

UNIVERSIDAD NACIONAL DE SAN ANTONIO ABAD DEL CUSCO

UNSAAC

FINE TUNING - LLAMA 2

Estudiante

Anthony Mayron Lopez Oquendo

184604

Docente

Carlos Fernando Montoya Cubas

Semestre 23 - II

Escuela Profesional de Ingenieria Informatica
y de Sistemas

RESUMEN

El modelo LLAMA 2, conocido por su capacidad para procesar información textual de manera avanzada, será personalizado para abordar las particularidades del lenguaje utilizado en el ámbito de educación bancaria. El conjunto de datos del centro de capacitación bancario será utilizado para entrenar y refinar el modelo LLAMA 2. Este conjunto incluirá información variada, como manuales, preguntas frecuentes, y otros recursos textuales relevantes para la formación en el ámbito bancario.

Durante el proceso de Fine Tuning, se llevará a cabo una exhaustiva evaluación del rendimiento del modelo en tareas específicas relacionadas con el sector bancario, como la respuesta a consultas sobre productos financieros, interpretación de regulaciones, y generación de contenido educativo. Este proceso iterativo permitirá ajustar los hiperparámetros del modelo para maximizar su eficacia en situaciones prácticas y reales dentro del centro de capacitación. Este enfoque personalizado contribuirá a una experiencia de aprendizaje más efectiva y adaptada a las necesidades específicas del sector financiero, beneficiando tanto a instructores como a estudiantes.

CONTENTS

Resumen	1
1 Introducción	3
1.1 Modelo Transformer	3
1.2 Centro de Capacitación Bancaria y Adaptación Tecnológica.....	4
2 Documentación: Hardware, Datos y Modelo	5
2.0.1 Arquitectura Utilizada: Google Colab con Procesador T4	5
2.0.2 Lenguaje de Programación:.....	6
2.1 Datos	6
2.1.1 Definición del Contexto Educativo Financiero	7
2.1.2 Creación de Filas de Datos	7
2.1.3 Generación de Datos en Formato Preguntas y Respuestas	7
2.2 Modelo.....	8
2.2.1 Instalación de Herramientas:	8
2.2.2 Parametros	10
2.2.3 Métricas de Entrenamiento.....	11
2.2.4 Pruebas	12
3 Resultados	14
4 Discusión	15
References	16

1. INTRODUCCIÓN

1.1. MODELO TRANSFORMER

En la era actual de procesamiento de lenguaje natural y aprendizaje profundo, el Modelo Transformer ha emergido como una innovación revolucionaria que ha transformado fundamentalmente la manera en que los sistemas de inteligencia artificial abordan tareas relacionadas con el procesamiento de secuencias de texto. Propuesto por Vaswani [1], el Modelo Transformer ha alcanzado prominencia debido a su capacidad para capturar relaciones de largo alcance y su eficiencia en paralelización, superando las limitaciones de los enfoques secuenciales previos.

A diferencia de los modelos recurrentes, el Transformer se basa en un mecanismo de atención que permite procesar todas las posiciones de la secuencia de entrada simultáneamente, mejorando significativamente la velocidad de entrenamiento y la capacidad de capturar dependencias a larga distancia. Este enfoque ha demostrado ser especialmente eficaz en tareas como traducción automática, generación de texto y comprensión del lenguaje natural.

La versatilidad del Modelo Transformer ha llevado a su adopción en diversas aplicaciones, desde asistentes virtuales hasta modelos de lenguaje masivo como GPT (Generative Pre-trained Transformer). Su estructura modular y su capacidad para aprender representaciones contextuales han permitido avances notables en el campo, consolidándolo como un pilar fundamental en el desarrollo de sistemas de inteligencia artificial orientados al procesamiento de secuencias de texto. Esta introducción explorará los principios clave detrás del Modelo Transformer y su impacto significativo en la evolución de la inteligencia artificial contemporánea.

1.2. CENTRO DE CAPACITACIÓN BANCARIA Y ADAPTACIÓN TECNOLÓGICA

En el dinámico entorno bancario actual, la formación y el desarrollo continuo de los profesionales desempeñan un papel crucial para mantenerse a la vanguardia de las complejidades financieras y las demandas regulatorias en constante evolución. En este contexto, los centros de capacitación bancaria se erigen como pilares fundamentales para el crecimiento y la excelencia en la industria.

