6: Actividades a realizar en el sprint 1*

<b>Sprint: N°2</b>
<b>Objetivo específico N°3</b>
<b>Actividad N°1:</b> Analizar y definir los mejores parámetros para ajustar el modelo a nuestro problema.
<b>Actividad N°2:</b> Adaptar el modelo de GPT-J para nuestro problema.
<b>Actividad N°3:</b> Entrenar el modelo GPT-J en base al repositorio de noticias.
<b>Actividad N°4:</b> Realizar pruebas de funcionalidad y rendimiento

*Tabla 7: Actividades a realizar en el sprint 2*

<b>Sprint: N°3</b>
<b>Objetivo específico N°4</b>
<b>Actividad N°1:</b> Definir requerimientos para una PWA.
<b>Actividad N°2:</b> Desarrollar los requerimientos para el desarrollo de la PWA
<b>Actividad N°3:</b> Generar los servicios REST para el consumo de GPT-J.
<b>Actividad N°4:</b> Diseño y generación de pruebas unitarias
<b>Actividad N°5:</b> Implementar una de la PWA en base a los requerimientos definidos.
<b>Actividad N°6:</b> Realizar pruebas de funcionalidad y rendimiento.
<b>Actividad N°7:</b> Realizar documentación sobre la funcionalidad y uso de la PWA.
<b>Actividad N°8:</b> Desplegar el sistema en la nube.

*Tabla 8: Actividades a realizar en el sprint 3*

<b>Sprint: N°4</b>
<b>Objetivo específico N°5</b>
<b>Actividad N°1:</b> Generar encuestas para probar y validar el correcto funcionamiento de GPT-J
<b>Actividad N°2:</b> Aplicar y analizar la retroalimentación obtenida de los usuarios.
<b>Actividad N°3:</b> Realizar el registro de los resultados obtenidos.

*Tabla 9: Actividades a realizar en el sprint 4*

## 6. Resultados

En esta sección se detalla a profundidad el proceso de desarrollo del sistema inteligente.

### 6.1 Especificación de requerimientos

#### 6.1.1 Requerimientos funcionales

Los requerimientos o requisitos funcionales son aquellas características explícitas que debe cumplir un proyecto de software. Cada requerimiento está relacionado con el comportamiento y la función que debe cumplir el sistema.

Seguidamente, se detalla los requerimientos del proyecto:

<b>Identificador: SI-01</b>	<b>Nombre: Clasificar Noticia</b>
<b>Descripción:</b> El sistema tiene que ser capaz de clasificar la noticia	
<b>Entrada:</b> Título de la noticia Contexto de la noticia	
<b>Proceso:</b> El usuario ingresa el título y contexto de la noticia, la aplicación llama a la API de Clasificación, procesa la información ingresada y finalmente asigna una categoría.	
<b>Salida:</b> La aplicación mostrara la categoría a la que pertenece la noticia ingresada	
<b>Prioridad:</b> Alta	

<b>Identificador: SI-02</b>	<b>Nombre: Generar Noticia</b>
<b>Descripción:</b> El sistema va a ser capaz de generar una noticia en base al título y palabras clave	
<b>Entrada:</b> Título de la noticia Palabras Clave	
<b>Proceso:</b> El usuario ingresa el título, la aplicación llama a la API Generación, procesa la información ingresada	
<b>Salida:</b> Redacta una noticia en base al título y las palabras clave	
<b>Prioridad:</b> Alta	

<b>Identificador:</b> SI-03	<b>Nombre:</b> Recomendar Noticia
<b>Descripción:</b> El sistema tiene que ser capaz recomendar una noticia que sea del interés del usuario	
<b>Entrada:</b> Título de la noticia	
<b>Proceso:</b> El usuario ingresa el posible título de la noticia, la aplicación llama a la API de Recomendación, el sistema procesa la información ingresada y finalmente obtenemos la noticia recomendado por el sistema	
<b>Salida:</b> Título de la noticia recomendado, un resumen del contexto de la noticia y la fuente para la noticia para mayor información	
<b>Prioridad:</b> Alta	

### 6.1.2 Requerimientos de interfaz

En esta sección se presenta el prototipo de la aplicación web. Se utilizó la herramienta proto.io para el desarrollo de las interfaces. A continuación, se presenta los prototipos de la aplicación web.

**Sección Principal de la Aplicación Web**

En esta imagen se puede visualizar la sección principal de nuestro sitio web de noticias, el cual cuenta con tres secciones:

***Sobre nosotros -- ¿Cómo Usar? -- Servicios***

Además cuenta con un botón de “EMPEZAR” el cual nos redirección a la sección de servicios

Tabla 10: Prototipo sección principal





Tabla 12: Prototipo servicio de recomendar



Tabla 13: Prototipo servicio de generación de noticias

**Sección Servicios - Clasificación**



**Noticias  
GPT-J**

Sobre  
Nosotros

¿Como Usar?

Servicios

**UPS**

Recomendaciones

Generación

Clasificación

**TITULAR**

**CUERPO DE LA NOTICIA**

GENERAR

○

**Clasificación de Noticia**

Lorem ipsum dolor sit amet, consetetur sadipscid Lorem ipsum dolor sit amet, consetetur sadipscing elit, sed diam nonu eirmod tempor invidut labore et dolore magna aliqu yam erat, seg elit, sed diam nonu eirmod tempor invid>Lorem ipsum dolor sit amet, consetetur sadipscing elit, sed diam nonu eirmod tempor invidut labore

**Categoría**

Categoría

**Título Noticia**

*Tabla 14: Prototipo servicio de clasificación de noticias*

## 6.2 Diseño detallado

En esta sección se trata de realizar el modelo del sistema con el objetivo de ilustrar el proceso de implementación del sistema inteligente.

### 6.2.1 Diagrama de actividades

El objetivo de desarrollar un diagrama de actividades es demostrar, describir, ilustrar la lógica, función y operación de un sistema. A continuación, se presenta diferentes diagramas que ilustran el proceso de desarrollo.

En la figura 13 podemos apreciar el proceso que se realiza para generar una recomendación

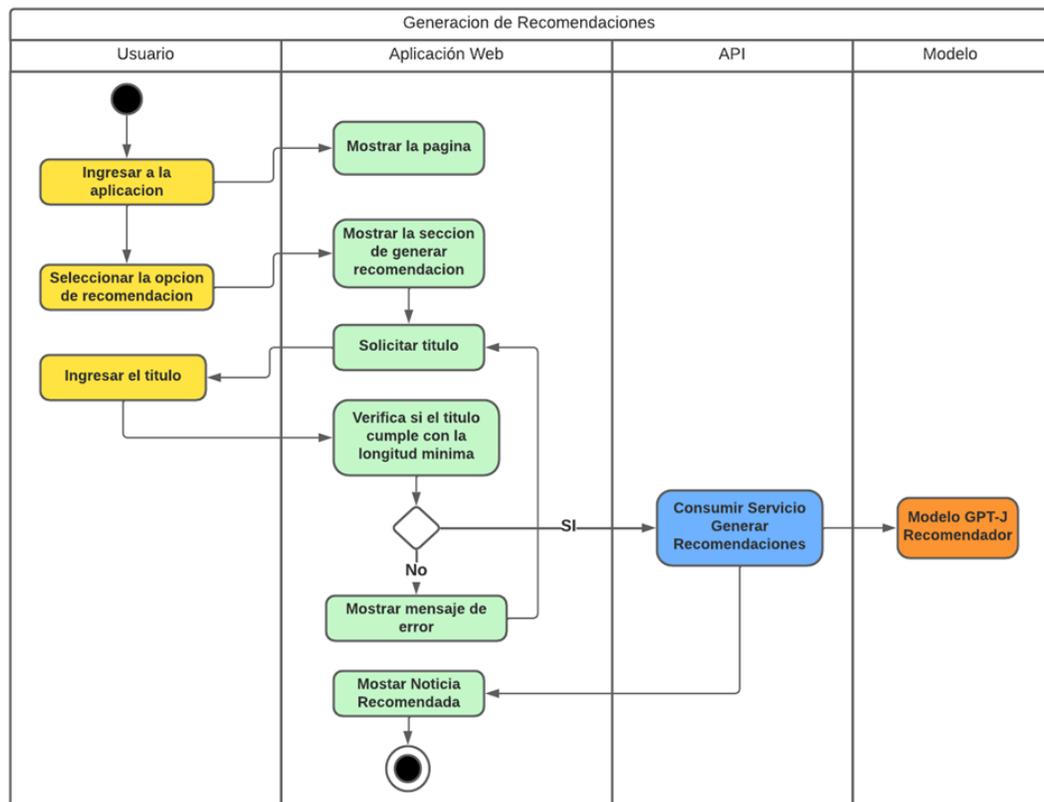


Figure 13::Diagrama de actividades generar recomendación de noticias

En la figura 14 podemos apreciar el proceso que se realiza para generar una noticia basado en el título y palabras clave

