GUÍA DE INSTALACIÓN DE UNLIMITED OCR (BAIDU)

El modelo gratis que lee un PDF escaneado completo de una sola vez y lo convierte en texto. Corre en tu propia computadora.

QUÉ ES

Unlimited OCR es un modelo de código abierto de Baidu con licencia MIT (uso comercial gratis). Tiene 3 mil millones de parámetros y convierte documentos escaneados o PDFs en texto estructurado con tablas y fórmulas.

Su diferencia: puede procesar varias páginas en una sola pasada, en lugar de leer cada página por separado. Su ventana de contexto es de 32,768 tokens.

Repositorio oficial: github.com/baidu/Unlimited-OCR. Modelo: huggingface.co/baidu/Unlimited-OCR.

LO QUE DEBES SABER ANTES

Necesitas una tarjeta gráfica NVIDIA. La documentación oficial lo prueba sobre GPUs NVIDIA con CUDA. No es una app que se instala y listo en cualquier laptop.

Cuántas páginas caben en una pasada depende del contenido. Un análisis externo sitúa el límite práctico en unas 40 páginas y la mejor precisión hasta unas 20. Para documentos más largos, divídelos.

Es un modelo nuevo y buena parte de las cifras de rendimiento vienen de terceros. Pruébalo con tus propios documentos antes de confiar en él.

Corre en local: tus documentos no salen de tu equipo al procesarlos. La descarga inicial del modelo sí requiere internet.

1 Revisa si tu equipo puede correrlo

Requisito	Qué necesitas
Tarjeta gráfica	NVIDIA con CUDA. Si no tienes una, mira la opción de rentar una GPU más abajo.
Memoria de video (VRAM)	La receta oficial de vLLM indica que 8 GB bastan para el modelo en BF16. Otra fuente recomienda 12 GB para ir cómodo y 24 GB para documentos largos.
Software probado	Python 3.12.3 y CUDA 12.9 (Transformers). Docker, si usas el servidor vLLM.
Espacio en disco	Cerca de 7 GB solo para los pesos del modelo.
Internet	Solo para la primera descarga del modelo y de las librerías.

¿NO TIENES GPU NVIDIA?

Puedes rentar una GPU por hora en proveedores de nube y correr ahí exactamente los mismos pasos. Ojo: en ese caso los documentos sí salen de tu computadora hacia el servidor que rentas, así que no subas documentos confidenciales.

2 Elige tu camino

Camino	Para quién
A. Que una IA lo instale por ti	Si no programas. Más rápido para empezar.
B. Instalación manual (Transformers)	Si quieres procesar PDFs con un script propio.
C. Servidor local (vLLM con Docker)	Si lo vas a conectar a un sistema o procesar muchos documentos.

A Camino A: que una IA lo instale por ti

Si usas Claude Code o una herramienta similar con acceso a tu computadora, pega este prompt:

PROMPT PARA CLAUDE CODE

```
Revisa si mi equipo tiene una GPU NVIDIA con CUDA y cuánta VRAM tiene. Si es suficiente, clona https://github.com/baidu/Unlimited-OCR, lee el README y sigue la instalación con Transformers en un entorno virtual con Python 3.12. Luego procesa este PDF: [RUTA DE TU PDF] con la función de varias páginas (infer_multi) y guarda el resultado en una carpeta llamada salida. Antes de instalar, dime qué vas a hacer y espera mi confirmación. Al final, resúmeme lo que quedó instalado y dónde está el resultado.
```

Si no tienes GPU, pega esto en un chat con tu IA de confianza:

PROMPT SI NO SABES TU HARDWARE

```
Quiero correr Unlimited OCR de Baidu en mi computadora. Mi equipo es: [MODELO, SISTEMA OPERATIVO, TARJETA GRÁFICA]. Dime si puedo hacerlo, qué VRAM necesito y qué alternativa tengo si no me alcanza. Ve la documentación oficial en github.com/baidu/Unlimited-OCR.
```

B Camino B: instalación manual con Transformers

Paso 1. Crea un entorno virtual con Python 3.12 e instala estas librerías (versiones probadas por Baidu):

TERMINAL

```
pip install torch==2.10.0 torchvision==0.25.0 transformers==4.57.1 Pillow==12.1.1 matplotlib==3.10.8 einops==0.8.2 addict==2.4.0 easydict==1.13 pymupdf==1.27.2.2 psutil==7.2.2
```

- Para una sola imagen (una foto de un recibo, por ejemplo), el README usa model.infer con el prompt <image>document parsing.

- Para varias páginas o PDFs, el README indica que solo funciona el modo base (image_size=1024), que es el que usa el script del siguiente paso.