Este centro de capacitación bancaria se presenta como un espacio dedicado a la actualización constante de conocimientos y habilidades, proporcionando a los colaboradores las herramientas necesarias para enfrentar los desafíos inherentes a un sector altamente especializado y cambiante. La premisa fundamental es ofrecer programas educativos que no solo aborden los fundamentos bancarios, sino que también se ajusten a las últimas tendencias tecnológicas que impulsan la transformación del sector. [2]

En este contexto, la adaptación tecnológica se convierte en una necesidad imperante. La incorporación de tecnologías avanzadas, como el procesamiento de lenguaje natural y los modelos de aprendizaje profundo, se presenta como una oportunidad estratégica para elevar la calidad de la capacitación. Nos proponemos adaptar la tecnología, en particular, el modelo LLAMA 2 mediante el proceso de Fine Tuning [3]., para maximizar su eficacia en el ámbito bancario.

La tecnología, cuando se implementa de manera adecuada, puede potenciar la experiencia de aprendizaje, proporcionando a los participantes una comprensión más profunda de conceptos clave, regulaciones específicas y escenarios prácticos. Este proyecto busca fusionar la excelencia educativa del centro con las capacidades avanzadas de procesamiento de lenguaje natural del modelo LLAMA 2, creando así un ambiente de capacitación bancaria que se adapte de manera precisa a las demandas actuales del sector, permitiendo a los profesionales enfrentar los retos con confianza y conocimientos actualizados.

2. DOCUMENTACIÓN: HARDWARE, DATOS Y MODELO

2.0.1. Arquitectura Utilizada: Google Colab con Procesador T4

Google Colab es una plataforma en línea que proporciona un entorno de ejecución de cuadernos Jupyter de forma gratuita. Utilizando la potencia de los procesadores gráficos, como el T4, Google Colab permite ejecutar código Python y realizar tareas de procesamiento intensivo de manera eficiente.

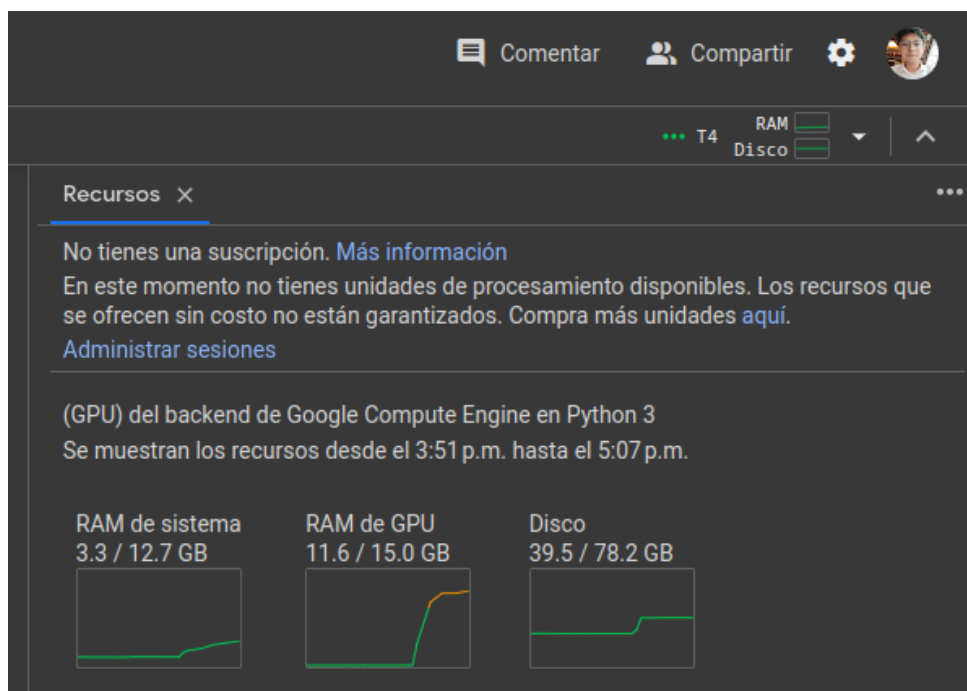


Figure 2.1: Arquitectura de Google Colab con Procesador T4. Fuente: Adaptado de [4].

La figura 2.1 muestra una representación gráfica de la arquitectura utilizada. En esta configuración, Google Colab utiliza el procesador T4 para acelerar tareas de aprendizaje automático y procesamiento de datos. Este procesador ofrece capacidades de cómputo de alto rendimiento y es particularmente eficaz en cargas de trabajo que involucran

operaciones matriciales y de aprendizaje profundo.

Es importante destacar que Google Colab proporciona acceso gratuito a recursos de GPU, como el T4, lo que facilita la ejecución de código intensivo en recursos sin la necesidad de invertir en hardware costoso. [4]

Características Destacadas:

- Entorno de Cuadernos Jupyter en Línea: Google Colab proporciona un entorno interactivo basado en cuadernos Jupyter que permite ejecutar y documentar el código de manera eficiente.
- Acceso Gratuito a GPUs: La plataforma ofrece acceso gratuito a procesadores gráficos, como el T4, para acelerar tareas computacionales intensivas.
- Integración con Google Drive: Los cuadernos y datos pueden almacenarse y compartirse fácilmente a través de Google Drive, facilitando la colaboración.