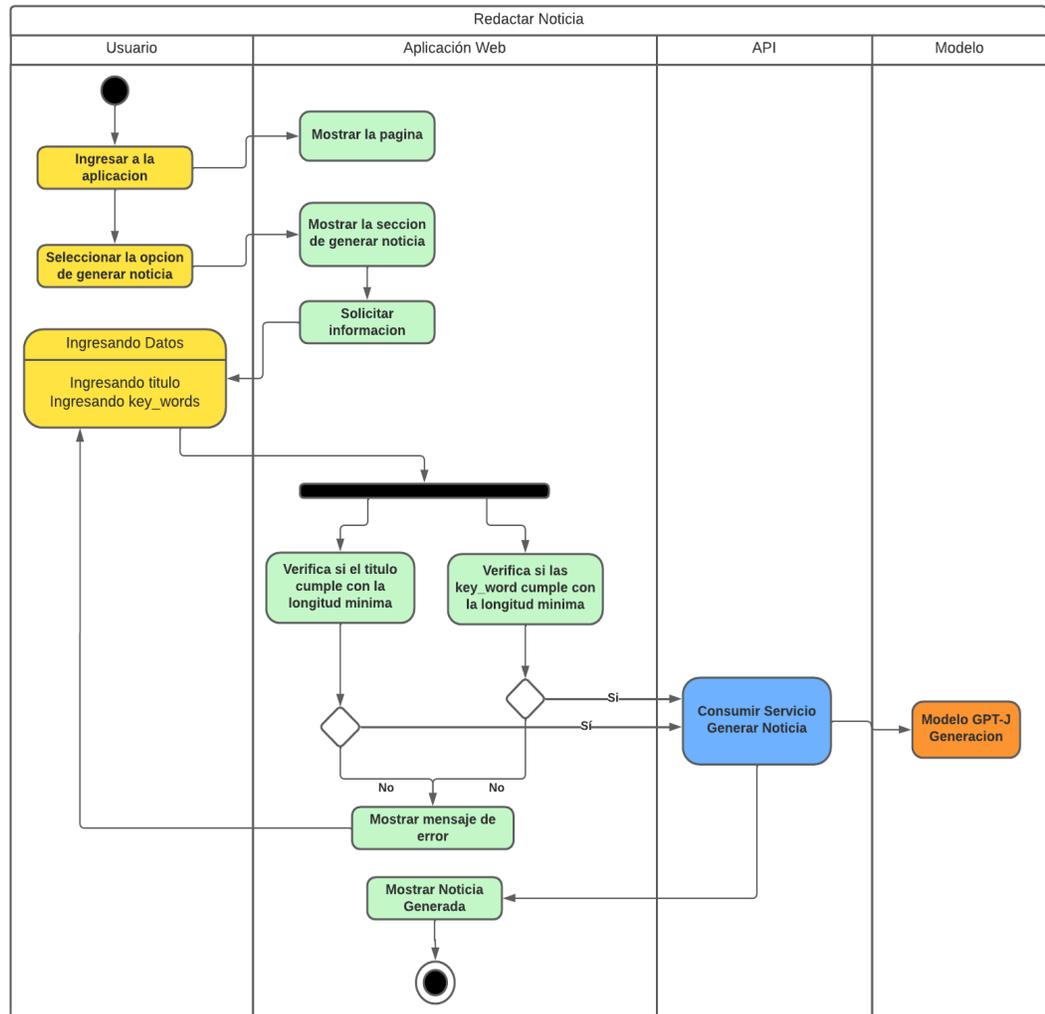


Figure 14: Diagrama de actividades generar noticias

En la figura 15 podemos apreciar el proceso que se realiza para clasificar una noticia tomando como base el título y el contenido de la noticia

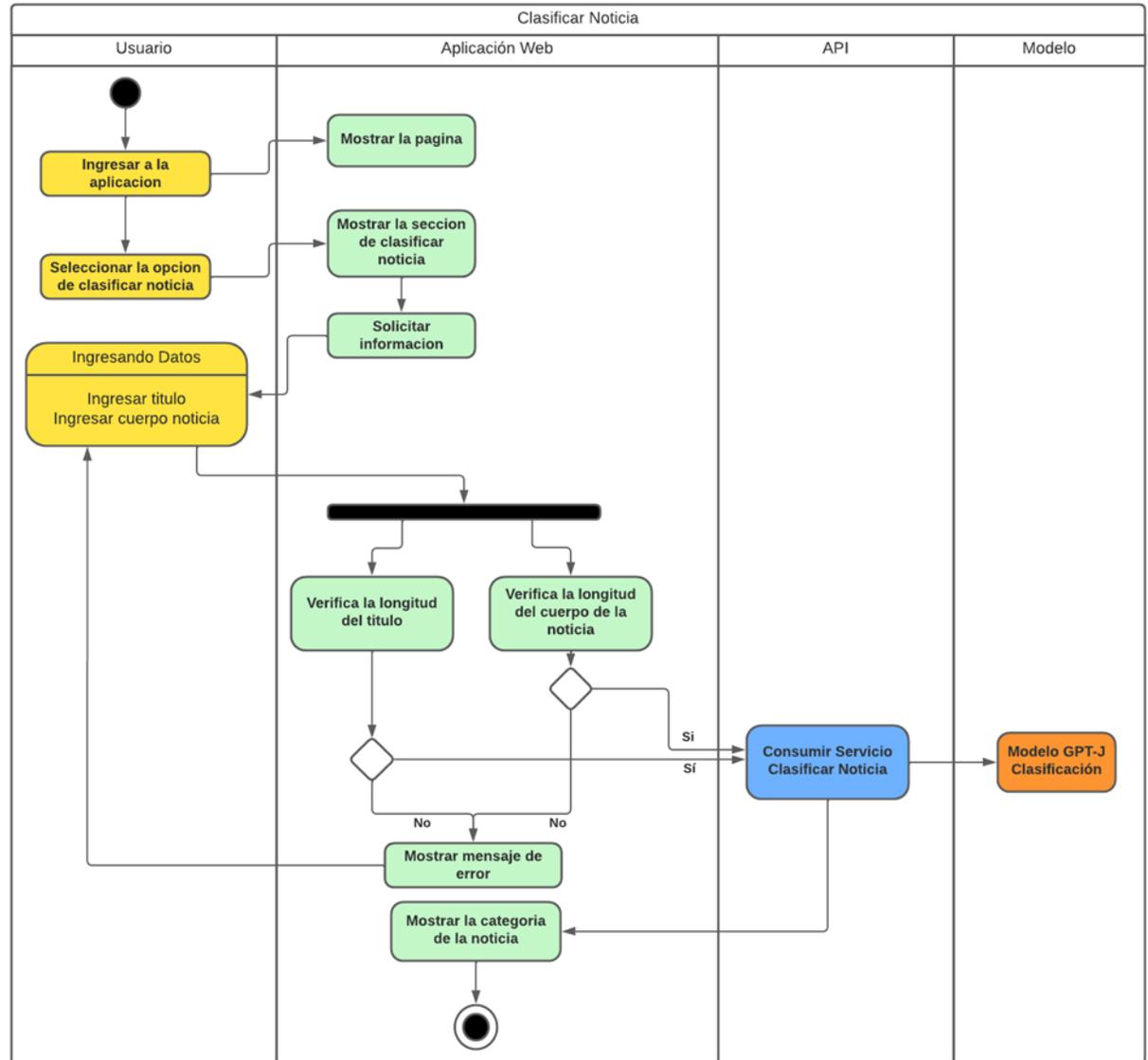


Figure 15: Diagrama de actividades clasificar noticias

## 6.3 Desarrollo

En esta sección se especifican las diferentes evidencias del desarrollo del sistema inteligente.

### 6.3.1 Desarrollo de las pruebas funcionales

En esta sección se detallará las pruebas correspondientes para verificar que la aplicación cumplan con los requerimientos previamente definidos.

<b>Prueba Funcional N° 1</b>	<b>PF-1</b>	<b>21-07-2022</b>
	<b>Responsable</b>	<b>Diana Tixi</b>
<b>Requerimiento:</b> Se recomiende una noticia en base a un titular del interés del usuario		
<b>Prueba aprobada</b>		
Resultado esperado: el usuario ingresa a la aplicación web, selecciona la opción de recomendar noticia, ingrese el tema de interés, pulse el botón de “generar”, finalmente la aplicación muestra la noticia recomendada en base al tema ingresa.		
		
