multi versica ACADEMY

Sistemas LLM:

FINE-TUNING PARA RAG CON LLaMa 3.1

Optimización Avanzada de Modelos de Lenguaje para Aplicaciones RAG

IA Avanzada

Procesamiento de Datos

Optimización

Deployment

Sector público LATAM

Nivel Medio Alto

Enfoque técnico

multiversica ACADEMY

MÓDULO 3:
DEPLOYMENT EN CPU CON OLLAMA

Estructura del Módulo

3.1 Introducción a Ollama

Comprenderemos qué es Ollama, sus ventajas y su arquitectura para el deployment de LLMs en CPU.

3.2 Instalación y Configuración

Instalaremos Ollama en diferentes sistemas operativos y configuraremos Llama 3.1 para su ejecución local.

3.3 API REST e Integración

Exploraremos cómo utilizar la API de Ollama para integrar Llama 3.1 en aplicaciones web y otros sistemas.

En este módulo adquirirás todas las habilidades necesarias para desplegar e integrar Llama 3.1 en entornos con recursos limitados.

Objetivos del Módulo

multi versica ACADEMY

1 **Conocimientos técnicos**

Entender qué es Ollama y su arquitectura interna para el deployment de modelos de lenguaje en CPU.

2 **Habilidades prácticas**

Instalar, configurar y ejecutar Llama 3.1 utilizando Ollama en diferentes sistemas operativos.

3 **Capacidades de integración**

Utilizar la API REST de Ollama para integrar Llama 3.1 en aplicaciones web y otros sistemas.

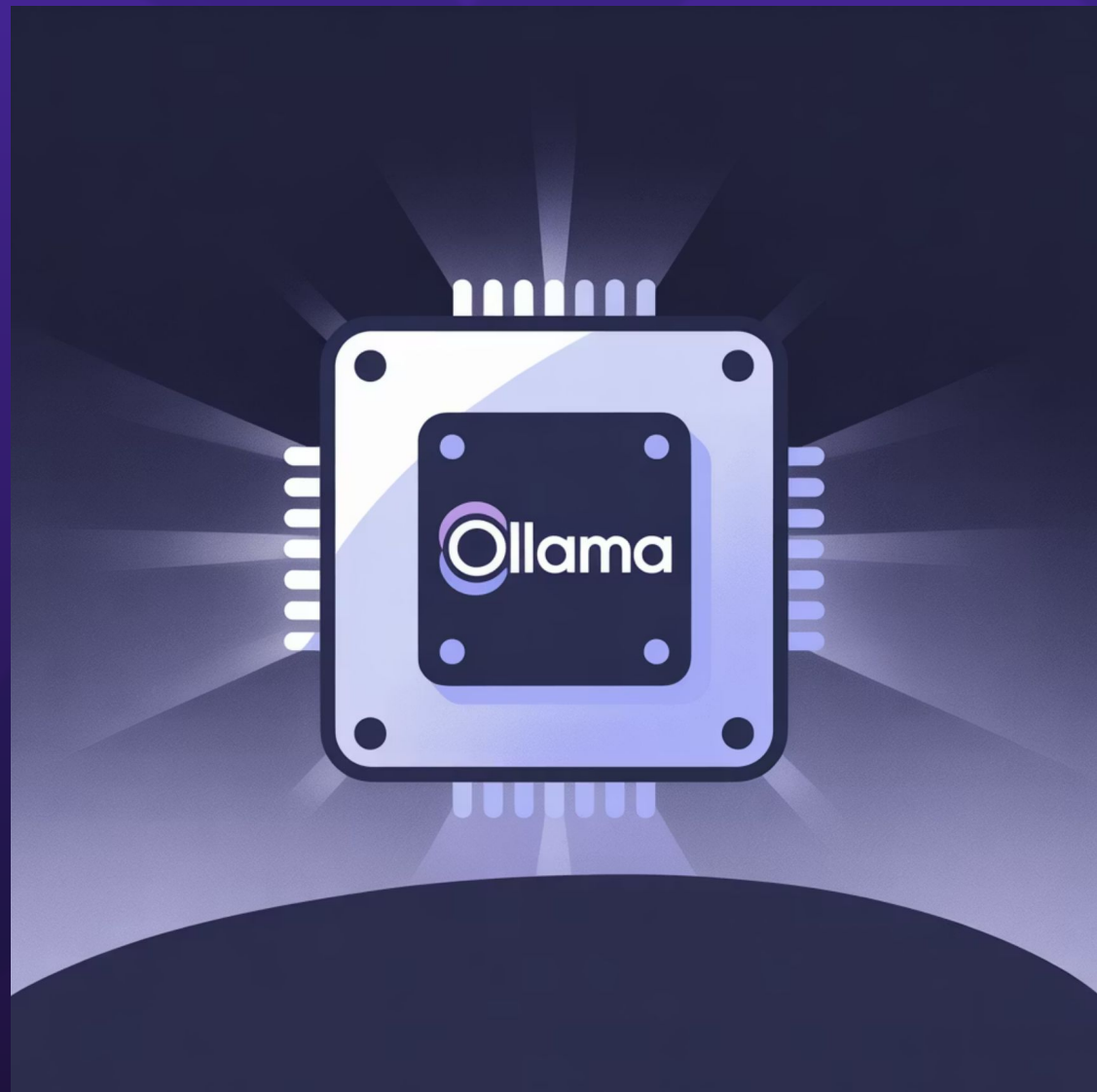
Al finalizar este módulo, serás capaz de implementar Llama 3.1 en entornos con recursos computacionales limitados y construir aplicaciones que aprovechen sus capacidades.