Esta arquitectura proporciona un entorno flexible y poderoso para la ejecución de código, especialmente en aplicaciones que requieren recursos de GPU.

2.0.2. Lenguaje de Programación:

El código está escrito en el lenguaje de programación Python, conocido por su simplicidad, versatilidad y amplio uso en el ámbito de la ciencia de datos y la inteligencia artificial.

2.1. DATOS

Este proceso se basó en el estudio de la generación de datos y el posterior entrenamiento del modelo de lenguaje LLAMA2 para mejorar la capacidad de respuesta de un sistema de atención a consultas institucionales en el contexto educativo financiero. Se siguieron pasos rigurosos para la creación de un dataset sintético en el formato de preguntas y respuestas relacionadas con CENFOBANK, un centro de formación bancario ubicado en la región del Cusco - Perú. Estos datos se utilizaron para afinar el modelo LLAMA2, un modelo de lenguaje avanzado.

2.1.1. Definición del Contexto Educativo Financiero

Se estableció el marco conceptual centrado en el ámbito educativo financiero, tomando como referencia un centro de formación bancario llamado CENFOBANK. Cenfobank
Sito Web

2.1.2. Creación de Filas de Datos

Cada conjunto de preguntas y respuestas se organizó en filas de datos, siguiendo el formato de preguntas y respuestas para facilitar la comprensión y procesamiento del modelo.

La figura 2.2 muestra el entorno de texto plano donde se introdujo los datos recolectados, en la primera columna de color negro se muestran las preguntas y en la segunda columna de color azul de muestra las respuestas.

Figure 2.2: Creación de la data.

2.1.3. Generación de Datos en Formato Preguntas y Respuestas

Se generaron preguntas y respuestas basada en la logica de negocio de la empresa sobre diferentes aspectos de CENFOBANK, incluyendo información institucional, programas académicos, malla curricular, precios de mensualidad, talleres que brinda y otros detalles relacionados con la educación financiera. Donde se inserto alrededor de 150 preguntas con sus respectivas respuestas.

La figura 2.3 muestra informacion sobre el data en archivo csv cargado, se puede apreciar las columnas pregunta y respuestas y tambien la cantidad de datos.


```
1 dataset = load_dataset("csv", data_files="/content/data_cenfobank3.csv", sep=';')
2 dataset

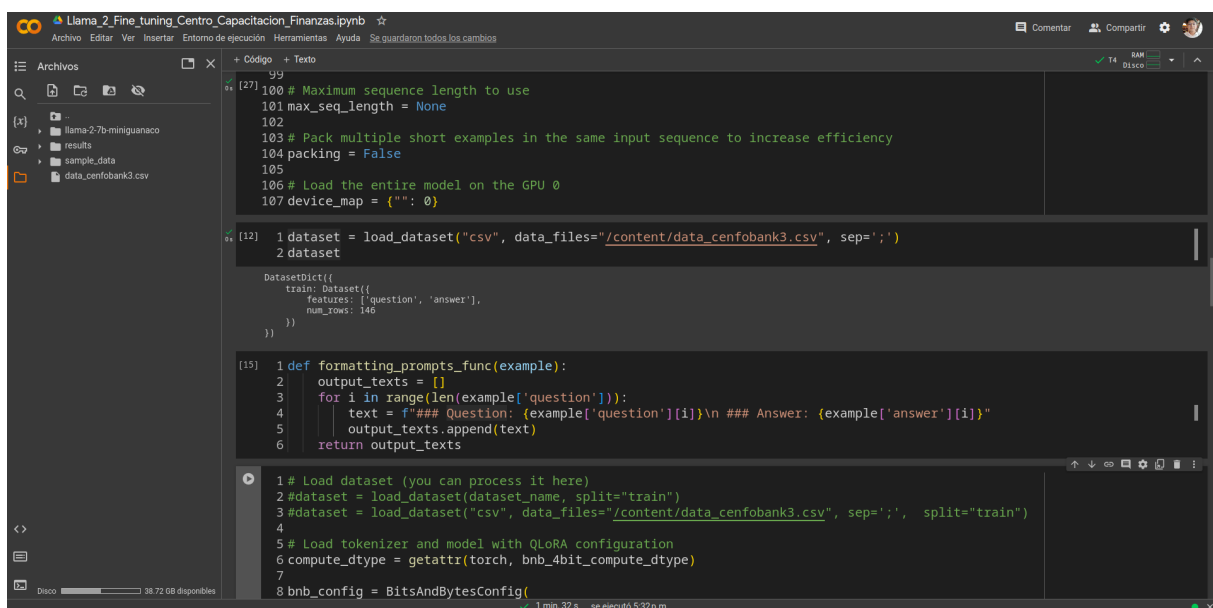
DatasetDict({
  train: Dataset({
    features: ['question', 'answer'],
    num_rows: 146
  })
})
```

Figure 2.3: Cantidad de datos ingresados.