<b>Resultado Obtenido</b>		



## Resultado Obtenido

**TITULAR**  
Paro Nacional en Ecuador

**PALABRAS CLAVE**  
CONAIE, Ecuador, Iza, Lasso

[Redactor](#)

**Redacción de noticias** **Paro Nacional en Ecuador**

**Palabras Claves:**  
CONAIE, Ecuador, Iza, Lasso

El presidente de la CONAIE, Leonidas Iza, fue detenido en Cotopaxi la madrugada de este martes 14 de junio de 2022. Foto: Captura de video

El presidente de la Confederación de Nacionalidades Indígenas (CONAIE), Leonidas Iza, fue detenido la madrugada de este martes 14 de junio de 2022 en Cotopaxi, según confirmó la Policía Nacional. De acuerdo con el organismo, lo capturaron en el primer día de plazo que tenía para presentar sus argumentos en defensa de la protesta social. Iza fue trasladado a la Unidad de Flagrancia de Quito, por sea lo que la organización lockenco espera que sea puesto en libertad condicional. El movimiento indígena calificó esta acción como un "secuestro" ilegal. Paro Nacional en Ecuador Desde el lunes 13 de junio, en Ecuador se vive un paro que en el último día ha provocado la movilización de cientos de personas hacia la capital. Los manifestantes buscan urge al gobierno de Guillermo Lasso para que atienda las 10 demandas que presentaron, entre ellas que se reduzcan y congelen los precios de los combustibles, que se controlen los precios de los productos de primera necesidad, que no se privaticen empresas estatales y que no se amplíe la actividad petrolera y minera en la Amazonía. Hasta el momento la movilización deja un saldo de cinco fallecidos y no menos de 200 heridos entre manifestantes y fuerzas de seguridad, así como más de 100 detenciones, según organizaciones de derechos humanos. El pasado lunes, el ministro del Interior, Patricio Carrillo, aseguró que la Policía ha detenido a más de 250 personas durante las manifestaciones por el paro nacional.

<b>Prueba Funcional N° 3</b>	<b>PF-3</b>	<b>21-07-2022</b>
	<b>Responsable</b>	<b>Diana Tixi</b>

**Requerimiento:**  
Clasificar una noticia en al título y contenido

**Prueba aprobada**

**Resultado esperado:** el usuario ingresa a la aplicación web, selecciona la opción de clasificación, ingrese el título y el contenido, pulse el botón de "clasificar", finalmente la aplicación muestra la categoría a la que pertenece la noticia



The screenshot shows the 'Noticias GPT-J' application interface. At the top, there are navigation links: 'Sobre Nosotros', '¿Como Usar?', 'Servicios', and 'UPS'. Below these are three tabs: 'Recomendaciones', 'Generación', and 'Clasificación', with 'Clasificación' being the active tab. The main form has two input fields: 'TITULAR' with the placeholder 'Ingrese el título' and 'CUERPO DE LA NOTICIA' with the placeholder 'Contenido .....'. A green 'GENERAR' button is located below the 'CUERPO DE LA NOTICIA' field. Below the button, there are two columns: 'Clasificación de Noticia' and 'Titulo Noticia'. The 'Clasificación de Noticia' column contains a large green circle and the text 'Categoria'. The 'Titulo Noticia' column contains a large green circle and the text 'Categoria'. The bottom of the screenshot shows a 'Resultado Obtenido' label.

RECOMENDACIÓN REDACCIÓN **CLASIFICACIÓN**

TITULAR

Johnny Depp podría interpretar de nuevo al Capitán Jack Sparrow. Esto es lo que asoman dos ejecutivos de Di

CUERPO DE LA NOTICIA

El Capitán Jack Sparrow tiene nombre y apellido. Más allá de cualquier situación que se genere fuera de cámaras, Johnny Depp siempre va a tener en sus manos a uno de los personajes principales de Piratas del Caribe. Por eso, enseguida se supo el dictamen de la justicia norteamericana, en su sonado caso con Amber Heard, los fanáticos comenzaron a aclamar su regreso a la saga que produce Disney. Dicha situación es común, no escapa de lo normal. Un fanático queriendo el regreso de un actor o actriz es parte de la

Clasificar

### Clasificación de noticias

## Johnny Depp podría interpretar de nuevo al Capitán Jack Sparrow: Esto es lo que asoman dos ejecutivos de Disney

El Capitán Jack Sparrow tiene nombre y apellido. Más allá de cualquier situación que se genere fuera de cámaras, Johnny Depp siempre va a tener en sus manos a uno de los personajes principales de Piratas del Caribe. Por eso, enseguida se supo el dictamen de la justicia norteamericana, en su sonado caso con Amber Heard, los fanáticos comenzaron a aclamar su regreso a la saga que produce Disney. Dicha situación es común, no escapa de lo normal. Un fanático queriendo el regreso de un actor o actriz es parte de la comidilla diaria del mundo del entretenimiento.

Categoría:  
Entretenimiento

### 6.3.2 Desarrollo del código fuente

Para realizar la codificación del sistema inteligente se utilizó diferentes herramientas. El frontend y backend se desarrolló en Python, utilizando novedosas tecnologías como fast api, graphql y redis se utilizó la herramienta Visual Studio Code, pues es un editor de código de fuente, a continuación, se presenta un diagrama de paquetes que ilustra la estructura básica del proyecto

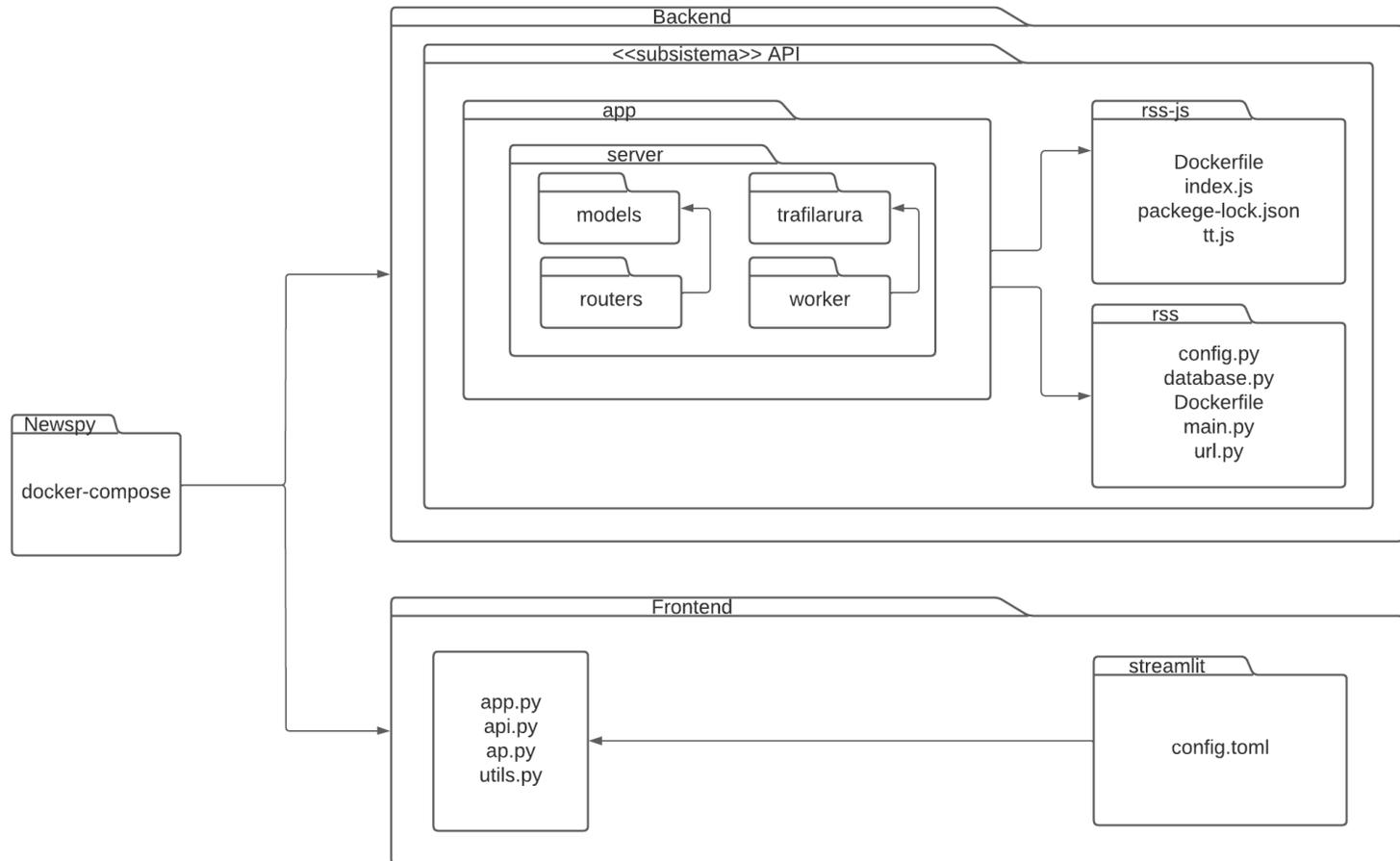


Figure 16: Estructura básica del sistema inteligente

Dentro de la estructura del backend se encuentra el proceso de recolección de datos que se divide en siete servicios por lo cual se utilizó un docker-compose para el despliegue dado que este permite unir los servicios para que actué como uno un solo sistema. A continuación, se presenta la topología del docker-compose