3.1 Introducción a Ollama

¿Qué es Ollama?

- Ollama es una herramienta de código abierto diseñada específicamente para ejecutar modelos de lenguaje de gran tamaño (LLMs) localmente en CPU, sin necesidad de hardware especializado.

- Permite desplegar modelos como Llama 3.1 de manera eficiente en ordenadores personales y servidores con recursos limitados, democratizando el acceso a estas tecnologías.



Ventajas principales de Ollama

1

Facilidad de uso

Interfaz de línea de comandos intuitiva y sencilla que permite desplegar modelos con comandos mínimos.

2

Optimización para CPU

Implementaciones altamente optimizadas que permiten la ejecución eficiente en CPUs estándar, sin requerir GPUs.

3

Gestión automatizada

Manejo automático de descarga, almacenamiento y carga de modelos, simplificando la administración.

4

API REST integrada

Incluye un servidor API REST listo para usar, facilitando la integración con aplicaciones existentes.

Estas características hacen de Ollama una opción ideal para desarrolladores, investigadores y entusiastas que desean experimentar con LLMs en entornos con recursos limitados.

Deployment: Transformers vs. Ollama

Mientras que Transformers ofrece más flexibilidad para modificaciones avanzadas, Ollama destaca por su simplicidad y optimización específica para CPUs, siendo ideal para quienes buscan una implementación rápida y eficiente.

¿Por qué usar Ollama para *deployment* en CPU?

multi versica ACADEMY

Optimización específica

Ollama está diseñado específicamente para maximizar el rendimiento en CPUs mediante optimizaciones a nivel de compilación y ejecución.

Gestión eficiente de memoria

Implementa técnicas avanzadas para la gestión de memoria, permitiendo ejecutar modelos grandes en hardware con limitaciones.

Menor sobrecarga

Su arquitectura ligera reduce la sobrecarga computacional, aprovechando mejor los recursos disponibles que otras soluciones más generales.

Estas ventajas hacen de Ollama la opción preferida para entornos donde la eficiencia en CPU es prioritaria.

Arquitectura de Ollama

Ollama implementa una arquitectura modular compuesta por tres componentes principales:

- Motor de inferencia optimizado para CPU que aprovecha instrucciones vectoriales y paralelismo
- Gestor de modelos para descarga, almacenamiento y carga eficiente
- Servidor API REST para comunicación e integración con aplicaciones externas

Esta arquitectura permite ejecutar modelos como Llama 3.1 de manera eficiente, incluso en hardware con recursos limitados.

Funcionamiento interno de Ollama

multi versica ACADEMY

Descarga y preparación

Ollama descarga modelos optimizados, los almacena localmente y los preprocesa para optimizar su carga.

Carga y cuantización

Carga eficiente del modelo en memoria, aplicando técnicas de cuantización para reducir requisitos.

Inferencia optimizada

Ejecución de inferencia mediante algoritmos optimizados para CPU utilizando instrucciones específicas.

Entrega de resultados

Procesamiento de salida y transmisión de resultados a través de CLI o API REST.

Este flujo de trabajo optimizado permite a Ollama ejecutar modelos complejos como Llama 3.1 con un rendimiento aceptable en CPU.

Comparación con otras herramientas de deployment

Característica	Ollama	HuggingFace	LangChain
Optimización para CPU	Alta	Media	Baja
Facilidad de uso	Muy alta	Media	Media
Personalización	Limitada	Alta	Muy alta
Integración con aplicaciones	API REST	Python, API	Python, API
Rendimiento en hardware limitado	Excelente	Bueno	Regular

Ollama destaca particularmente en entornos con recursos limitados y casos de uso que priorizan la simplicidad y el rendimiento.