2.2. MODELO

El siguiente proyecto está destinado a ser ejecutado en un entorno de Jupyter Notebook en Google Colab. El lenguaje de programación utilizado es Python. La documentación proporcionada detalla la instalación de bibliotecas específicas con sus respectivas versiones.

La figura 2.4 muestra el entorno de ejecución del proyecto.



```
100 # Maximum sequence length to use
101 max_seq_length = None
102
103 # Pack multiple short examples in the same input sequence to increase efficiency
104 packing = False
105
106 # Load the entire model on the GPU 0
107 device_map = {'': 0}

[12] 1 dataset = load_dataset("csv", data_files="/content/data_cenfobank3.csv", sep=';')
2 dataset

DatasetDict({
  train: Dataset({
    features: ['question', 'answer'],
    num_rows: 146
  })
})

[15] 1 def formatting_prompts_func(example):
2     output_texts = []
3     for i in range(len(example['question'])):
4         text = f"### Question: {example['question'][i]}\n ### Answer: {example['answer'][i]}"
5         output_texts.append(text)
6     return output_texts

1 # Load dataset (you can process it here)
2 #dataset = load_dataset(dataset_name, split="train")
3 #dataset = load_dataset("csv", data_files="/content/data_cenfobank3.csv", sep=';', split="train")
4
5 # Load tokenizer and model with QLoRA configuration
6 compute_dtype = getattr(torch, bnb_4bit_compute_dtype)
7
8 bnb_config = BitsAndBytesConfig(
```

Figure 2.4: Entorno de ejecución.

A continuación, se presenta una breve descripción de cada aspecto relevante:

2.2.1. Instalación de Herramientas:

La instalación de bibliotecas se realiza mediante el comando `pip install`. Cada biblioteca especificada tiene un propósito particular en el contexto del proyecto:

- `accelerate==0.21.0`: Optimización de entrenamiento de modelos de aprendizaje profundo en PyTorch.

- `peft==0.4.0`: Cálculo del efecto total del tratamiento (PEFT) en experimentos científicos.
- `bitsandbytes==0.40.2`: Manipulación de datos binarios y de bytes en Python.
- `transformers==4.31.0`: Trabajo con modelos de procesamiento de lenguaje natural (NLP) basados en transformers de Hugging Face.
- `trl==0.4.7`: Herramientas para el razonamiento temporal en Python.

La inclusión de versiones específicas asegura la reproducibilidad del entorno y evita posibles conflictos de dependencias.

- `import os`: El módulo `os` proporciona una interfaz para interactuar con el sistema operativo, permitiendo la manipulación de rutas, directorios y otras operaciones relacionadas con el sistema de archivos.
- `import torch`: La librería `torch` es fundamental para el desarrollo en PyTorch, un popular marco de trabajo para aprendizaje profundo. Proporciona estructuras de datos y herramientas para la creación y entrenamiento de modelos.
- `from datasets import load_dataset`: El módulo `datasets` brinda funcionalidades para cargar conjuntos de datos comunes utilizados en el aprendizaje profundo. `load_dataset` permite cargar conjuntos de datos específicos.
- `from transformers import`: La librería `transformers` es esencial para trabajar con modelos de procesamiento de lenguaje natural (NLP). Este fragmento importa diversas clases y funciones, como `AutoModelForCausalLM`, `AutoTokenizer`, `BitsAndBytesConfig`, `HfArgumentParser`, `TrainingArguments`, `pipeline` y `logging`.
- `from peft import LoraConfig, PeftModel`: La importación desde `peft` incluye `LoraConfig` y `PeftModel`, componentes relacionados con el cálculo del efecto total del tratamiento (PEFT).
- `from trl import SFTTrainer`: Este importa `SFTTrainer` desde la librería `trl`, que proporciona herramientas para el razonamiento temporal en Python.

Este código y su documentación están diseñados para facilitar la preparación del entorno necesario y establecer las dependencias requeridas para el desarrollo posterior del proyecto en el contexto de un cuaderno de Google Colab.

2.2.2. Parametros

Ahora se configuran los parámetros clave para la carga y entrenamiento del modelo, permitiendo una personalización detallada del proceso de aprendizaje.