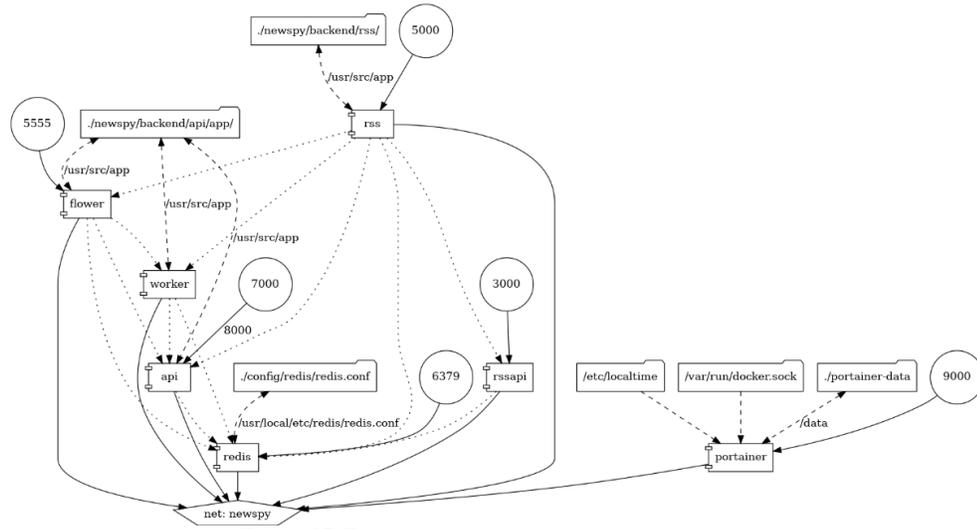


Figure 17: Topología del docker-compose

En la siguiente figura se puede apreciar un fragmento del código del backend que corresponde al modelo base para extraer los atributos de la noticia, en este caso este modelo es el formato para todos los diarios

```

1 from typing import Optional, List, Set
2 from pydantic import BaseModel, Field, HttpUrl, ValidationError, AnyUrl
3 from datetime import date, datetime, time, timedelta
4 (class) ArticleSchema
5 class ArticleSchema(BaseModel):
6     sitename: str
7     title: str
8     description: Optional[str] = None
9     author: Optional[str] = None
10    date: Optional[datetime] = None
11    main_image: Optional[HttpUrl] = None
12    image_description: Optional[str] = None
13    url: HttpUrl
14    hostname: str
15    categories: List[Set[str]] = None
16    tags: List[Set[str]] = []
17    tweets: List[Set[str]] = []
18    tweets_hashtags: List[Set[str]] = []
19    tweets_mentions: List[Set[str]] = []
20    text_keywords: List[Set[str]] = []
21    yake_keywords: List[Set[str]] = []
22    text: str

```

Figure 18: Código Fuente

Para mantener un control de versiones y almacenar el código fuente del sistema inteligente se utilizó el repositorio en la nube GitHub, a continuación, se adjunta el link del repositorio. <https://github.com/dguzmanc4/newspy.git>

### 6.3.3 Fase de aceptación

Para la fase de aceptación se realizó una encuesta a 30 personas (público en general) con el propósito de conocer el grado de aceptación de la aplicación web, esta encuesta está compuesta de 10 preguntas, de las cuales 9 son para el análisis de la aplicación y la pregunta n. °10 es para un análisis de las recomendaciones. A continuación, se presenta el análisis realizado a los encuestadores.

#### 1. ¿Con qué frecuencia usted lee noticias?

Alternativas	Respuesta	Porcentaje
Nunca	1	3%
Casi nunca	1	3%
A veces	19	64%
Casi siempre	7	23%
Siempre	2	7%

Tabla 15: Conteo de respuesta - pregunta 1



Figure 19: Resultado pregunta 1

En la figura 19 se puede apreciar los resultados obtenidos de la pregunta n. °1, en este caso se observa que la mayor parte de las personas no lee noticias con gran frecuencia.

#### 2. ¿Qué medio de comunicación utiliza para informarse?

Alternativas	Respuesta	Porcentaje
Diarios Digitales	7	23%
Radios	1	4%
Redes Sociales	9	30%
Redes Sociales, Diarios Di	10	34%
Redes Sociales, Diarios Digitales, Sitios Web	1	3%
Redes Sociales, Sitios Web	1	3%
Sitios Web	1	3%

Tabla 16: Conteo de respuesta - pregunta 2

**Medio de comunicación para informarse**

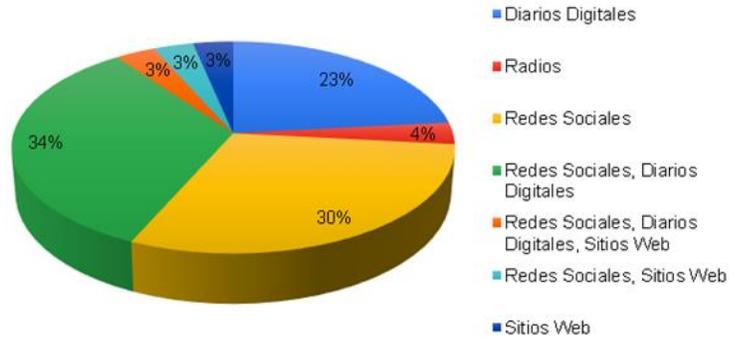


Figure 20: Resultado pregunta 2

En este caso se analizó los medios de comunicación que son utilizados como herramienta de información, como se puede apreciar en la figura presentada se obtiene que el 34% de las personas utilizan redes sociales y diarios digitales para enterarse de los acontecimientos, pero también cabe recalcar que el 30% de personas utiliza diarios digitales como fuente de información.

3. ¿En base a su criterio que calificación le pondría a la recomendación generada por la aplicación?

Alternativas	Respuesta	Porcentaje
Buena	13	43%
Aceptable	13	44%
Mala	4	13%

Tabla 17: Conteo de respuesta - pregunta 3

**Sistema de Recomendación**

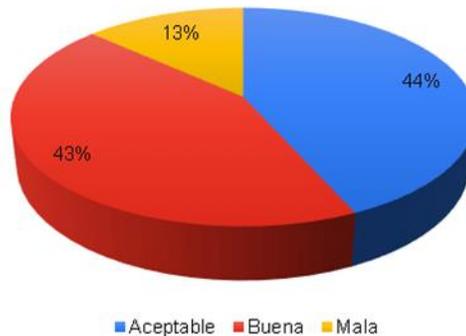


Figure 21: Resultado pregunta 3

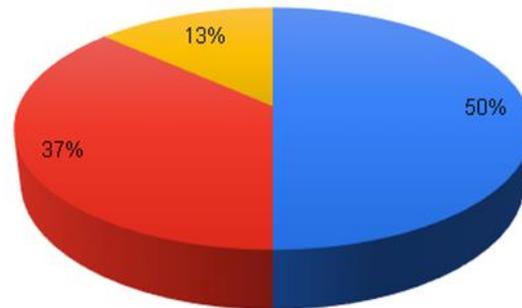
En la figura 21 se puede apreciar que el resultado obtenido ante el análisis del servicio de recomendación basado en un tema de interés fue aceptable.

4. ¿En base a su criterio, usted cree que la noticia generada por el sistema es?

Alternativas	Respuesta	Porcentaje
Buena	11	37%
Aceptable	15	50%
Mala	4	13%

Tabla 18: Conteo de respuesta - pregunta 4

#### Sistema de Generación



■ Aceptable ■ Buena ■ Mala

Figure 22: Resultado pregunta 4

Como resultado de la pregunta n. °4 se obtuvo que el 50% de las personas considera la noticia generada es aceptable, pero cabe recalcar que el 37% de las personas considera a la noticia como buena, deduciendo que el sistema de generación si tiene una gran acogida.