Paso 2. Guarda este script como ocr_pdf.py, cambia la ruta de tu PDF y ejecútalo con python ocr_pdf.py. Está armado con el código del README oficial:

ocr_pdf.py

```
import os, tempfile
import torch
import fitz  # PyMuPDF
from transformers import AutoModel, AutoTokenizer

PDF = 'tu_documento.pdf'   # cambia esta ruta
SALIDA = 'salida'          # carpeta de resultados

model_name = 'baidu/Unlimited-OCR'
tokenizer = AutoTokenizer.from_pretrained(model_name, trust_remote_code=True)
model = AutoModel.from_pretrained(
    model_name,
    trust_remote_code=True,
    use_safetensors=True,
    torch_dtype=torch.bfloat16,
)
model = model.eval().cuda()

def pdf_to_images(pdf_path, dpi=300):
    doc = fitz.open(pdf_path)
    tmp_dir = tempfile.mkdtemp(prefix='pdf_ocr_')
    mat = fitz.Matrix(dpi / 72, dpi / 72)
    paths = []
    for i, page in enumerate(doc):
        out = os.path.join(tmp_dir, f'page_{i+1:04d}.png')
        page.get_pixmap(matrix=mat).save(out)
        paths.append(out)
    doc.close()
    return paths

os.makedirs(SALIDA, exist_ok=True)
model.infer_multi(
    tokenizer,
    prompt='<image>Multi page parsing.',
    image_files=pdf_to_images(PDF, dpi=300),
    output_path=SALIDA,
    image_size=1024,
    max_length=32768,
    no_repeat_ngram_size=35, ngram_window=1024,
    save_results=True,
)
```

C Camino C: servidor local con vLLM y Docker

Levanta un servidor compatible con la API de OpenAI en tu propia máquina, para conectarlo a tus sistemas:

TERMINAL

```
docker run --rm --gpus all --network host --ipc host \
  vllm/vllm-openai:unlimited-ocr \
  baidu/Unlimited-OCR \
  --trust-remote-code \
  --logits_processors vllm.model_executor.models.unlimited_ocr:NGramPerReqLogitsProcessor
\
  --no-enable-prefix-caching \
  --mm-processor-cache-gb 0
```

- El procesador de lógica (logits_processors) es obligatorio: sin él, los documentos largos pueden entrar en bucles.
- El prompt debe empezar con la etiqueta <image>, o la respuesta llega vacía.
- Para varias páginas o PDFs, la receta indica usar una ventana de 1024 y convertir cada página a imagen antes de enviarla. La receta no trae un ejemplo de PDF, así que pruébalo con un documento corto primero.
- Con GPUs Hopper (H100), el repositorio ofrece una imagen distinta: vllm/vllm-openai:unlimited-ocr-cu129.

D Revisa y limpia el resultado

La salida trae marcadores que indican el tipo de bloque y sus coordenadas en la página. Para usar el texto en un buscador o en un sistema RAG, pídele a una IA el script de limpieza con una muestra real de tu salida:

PROMPT DE LIMPIEZA

```
Te pego una muestra de la salida de Unlimited OCR: [PEGA 30 LÍNEAS DE
TU SALIDA]. Escribe un script en Python que quite los marcadores de
coordenadas y deje texto en markdown limpio, conservando las tablas y
el orden de lectura. Dime cómo comprobar que no perdió contenido.
```

PRUEBA ANTES DE CONFIAR

Con documentos de tu negocio, haz esta prueba antes de usarlo en serio:

☐	Procesa un documento corto de 2 a 3 páginas y compara el texto con el original.
☐	Prueba uno con tablas y revisa que las cifras coincidan una por una.
☐	Prueba un escaneo de mala calidad (torcido, con sombras o manchas).
☐	Prueba uno largo, de 20 páginas o más, y revisa que no se repitan ni salten párrafos.
☐	Revisa siempre cifras, fechas y nombres antes de usarlos en contratos o cuentas.
☐	Si falla con un documento largo, divídelo en bloques de unas 20 páginas.

QUÉ SIGUE DESPUÉS DE LEER EL DOCUMENTO

Convertir un PDF en texto es solo el primer paso. El valor aparece cuando ese texto se clasifica, se guarda en el lugar correcto y alimenta tus procesos, por ejemplo contratos, facturas o expedientes que se leen y se archivan solos. Montar ese flujo completo en tu negocio es justo lo que hacemos en Vianta AI.

¿Quieres que tus documentos se lean y se archiven solos?

Escríbenos directamente a nuestro agente de WhatsApp
y agendemos una llamada.

+52 81 2188 4401