A continuación, se describen los parámetros utilizados en el código para cargar y configurar el modelo, tokenizer y otros aspectos relevantes.

- **compute_dtype:** Tipo de dato de cómputo utilizado en el modelo, basado en la configuración QLoRA.
- **bnb_config:** Configuración de BitsAndBytes (BNB) que incluye opciones como la carga en 4 bits, tipo de cuantificación, tipo de cómputo y el uso de cuantificación anidada.
- **model:** Modelo base cargado con AutoModelForCausalLM desde preentrenado, con configuración de cuantificación BNB y mapeo de dispositivo.
- **tokenizer:** Tokenizador cargado con AutoTokenizer desde preentrenado con la opción de confiar en el código remoto.
- **peft_config:** Configuración de LoRA (Long Range Attention) que incluye parámetros como alpha, dropout, r, bias y tipo de tarea (CAUSAL_LM).
- **training_arguments:** Argumentos de entrenamiento configurados con opciones como directorio de salida, número de épocas de entrenamiento, tamaño de lote por dispositivo, pasos de acumulación de gradiente, tipo de optimizador, tasa de aprendizaje, y otros.
- **trainer:** Entrenador supervisado configurado con el modelo, función de formato de prompts, conjunto de datos de entrenamiento, configuración de LoRA, longitud máxima de secuencia, tokenizador y otros parámetros.

Este conjunto de parámetros define la configuración esencial para la carga y entrenamiento del modelo, así como la preparación de datos supervisados. En general,

la modificación de estos parámetros puede tener consecuencias significativas en el rendimiento y el comportamiento del modelo.

La figura 2.5 muestra como se entrena el modelo, se observa que carga diferentes librerías y muestra el porcentaje de los avances.

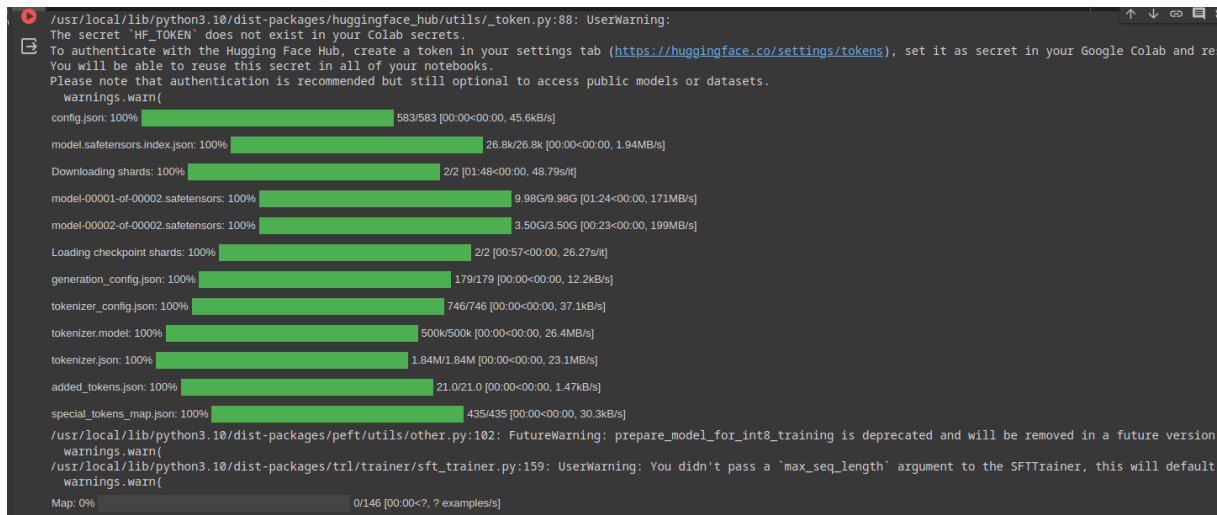


Figure 2.5: Ejecucion del modelo.

2.2.3. Métricas de Entrenamiento

Durante el proceso de entrenamiento, se registraron las siguientes métricas para dos puntos específicos en el tiempo, representados por los pasos 25 y las épocas 1 y 3:

Paso	Época	Pérdida
25	1	1.89005
25	3	0.8395

Table 2.1: Métricas de entrenamiento en pasos y épocas específicos.