5. ¿En base al resultado obtenido, usted cree que el proceso de clasificación de noticias es?

Alternativas	Respuesta	Porcentaje
Buena	13	43%
Aceptable	16	54%
Mala	1	3%

Tabla 19: Conteo de respuesta - pregunta 5

### Sistema Clasificación

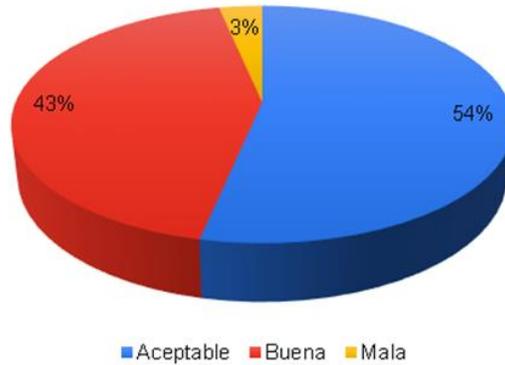


Figure 23: Resultado pregunta 5

En la figura anterior se puede apreciar que el 54% de las personas catalogan al servicio como aceptable y un 43% como bueno, dicho esto se puede asumir que el 97% de las personas encuestadas consideran el que servicio de clasificación de noticias es bueno.

6. ¿Para realizar el proceso de recomendación de noticias basado en el titulo se utilizó un modelo recomendador el cual tiene como tiempo promedio de respuesta de 3 minutos, como considera usted este tiempo?

Alternativas	Respuesta	Porcentaje
Buena	10	33%
Aceptable	15	50%
Mala	5	17%

Tabla 20: Conteo de respuesta - pregunta 6

### Tiempo de respuesta del sistema de recomendación

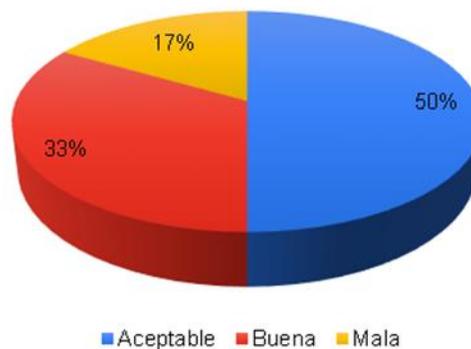


Figure 24: Resultado pregunta 6

La figura 21 nos indica que el 50% de las personas consideran que el tiempo de respuesta para obtener una recomendación en base a un tema de interés es aceptable, mientras que el 33% considera un tiempo de respuesta bueno.

7. ¿Para realizar el proceso de generación de noticias basado en el título y palabras clave se utilizó un modelo el cual tiene como tiempo promedio de respuesta de 2 - 3 minutos, como considera usted este tiempo?

Alternativas	Respuesta	Porcentaje
Buena	9	30%
Aceptable	14	47%
Mala	7	23%

Tabla 21: Conteo de respuesta - pregunta 7

### Tiempo de respuesta del sistema de generación

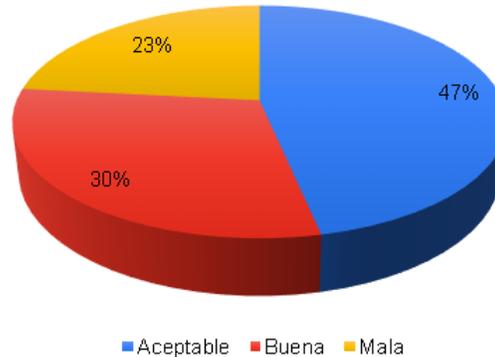


Figure 25: Resultado pregunta 7

En esta figura, se puede observar que la mayor parte de personas respondieron que el tiempo de respuesta del servicio de generación de noticias es aceptable, mientras que un 30% respondió que el tiempo es bueno, pero el 23% categorizó como malo, deduciendo que existe un pequeño grupo de personas que consideran que debería mejorar el tiempo de respuesta.

8. ¿Para realizar el proceso de clasificación de noticias se utilizó un modelo que tarda en promedio 20 segundos, usted considera que el tiempo de respuesta es?

Alternativas	Respuesta	Porcentaje
Buena	15	50%
Aceptable	14	47%
Mala	1	3%

Tabla 22: Conteo de respuesta - pregunta 8

### Tiempo de respuesta del sistema de clasificación

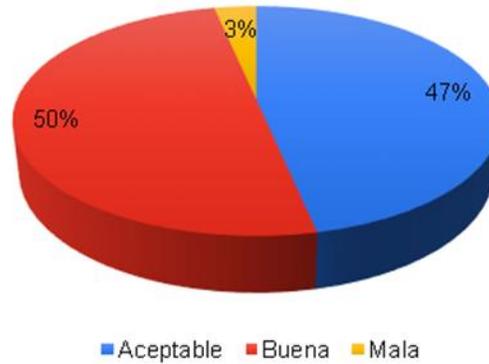


Figure 26: Resultado pregunta 8

En este caso se puede verificar en la figura 8 que el tiempo de respuesta del servicio de clasificación es considerado como bueno dado que la mayor parte de las personas encuestas respondieron como bueno y aceptable.

9. ¿Después de analizar los diferentes servicios que proporciona nuestro sistema inteligente usted considera que la aplicación es de gran ayuda?

Alternativas	Respuesta	Porcentaje
SI	1	97%
NO	29	3%

Tabla 23: Conteo de respuesta - pregunta 9

### Importancia de la aplicación

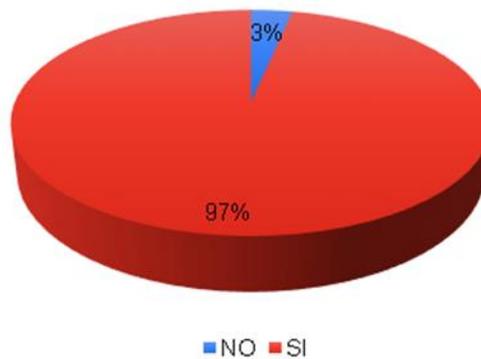


Figure 27: Resultado pregunta 9

Finalmente se obtuvo como resultado de la pregunta n. °9, que una persona considera que la aplicación no es de gran ayuda, mientras que el resto de encuestados difieren ya que respondieron que ellos respondieron que la aplicación si es de ayuda.

Para el análisis de los resultados se aplicó el coeficiente alfa de Cronbach, según (Oviedo1 & Campo-Arias, 2005) establece que “es un índice utilizado para medir la confiabilidad de interna de una escala”. El coeficiente de alfa de Cronbach tiene un enfoque que permite determinar la correlación que existe entre ítems (preguntas) de una prueba, Lee Cronbach establece que el valor obtenido de la correlación debe ser entre 0.70 y 0.90 dado que al ser menor indica que la consistencia interna es baja, como también si dicho valor excede del rango se puede considerar que existe redundancia entre ítems.

Tomando como base de conocimientos, se realizo el cálculo del coeficiente alfa de Cronbach mediante la varianza de los ítems, obteniendo como resultado un valor de 0.87 llegando a la conclusión de que la prueba realiza tiene un grado de confiabilidad bueno. Basándonos en el análisis interior se puede decir que la aplicación tiene un grado de aceptación considerable, dado que los resultados obtenidos son favorables, de tal forma se considera a la aplicación como una herramienta que facilita la búsqueda de noticias ecuatorianas, ya que hoy en día gran parte de la población se informa a través de sitios web, concluyendo se puede decir que aplicación tendría a una gran acogida debido a que es un sitio web que ha mas de identificar, clasificar, recomendar, genera un resumen rápido del contenido de tal forma el usuario comprenda la noticia de manera rápida.

## 7. Cronograma

En este capítulo se presenta el cronograma de las actividades a realizar durante el desarrollo de la aplicación móvil, y finalmente el presupuesto tentativo para este proyecto.