Casos de uso ideales para Ollama

multi versica ACADEMY

Desarrollo y prototipado

Ideal para desarrolladores que necesitan iterar rápidamente sin depender de servicios en la nube.

Entornos con restricciones de privacidad

Perfecto para aplicaciones que requieren procesamiento local de datos sensibles sin conexión a internet.

Edge computing

Excelente para dispositivos con CPU limitada donde se necesita procesamiento de lenguaje natural.

Educación e investigación

Óptimo para estudiantes y académicos que necesitan experimentar con LLMs sin incurrir en costos de computación en la nube.

Estos escenarios aprovechan al máximo las ventajas de Ollama para el deployment de Llama 3.1 en entornos con CPU.

Limitaciones de Ollama

Principales limitaciones a considerar:

- Velocidad de inferencia más lenta comparada con GPU (5-20x)
- Capacidades limitadas de paralelización
- Opciones reducidas de personalización del modelo
- Menos flexibilidad para técnicas avanzadas como LoRA o QLoRA
- Consumo elevado de memoria RAM para modelos grandes

Requisitos mínimos recomendados

multi versica ACADEMY

8GB

RAM mínima

Para modelos pequeños como Llama 3.1 8B (16GB recomendado para mejor rendimiento)

4

Núcleos CPU

Mínimo recomendado para una experiencia aceptable (8+ para mejor rendimiento)

10GB

Espacio en disco

Para almacenar el modelo Llama 3.1 8B y sus dependencias

1-3

Tokens/segundo

Velocidad típica de generación con CPU estándar (varía según hardware)

Estos requisitos son orientativos y pueden variar según el modelo específico y la carga de trabajo.

3.2 Instalación y Configuración de Ollama

multi versica ACADEMY

Sistemas operativos compatibles

Windows

Compatible con Windows 10/11 de 64 bits. Requiere Windows Subsystem for Linux (WSL) 2 para algunas funcionalidades.

macOS

Compatible con macOS 12 (Monterey) o superior. Soporte nativo para Apple Silicon (M1/M2/M3) y x86.

Linux

Compatible con distribuciones modernas como Ubuntu 20.04+, Debian 11+, Fedora 35+ y sus derivados.

Ollama ofrece soporte multiplataforma, lo que facilita su adopción en diversos entornos de desarrollo y producción.

Instalación en Windows

Pasos para la instalación:

1. Habilitar Windows Subsystem for Linux 2 (WSL2): `wsl --install`
2. Descargar el instalador desde la página oficial de Ollama (https://ollama.ai/download)
3. Ejecutar el archivo .exe descargado y seguir las instrucciones del asistente
4. Verificar la instalación abriendo PowerShell y ejecutando: `ollama --version`

La instalación en Windows crea automáticamente los accesos directos necesarios y configura las variables de entorno para facilitar su uso inmediato.

Instalación en macOS

Instalación mediante Homebrew:

```
brew install ollama
```

Instalación manual:

1. Descargar el instalador .dmg desde https://ollama.ai/download
2. Abrir el archivo .dmg y arrastrar Ollama a la carpeta Aplicaciones
3. Ejecutar Ollama desde Aplicaciones o Launchpad

Instalación en Linux

multi versica ACADEMY

Instalación mediante script oficial:

```
curl -fsSL https://ollama.ai/install.sh | sh
```

Instalación manual en Ubuntu/Debian:

```
# Descargar el paquete .debwget https://github.com/ollama/ollama/releases/latest/download/ollama-linux-amd64.deb#
Instalar el paquetesudo apt install ./ollama-linux-amd64.deb
```

Para otras distribuciones como Fedora, Arch o CentOS, consulta la documentación oficial que proporciona instrucciones específicas para cada sistema.

Después de la instalación, verifica que todo funcione correctamente con: ollama --version

Primera Ejecución

multi versica ACADEMY

¡Tu transformación comienza ahora!

Únete a los líderes gubernamentales que ya están revolucionando el sector público con IA responsable

- 8 módulos especializados con metodología MPE
- Entorno inmersivo VR/WebVR incluido
- Asistente IA 24/7 y soporte (6 meses)

Inscripción abierta

Reserva tu lugar en la próxima cohorte de líderes en IA gubernamental

Inversión completa

(Incluye todas las características especiales y soporte)

Inscribirse ahora

Garantía de satisfacción de 30 días

Información y contacto

multiversica.academy

educacion@augexp.com