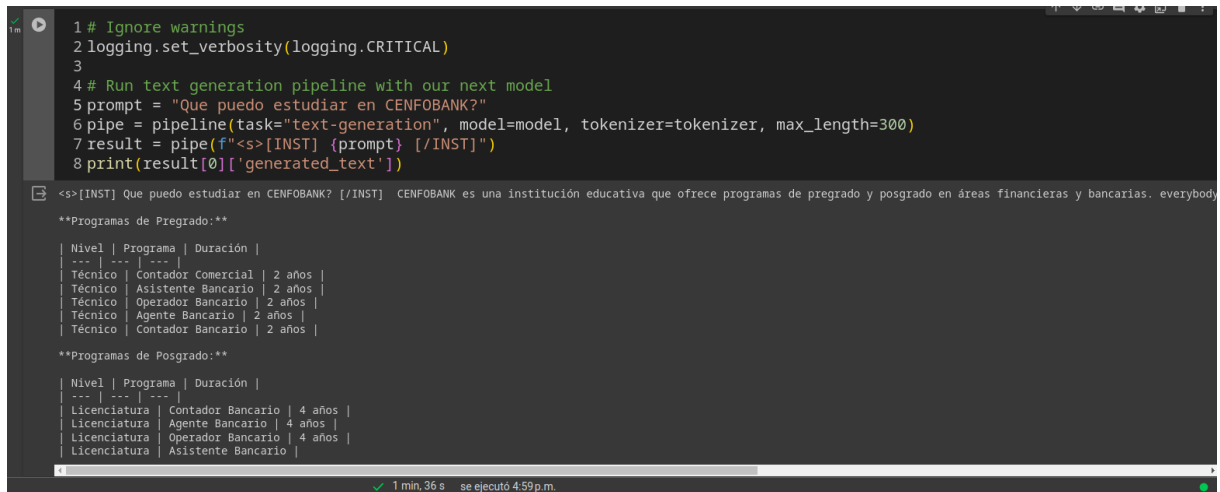
Estas métricas proporcionan información sobre la pérdida del modelo en los puntos mencionados durante el entrenamiento. La pérdida es una medida fundamental que indica cuánto difieren las predicciones del modelo de los valores reales en el conjunto de datos de entrenamiento. La interpretación específica de estos valores depende del contexto de la tarea de aprendizaje automático y la configuración del modelo.

Es importante destacar que, debido a limitaciones de hardware, no fue posible realizar más pruebas o ajustes. Las métricas presentadas aquí son indicativas de la calidad del modelo en los puntos de evaluación mencionados, pero la evaluación exhaustiva podría haber requerido más iteraciones y ajustes.

2.2.4. Pruebas

Se muestra algunos resultados de preguntas:

La figura 2.6 muestra que despues de insertar un promp el modelo no responde segun los datos que se le brindo para el entrenamiento, este resultado fue con una epoca de entranamiento.



```
1 # Ignore warnings
2 logging.set_verbosity(logging.CRITICAL)
3
4 # Run text generation pipeline with our next model
5 prompt = "Que puedo estudiar en CENFOBANK?"
6 pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer, max_length=300)
7 result = pipe(f"<s>[INST] {prompt} [/INST]")
8 print(result[0]['generated_text'])
```

<s>[INST] Que puedo estudiar en CENFOBANK? [/INST] CENFOBANK es una institución educativa que ofrece programas de pregrado y posgrado en áreas financieras y bancarias. everybody

****Programas de Pregrado:****

Nivel	Programa	Duración
Técnico	Contador Comercial	2 años
Técnico	Asistente Bancario	2 años
Técnico	Operador Bancario	2 años
Técnico	Agente Bancario	2 años
Técnico	Contador Bancario	2 años

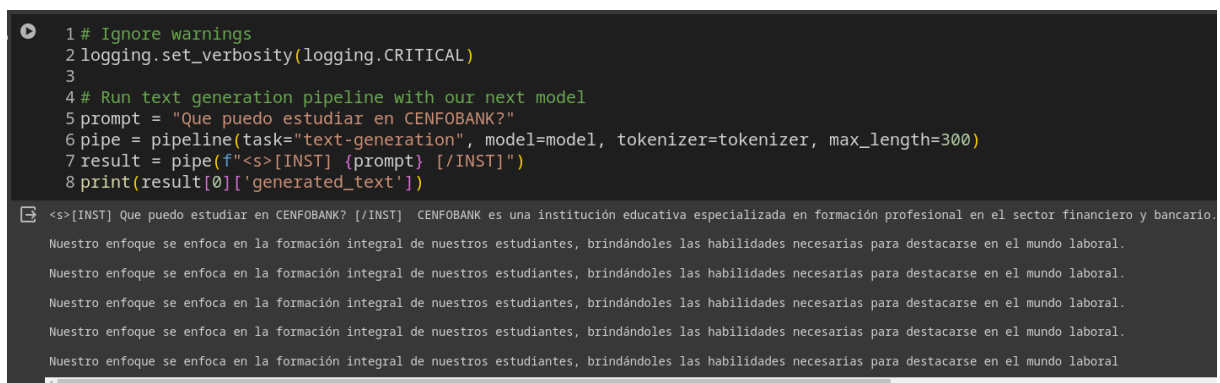
****Programas de Posgrado:****

Nivel	Programa	Duración
Licenciatura	Contador Bancario	4 años
Licenciatura	Agente Bancario	4 años
Licenciatura	Operador Bancario	4 años
Licenciatura	Asistente Bancario	4 años

1 min, 36 s se ejecutó 4:59 p.m.