Actividades		Responsable	Duración (Horas)	Comienzo	Fin
	<b>Proyecto</b>		600	11/04/2022	17/07/2022
	<b>Sprint 1</b>		220	11/04/2022	06/05/2022
<b>OE.1</b>			<b>90</b>		
	<b>ACT.1</b> Realizar el estado del arte de Open IA, GPT-J y tecnologías relacionadas.	DG-DT	20	11/04/2022	12/04/2022
	<b>ACT.2</b> Estudiar la arquitectura y usos de GPT-J.	DG-DT	10	13/04/2022	14/04/2022
	<b>ACT.3</b> Identificar los modelos y técnicas más efectivas para la implementación y ajustes de GPT-J.	DG-DT	30	15/04/2022	19/04/2022
	<b>ACT.4</b> Realizar una documentación de la información relevante para la implementación de GPT-J.	DG-DT	20	20/04/2022	22/04/2022
	Revisión/Corrección	DG-DT-DQ	10	22/04/2022	22/04/2022
<b>OE.2</b>			<b>130</b>		
	<b>ACT.1</b> Recolectar noticias de ámbito nacional haciendo uso de Web Scraping en páginas de diversos medios de comunicación.	DG-DT	40	23/04/22	27/04/2022
	<b>ACT.2</b> Limpiar y estructurar las noticias recolectadas.	DG-DT	40	27/04/2022	30/04/2022

	<b>ACT.3</b> Generar repositorio de noticias.	DG-DT	30	30/04/2022	04/05/2022
Revisión/Corrección		DG-DT-DQ	20	04/05/2022	06/05/2022
	<b>Sprint 2</b>		230	09/05/2022	15/06/2022
<b>OE.3</b>			<b>230</b>		
	<b>ACT.1</b> Analizar y definir los mejores parámetros para ajustar el modelo a nuestro problema.	DG-DT	10	09/05/2022	10/05/2022
	<b>ACT.2</b> Adaptar el modelo de GPT-J para nuestro problema.	DG-DT	50	11/05/2022	25/05/2022
	<b>ACT.4</b> Entrenar el modelo GPT-J en base al repositorio de noticias.	DG-DT	150	25/05/2022	13/06/2022
	<b>ACT.5</b> Realizar pruebas de funcionalidad y rendimiento.	DG-DT	10	14/06/2022	14/06/2022
Revisión/Corrección		DG-DT-DQ	10	15/06/2022	15/06/2022
	<b>Sprint 3</b>		105	16/06/2022	07/07/2022
<b>OE.4</b>			<b>105</b>		
	<b>ACT.1</b> Definir requerimientos para una PWA.	DG-DT	5	16/06/2022	16/06/2022
	<b>ACT.2</b> Desarrollar los requerimientos para el desarrollo de la PWA.	DG-DT	5	17/06/2022	17/06/2022
	<b>ACT.3</b> Generar los servicios REST para el consumo de GPT-J.	DG-DT	10	20/06/2022	22/06/2022

	<b>ACT.4</b> Diseño y generación de pruebas unitarias.	DG-DT	5	23/06/2022	23/06/2022
	<b>ACT.5</b> Implementar una de la PWA en base a los requerimientos definidos.	DG-DT	40	24/06/2022	30/06/2022
	<b>ACT.6</b> Realizar pruebas de funcionalidad y rendimiento.	DG-DT	05	01/07/2022	01/07/2022
	<b>ACT.7</b> Realizar documentación sobre la funcionalidad y uso de la PWA.	DG-DT	10	02/07/2022	04/07/2022
	<b>ACT.8</b> Desplegar el sistema en la nube.	DG-DT	15	05/07/2022	06/07/2022
	Revisión/Corrección	DG-DT-DQ	10	07/07/2022	07/07/2022
	<b>Sprint 4</b>		45	08/07/2022	17/07/2022
<b>OE.5</b>			<b>45</b>		
	<b>ACT.1</b> Generar encuestas para probar y validar el correcto funcionamiento de GPT-J.	DG-DT	10	08/07/2022	08/07/2022
	<b>ACT.2</b> Aplicar y analizar la retroalimentación obtenida de los usuarios.	DG-DT	10	11/07/2022	13/07/2022
	<b>ACT.3</b> Realizar el registro de los resultados obtenidos.	DG-DT	5	14/07/2022	14/07/2022
	Revisión/Corrección	DG-DT-DQ	10	15/07/2022	17/07/2022

## 8. Presupuesto

A continuación, se detallará el presupuesto necesario para el desarrollo del sistema inteligente

Denominación	Cant.	Costo unitario	Costo total
	Unidades	Dólares	Dólares
<b>1. Tecnológico</b>			
Computador portátil	2	\$1500,00	\$3.000,00
<b>2. Servicios</b>			
Servicio de Internet	2	\$28,00	\$168,00
Servicio en la Nube	3	\$300	\$000,00
Google Colab Pro+	2	\$54	\$108,00
Servicios de Transporte	60	\$1,00	\$60,00
<b>3. Personal</b>			
Estudiante/Desarrollador	600 horas	\$20,00	\$12.000,00
Asesoría especializada	60 horas	\$20,00	\$1.200,00
Curso de Conocimientos	2 horas	\$ 12.00	\$12.00
<b>4. Otros</b>			
Imprevisto	1	\$200,00	\$200,00
<b>TOTAL</b>			<b>\$16.548,00</b>

## 9. Conclusiones

Para el desarrollo del sistema inteligente se realizó una revisión del estado del arte de los diferentes temas de interés tales como: procesamiento de lenguaje natural, sistemas recomendadores, generación de texto, clasificación de texto, generación de resumen, recopilación de datos a través de Feed RSS, modelos pre entrenados como GPT-J, programación con React.js, implementación de una TPU, entre otros. El propósito de esta revisión es adquirir un conocimiento base que sea útil al momento de implementar.

Dentro de la parte de la recolección de datos, podemos concluir que la parte más relevante de este proceso es identificar la prensa digital más relevante y la frecuencia de actualización que esta tiene dado que la recolección de datos se realiza con una frecuencia de 30 min para los diarios con prioridad alta, 2 horas para los diarios de prioridad media y 6 horas para los diarios de prioridad baja, cabe mencionar que la extracción de la información se realiza de 46 diarios digitales o sitios web informativos, a cada diario o sitio se le categorizo como alto, medio y bajo, dependiendo la frecuencia de actualización. Además, debe contar con los sitios o prensa relevante se debe considerar los Feed RSS y la forma de extraer los datos, ya que a través de estos lectores se obtiene la información porque en ellos se consolida todo el contenido actualizado del sitio web

La parte del diseño e implementación del modelo GPT-J es un punto de suma importancia dado que el objetivo es identificar y clasificar la noticia que el usuario ingresa, como también crear una noticia ingresando el título y palabras clave, pero cabe mencionar que noticia generada debe tener un contexto coherente en base a los parámetros ingresados. Finalmente, debe ser capaz de recomendar una noticia solo ingresando el tema de interés del usuario.

Para desarrollar cada modelo se utilizó el modelo pre –entrenado GPT-J el cual se acoplo y se entrenó dependiendo de nuestros requerimientos, en caso el modelo debía cumplir un formato pre establecido por los autores dado que es la única forma de entrenar, para ello se realizó un proceso adicional, se utilizó modelos pre entrenados para crear dicha data y en base a esa data genera poder entrenar el modelo GPT-J.

Cabe mencionar que en el artículo título “SGPT: GPT Sentence Embeddings for Semantic Search” se demuestra que los modelos GPT son capaz de realizar tareas de búsqueda semántica a través de transformadores, decodificadores y otras configuraciones, en nuestro caso esta solución no es viable ya que al no contar con computadoras de alto rendimiento el tiempo de respuesta no es óptimo, por dicha razón se concluye que el modelo GPT-J requiere de un hardware sofisticando, por lo cual se recomienda implementar modelos que realizan tareas específicas.

La implementación de la PWA se desarrolló con la React.js dado que nos permite crear aplicaciones web SPA. Se optó por esta librería, dado que nos permite crear interfaces a partir de componentes, esto quiere decir que no es necesario cambiar el código de la página para la manipulación de los datos, ya que React.js lo realiza de forma automática. Garantizando el uso de tecnologías innovadoras y de bajo costo de implementación

Finalmente analizando el resultado de la encuesta que validan el funcionamiento y el grado de satisfacción de la aplicación, se puede verificar que se cumple con el objetivo planteado, dado que se obtuvo el 97% del grado de aceptación en las personas encuestadas, pero cabe recalcar que dicho valor nos da paso a un análisis profundo de las recomendaciones y sugerencias de nuestra aplicación.

## 10. Recomendaciones

Este trabajo ha demostrado que GPT-J se puede ajustar a tareas de clasificación, redacción y recomendación, a la vez que se obtienen buenos resultados. Sin embargo, el mayor desafío de estos modelos es la implementación para su producción en el mundo real, en donde se necesita hardware sofisticado y tiempos de respuesta inferiores a los 60 segundos para mantener el modelo escalable y confiable.