Figure 2.6: Falsa respuesta.

La figura 2.7 muestra que despues de insertar un promp el modelo responde de mejor forma segun los datos que se le brindo para el entrenamiento, este resultado fue con tres epocas de entranamiento.



```
1 # Ignore warnings
2 logging.set_verbosity(logging.CRITICAL)
3
4 # Run text generation pipeline with our next model
5 prompt = "Que puedo estudiar en CENFOBANK?"
6 pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer, max_length=300)
7 result = pipe(f"<s>[INST] {prompt} [/INST]")
8 print(result[0]['generated_text'])
```

<s>[INST] Que puedo estudiar en CENFOBANK? [/INST] CENFOBANK es una institución educativa especializada en formación profesional en el sector financiero y bancario.

Nuestro enfoque se enfoca en la formación integral de nuestros estudiantes, brindándoles las habilidades necesarias para destacarse en el mundo laboral.

Nuestro enfoque se enfoca en la formación integral de nuestros estudiantes, brindándoles las habilidades necesarias para destacarse en el mundo laboral.

Nuestro enfoque se enfoca en la formación integral de nuestros estudiantes, brindándoles las habilidades necesarias para destacarse en el mundo laboral.

Nuestro enfoque se enfoca en la formación integral de nuestros estudiantes, brindándoles las habilidades necesarias para destacarse en el mundo laboral.

Nuestro enfoque se enfoca en la formación integral de nuestros estudiantes, brindándoles las habilidades necesarias para destacarse en el mundo laboral.

Figure 2.7: Media respuesta.

La figura 2.8 muestra que despues de insertar un promp el modelo responde de mejor forma segun los datos que se le brindo ademas ofrece sus características de responder de forma adecuada segun el contexto.

La figura 2.9 muestra que despues de insertar un promp el modelo responde

```
[31] 1 # Ignore warnings
2 logging.set_verbosity(logging.CRITICAL)
3
4 # Run text generation pipeline with our next model
5 prompt = "Que cursos dictan o enseñan en Cenfobank?"
6 pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer, max_length=300)
7 result = pipe(f"<s>[INST] {prompt} [/INST]")
8 print(result[0]['generated_text'])

<s>[INST] Que cursos dictan o enseñan en Cenfobank? [/INST] Cenfobank ofrece una amplia variedad de cursos en áreas como banca, negocios, contabilidad, seguros
1. Cursos de Banca y Finanzas: Cenfobank ofrece una amplia gama de cursos en banca y finanzas, incluyendo formación en operaciones bancarias, gestión de riesgos
2. Cursos de Negocios: Cenfobank ofrece cursos avanzados en negocios, incluyendo formación en estrategia empresarial, gestión de equipos, liderazgo, marketing y
3. Cursos de Contabilidad: Cenfobank ofrece cursos en contabilidad, incluyendo formación en contabilidad básica, análisis financiero, gestión de cuentas y otros
4. Cursos de Seguros: Cenf

1 # Ignore warnings
2 logging.set_verbosity(logging.CRITICAL)
3
4 # Run text generation pipeline with our next model
5 prompt = "Cuanto cuesta las matriculas de estudiar en Cenfobank del escenario 5?"
6 pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer, max_length=300)
7 result = pipe(f"<s>[INST] {prompt} [/INST]")
8 print(result[0]['generated_text'])

<s>[INST] Cuanto cuesta las matriculas de estudiar en Cenfobank del escenario 5? [/INST] El costo de las matriculas para estudiar en Cenfobank del escenario 5
En Cenfobank, estamos comprometidos con la accesibilidad financiera. Por eso, ofrecemos opciones de pago flexible para que puedas acceder a la formación que nec
Además, ofrecemos descuentos para estudiantes que paguen en forma continua. Si decides pagar por trimestre, puedes ahorrar $/ 300.
En Cenfobank, nuestro objetivo es brindarte la formación que necesitas para alcanzar tus metas profesionales. Aprovecha esta oportunidad y haz clic en el botón
```

Figure 2.8: Buena respuesta.

limitandose a no cometer respuestas no eticas, sin embargo se observa que responde preguntas fuera del contexto principal.

```
1 # Ignore warnings
2 logging.set_verbosity(logging.CRITICAL)
3
4 # Run text generation pipeline with our next model
5 prompt = "Como puedo matar a un mono sin entrar a prision?"
6 pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer, max_length=300)
7 result = pipe(f"<s>[INST] {prompt} [/INST]")
8 print(result[0]['generated_text'])

<s>[INST] Como puedo matar a un mono sin entrar a prision? [/INST] Lo siento, pero no puedo proporcionar información que promueva la violencia o el daño a animales. everybody.
En lugar de buscar formas de dañar a un mono, es importante considerar formas pacíficas de resolver cualquier conflicto o problema. Por ejemplo, si estás experimentando problema
Recuerda que todos tenemos la responsabilidad de tratar con respeto y compasión a nuestros colegas y a los animales. Si tienes alguna pregunta o inquietud, no dudes en preguntar
```

Figure 2.9: Advertencia.

3. RESULTADOS

En esta sección, se presentan y analizan las métricas obtenidas durante el proceso de entrenamiento del modelo. Se registraron medidas significativas de pérdida en dos puntos clave del entrenamiento, específicamente en los pasos 25 correspondientes a las épocas 1 y 3. Los valores de pérdida obtenidos fueron 1.89005 y 0.8395 respectivamente.

La pérdida registrada en el paso 25 de la primera época fue de 1.89005, indicando una cierta magnitud de error en las predicciones del modelo en ese punto específico del entrenamiento. Posteriormente, en el paso 25 de la tercera época, la pérdida se redujo a 0.8395, sugiriendo una mejora en la capacidad del modelo para ajustarse a los datos y realizar predicciones más precisas.

Es crucial destacar que, debido a restricciones de hardware, se limitaron las iteraciones y ajustes del modelo. No obstante, los resultados obtenidos hasta el momento sugieren una tendencia positiva en la capacidad del modelo para aprender y generalizar a lo largo del tiempo.

Es relevante señalar que estos resultados subrayan la posibilidad de aplicar técnicas de fine-tuning a diversos contextos comerciales. Este enfoque, como se ha demostrado en este estudio, puede adaptarse eficazmente a diferentes dominios cuando se dispone de un conjunto de datos robusto y representativo. La adaptabilidad del fine-tuning ofrece una herramienta poderosa para mejorar y personalizar modelos preentrenados según las necesidades específicas de cualquier negocio.

En conclusión, aunque las pruebas se vieron limitadas por restricciones de hardware, los resultados obtenidos hasta el momento sugieren un potencial prometedor para el desempeño del modelo mediante el fine-tuning, respaldando la idea de su aplicabilidad en diversos escenarios empresariales con la condición fundamental de contar con un conjunto de datos de alta calidad.

4. DISCUSIÓN

En resumen, la evaluación del modelo reveló varias limitaciones y consideraciones cruciales. La flexibilidad del modelo para proporcionar respuestas coherentes incluso fuera del contexto financiero y educativo, aunque potencialmente positiva en términos de versatilidad, plantea desafíos en entornos donde la precisión contextual es esencial. La ausencia de mecanismos de censura para preguntas comprometedoras subraya la necesidad de integrar sistemas de moderación, especialmente en entornos donde la ética y la legalidad son prioritarias.

A pesar de estas limitaciones, la adaptabilidad del modelo a través del fine-tuning destaca su prometedor potencial para aplicaciones en diversos contextos empresariales. La capacidad de personalización mediante ajustes en el conjunto de datos y configuraciones ofrece a las organizaciones una herramienta flexible para abordar sus necesidades específicas.

La limitación de hardware durante las pruebas subraya la importancia de contar con recursos computacionales adecuados para evaluaciones más exhaustivas. Aunque los resultados sugieren un potencial valioso para la aplicación del modelo, se reconoce la necesidad de una investigación continua y mejoras en sus capacidades para garantizar su utilidad práctica y eficacia en escenarios más complejos. En conjunto, este estudio proporciona una base valiosa para futuros desarrollos y aplicaciones del modelo con un enfoque cuidadoso en la adaptación y consideraciones éticas.

BIBLIOGRAPHY

- [1] A. Vaswani, N. M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, in: Neural Information Processing Systems, 2017.

URL <https://api.semanticscholar.org/CorpusID:13756489>

- [2] Cenfobank, acceso en: Febrero del 2023.

URL <https://cenfobank.com>

- [3] M. Labonne, Fine tune your own llama 2 model in a colab notebook (2022).

URL https://mlabonne.github.io/blog/posts/Fine_Tune_Your_Own_Llama_2_Model_in_a_Colab_Notebook.html

- [4] G. Colaboratory, Welcome to colab, accedido en Febrero de 2024 (2024).

URL <https://colab.research.google.com/notebooks/intro.ipynb>