Teniendo en cuenta los tiempos de respuestas obtenidos al implementar los diferentes modelos, se considera que GPT-J no es adecuado para todas las organizaciones, cabe mencionar que existe diferentes modelos diseñados para realizar tareas en específico como la clasificación, resumen, redacción y recomendación, con los cuales se obtienen resultados similares y un mejor tiempo de respuesta de la que se obtuvo en este trabajo

A continuación, se mencionan alternativas a GPT-J para realizar tareas en específico.

- Similitud semántica: all-mpnet-base-v2
- Resumen: mT5 multilingual XLSum
- Clasificación: finbert
- Generación de texto: gpt2

Finalmente, en base al desarrollo e implementación de dicho proyecto se plantea como trabajo a futuro, mejorar el sistema en lo que se refiere a tiempos de respuesta y entrenamiento a de los modelos, dado que son fases estratégicas para obtener un mejor resultado.

## Referencias Bibliográficas

- Beltrán, N. C., & Rodríguez, E. C. (2021). Procesamiento del lenguaje natural (PLN) -GPT-3, y su aplicación en la Ingeniería de Software. TIA, 23.
- Forefront Team. (2021, October 14). GPT-J-6B: An Introduction to the Largest Open Source GPT Model | Forefront. forefront.ai. Retrieved April 6, 2022, from <https://www.forefront.ai/blog-posts/gpt-j-6b-an-introduction-to-the-largest-open-sourced-gpt-model>
- Salinas, J. (2021, December 8). Cómo construir un chatbot con GPT-3 o GPT-J. NLP Cloud. Retrieved April 6, 2022, from <https://nlplcloud.io/es/how-to-build-chatbot-gpt-3-gpt-j.html>
- Cyphert, A. B. (2021). A Human Being Wrote This Law Review Article: GPT-3 and the Practice of Law. West Virginia University - College of Law, Virginia. Obtenido de [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=3973961](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3973961)
- INEC. (2021). Tecnologías de la Información y Comunicación, 2020. INEC. Recuperado el 30 de 03 de 2022, de [https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas\\_Sociales/TIC/2020/202012\\_Principales\\_resultados\\_Multiproposito\\_TIC.pdf](https://www.ecuadorencifras.gob.ec/documentos/web-inec/Estadisticas_Sociales/TIC/2020/202012_Principales_resultados_Multiproposito_TIC.pdf)
- OpenAI. (25 de marzo de 2021). GPT-3 impulsa la próxima. Recuperado el 25 de marzo de 2022, de <https://openai.com/blog/gpt-3-apps/>
- Reuters Institute. (2020). Reuters Institute. Recuperado el 30 de 03 de 2022, de Reuters Institute: <https://www.digitalnewsreport.org/survey/2020/overview-key-findings-2020/>
- The Pile. (2020). The Pile. Retrieved April 6, 2022, from <https://pile.eleuther.ai/>
- Wang, B. (2021, mayo). EleutherAI/gpt-j-6B · Hugging Face. Hugging Face. Retrieved April 6, 2022, from <https://huggingface.co/EleutherAI/gpt-j-6B>
- Jimenez, J. (2016). Descubre React. Recuperado el 11 de July de 2022, de ACADEMIA: [https://www.academia.edu/download/60098162/Descubre\\_React\\_\\_2da\\_Edicion\\_-\\_Javi\\_Jimenez-FREELIBROS20190723-17083-qn5tuy.pdf](https://www.academia.edu/download/60098162/Descubre_React__2da_Edicion_-_Javi_Jimenez-FREELIBROS20190723-17083-qn5tuy.pdf)
- Mueller, V. (2021, August 26). Access and use GPT-J (GPT J ). Towards Data Science. Retrieved April 8, 2022, from <https://towardsdatascience.com/how-you-can-use-gpt-j-9c4299dd8526>
- Gallego M. T. (2012). Metodología SCRUM Universidad Oberta de Catalunya.
- Cunningham, R. (2021, 25 mayo). Introducing Power Apps Ideas: AI-powered assistance now helps anyone create apps using natural language. Microsoft Power Apps.

- Berbel, P. (27 de Julio de 2018). *Desarrollo de un frontend en ReactJS*. Recuperado el 11 de July de 2022, de Repositori UJI: <http://repositori.uji.es/xmlui/handle/10234/179297>
- Brings, J. (5 de Mayo de 2021). *BERT For Measuring Text Similarity*. Obtenido de BERT For Measuring Text Similarity: <https://towardsdatascience.com/bert-for-measuring-text-similarity-eec91c6bf9e1>
- Forefront. (23 de Septiembre de 2021). *Preparing a Dataset to Fine-tune GPT-J | Forefront*. Obtenido de Preparing a Dataset to Fine-tune GPT-J | Forefront: <https://www.forefront.ai/blog-posts/preparing-a-dataset-to-fine-tune-gpt-j>
- Holtzman, A., Buys, J., Du, L., Forbes, M., & Choi, Y. (2020). THE CURIOUS CASE OF NEURAL TEXT DeGENERATION. *ICLR*, (pág. 16). Washington. Obtenido de <https://arxiv.org/pdf/1904.09751.pdf>
- Jimenez, J. (2016). *Descubre React*. Recuperado el 11 de July de 2022, de ACADEMIA: [https://www.academia.edu/download/60098162/Descubre\\_React\\_\\_2da\\_Edicion\\_-\\_Javi\\_Jimenez-FREELIBROS20190723-17083-qn5tuy.pdf](https://www.academia.edu/download/60098162/Descubre_React__2da_Edicion_-_Javi_Jimenez-FREELIBROS20190723-17083-qn5tuy.pdf)
- Karakaya, M. (19 de Mayo de 2021). *medium.com*. Obtenido de Fundamentals of Text Generation: <https://medium.com/deep-learning-with-keras/fundamentals-of-text-generation-745d66238a1f>
- Komatsuzaki, A. (4 de 06 de 2021). *GPT-J-6B: Transformador basado en JAX 6B*. Obtenido de GPT-J-6B: Transformador basado en JAX 6B: <https://arankomatsuzaki.wordpress.com/2021/06/04/gpt-j/>
- Kublik, S., & Saboo, S. (2022). *GPT-3 Building Innovative NLP Products Using Large Language Models*. O'Reilly Media; 1er edición.
- Mahesh, B. (2018). Machine Learning Algorithms - A Review. *Revista Internacional de Ciencia e Investigacion (IJSR)*, 6. Obtenido de [https://www.researchgate.net/profile/Batta-Mahesh/publication/344717762\\_Machine\\_Learning\\_Algorithms\\_-\\_A\\_Review/links/5f8b2365299bf1b53e2d243a/Machine-Learning-Algorithms-A-Review.pdf?eid=5082902844932096](https://www.researchgate.net/profile/Batta-Mahesh/publication/344717762_Machine_Learning_Algorithms_-_A_Review/links/5f8b2365299bf1b53e2d243a/Machine-Learning-Algorithms-A-Review.pdf?eid=5082902844932096)
- Muennighoff, N. (2022). *SGPT: GPT Sentence Embeddings for Semantic*. Peking. Obtenido de <https://arxiv.org/pdf/2202.08904.pdf>
- Oviedo1, H. C., & Campo-Arias, A. (2005). Aproximación al uso del coeficiente alfa. *Revista Colombiana de Psiquiatría*, 9.
- Pérez Borrero, I., & Gegúndez Arias, M. E. (2021). *Deep learning: fundamentos, teoría y aplicación*. Universidad de Huelva. Obtenido de

[https://books.google.com.ec/books?id=kzsvEAAAQBAJ&pg=PA192&dq=deep+learning&hl=es&sa=X&ved=2ahUKEwj2\\_bWf-e74AhXyRzABHRMzAXkQ6AF6BAGLEAI#v=onepage&q=deep%20learning&f=true](https://books.google.com.ec/books?id=kzsvEAAAQBAJ&pg=PA192&dq=deep+learning&hl=es&sa=X&ved=2ahUKEwj2_bWf-e74AhXyRzABHRMzAXkQ6AF6BAGLEAI#v=onepage&q=deep%20learning&f=true)

Pino Díez, R., Gómez Gómez, A., & Abajo Martínez, N. d. (2001). *Introducción a la inteligencia artificial: sistemas expertos, redes neuronales artificiales y computación evolutiva*. Servicio de Publicaciones, Universidad de Oviedo.

Singh, G., & Sahu, S. (2015). Review on “Really Simple Syndication (RSS) Technology Tools”. *2015 IEEE International Conference on Computational Intelligence & Communication Technology*, (pág. 5).

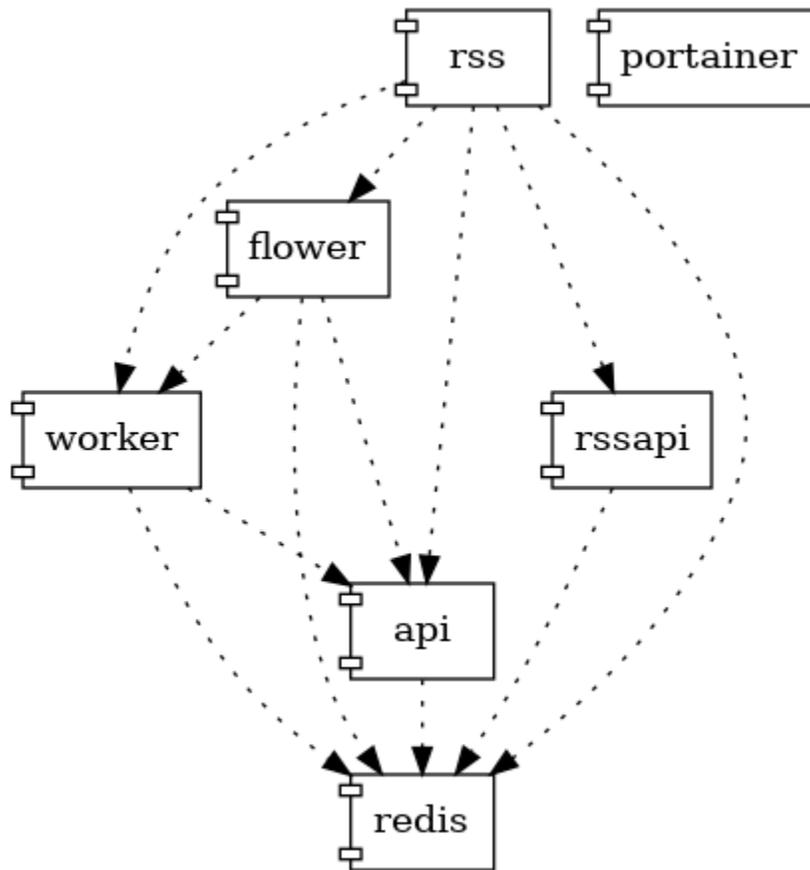
Tamayo, S. C. (2012). *Diseño y Validación de Modelos para Sistemas de Recomendación*. Universidad de Granada. Granada: Editorial de la Universidad de Granada. Obtenido de <https://digibug.ugr.es/bitstream/handle/10481/29800/21559533.pdf?sequence=1&isAllowed=y>

Tingiris, S. (2021). *Exploring GPT-3*. Packt Publishing.

Wang, & Ben. (Mayo de 2021). *Mesh-Transformer-JAX: Model-Parallel Implementation of Transformer Language Model with JAX*. Obtenido de Mesh-Transformer-JAX: Model-Parallel Implementation of Transformer Language Model with JAX: <https://github.com/kingoflolz/mesh-transformer-jax>

# Anexos

Anexo 1: Topología recolección de datos



Anexo 2: Total de noticias recolectadas

articles				
Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
114.28 MB	53 K	3.84 kB	1	983.04 kB

Anexo 3: Parámetros de la red neuronal del modelo de clasificación

<b>Parámetros de la red</b>	
<b>Parámetro</b>	<b>Valor</b>
<b>anneal_steps</b>	3146
<b>d_model</b>	4096
<b>end_lr</b>	0.000003
<b>gradient_accumulation_steps</b>	32
<b>layers</b>	28
<b>lr</b>	0.00003
<b>n_heads</b>	16
<b>n_vocab</b>	50400
<b>norm</b>	layernorm
<b>pe</b>	rotary
<b>pe_rotary_dims</b>	64
<b>per_replica_batch</b>	1
<b>seq</b>	2048
<b>total_steps</b>	3420
<b>val_batches</b>	920
<b>warmup_steps</b>	274
<b>epochs</b>	10

Anexo 4: Parámetros de la red neuronal del modelo de generación

<b>Parámetros de la red</b>	
<b>Parámetro</b>	<b>Valor</b>
<b>anneal_steps</b>	3839
<b>d_model</b>	4096
<b>end_lr</b>	0.000003
<b>gradient_accumulation_steps</b>	32
<b>layers</b>	28
<b>lr</b>	0.00003
<b>n_heads</b>	16
<b>n_vocab</b>	50400
<b>norm</b>	layernorm
<b>pe</b>	rotary
<b>pe_rotary_dims</b>	64
<b>per_replica_batch</b>	1
<b>seq</b>	2048
<b>total_steps</b>	4128
<b>val_batches</b>	3326
<b>warmup_steps</b>	289
<b>epochs</b>	8

Anexo 5: Parámetros de la red neuronal del modelo de resumen

<b>Parámetros de la red</b>	
<b>Parámetro</b>	<b>Valor</b>
<b>anneal_steps</b>	3839
<b>d_model</b>	4096
<b>end_lr</b>	0.000003
<b>gradient_accumulation_steps</b>	32
<b>layers</b>	28
<b>lr</b>	0.00003
<b>n_heads</b>	16
<b>n_vocab</b>	50400
<b>norm</b>	layernorm
<b>pe</b>	rotary
<b>pe_rotary_dims</b>	64
<b>per_replica_batch</b>	1
<b>seq</b>	2048
<b>total_steps</b>	4128
<b>val_batches</b>	3326
<b>warmup_steps</b>	289
<b>epochs</b>	8

*Anexo 6: Resultados del entrenamiento del modelo de clasificación*

<b>Valor de pérdida para la validación (val loss)</b>	2.3576934337615967
<b>Tokens procesados</b>	224133120
<b>Tiempo empleado</b>	12h 23m 47s
<b>Valor de pérdida para la validación (val loss)</b>	2.3576934337615967

*Anexo 7: Resultados del entrenamiento del modelo de generación*

<b>Valor de pérdida (loss)</b>	0.004068050999194384
<b>Valor de pérdida para la validación (val loss)</b>	2.0450634956359863
<b>Tokens procesados</b>	270532608
<b>Tiempo empleado</b>	15h 55m 46s

*Anexo 8: Resultados del entrenamiento del modelo de resumen*

<b>Valor de pérdida (loss)</b>	0.003043978475034237
<b>Valor de pérdida para la validación (val loss)</b>	2.0965466499328613
<b>Tokens procesados</b>	237895680
<b>Tiempo empleado</b>	13 17m 16s



## Guia de uso

Newspy cuenta con tres servicios, por favor lea detenidamente el uso de cada uno para poder obtener resultados deseados.

**RECOMENDACIÓN**  
Este servicio recomienda una noticia en base al titular ingresado por el usuario, mediante el uso de similitud semántica, Newspy recomendará una noticia de entre las 46k disponibles. Se recomienda ingresar un titular de más de 4 palabras. **Ejemplo:**

**Titular:** Alcaldía de Quito tiene más precandidatos.

**REDACCIÓN**  
Este servicio genera una noticia en base a un titular y palabras clave ingresadas por el usuario. Se recomienda que el titular sea más de 4 palabras, mientras que las palabras clave deberán estar separadas por una coma,

**CLASIFICACIÓN**  
Este servicio clasificará una noticia en base al titular y cuerpo de noticia ingresado por el usuario. Para ello se deberá ingresar un titular con más de 4 palabras, mientras que el cuerpo de la noticias deberá contener al menos 50 palabras para obtener buenos resultados. **Ejemplo:**

**Titular:** Alcaldía de Quito tiene más precandidatos.  
**Cuerpo de la noticia:** A ocho días de que empiecen las

Anexo 11: Aplicación web sección servicios

NEWSPY Sobre Nosotros ¿Como usar? Servicios UPS

RECOMENDACIÓN REDACCIÓN CLASIFICACIÓN

TITULAR

Leonidas Iza sera enjuiciado

Generar

Anexo 12: Aplicacion web sección redacción de noticias

NEWSPY Sobre Nosotros ¿Como usar? Servicios UPS

RECOMENDACIÓN REDACCIÓN CLASIFICACIÓN

TITULAR

Leonidas Iza sera enjuiciado

PALABRAS CLAVE

Iza,paro Ecuador, Quito

Redactar

Anexo 13: Aplicación web sección clasificar noticia

NEWSPY [Sobre Nosotros](#) [¿Como usar?](#) [Servicios](#) UPS

RECOMENDACIÓN REDACCIÓN **CLASIFICACIÓN**

**TITULAR**

Leonidas Iza sera enjuiciado

**CUERPO DE LA NOTICIA**

Fue encontrado ...

Clasificar