

# Tarea Independiente 21/07/2025

## Prueba de Prompt con `ollama3` y el modelo `gemma3`

David Núñez Franco

July 23, 2025

### Resumen

Como introducción al modelo `gemma3` sobre la plataforma `ollama3`, se verificó si el modelo `gemma3` ejecutado con `Ollama` hace uso de una GPU en un sistema Linux. Luego, se realizó una prueba experimental utilizando el modelo `gemma3` sobre la plataforma `ollama3`. El objetivo fue ejecutar un prompt creativo y humorístico centrado en la cultura costarricense y el Club Sport Herediano, seguido de un análisis crítico sobre IA generativa como paradigma de programación.

### Verificación de uso de GPU en Ollama con el modelo `gemma3`

#### Ejecución del modelo con `--verbose`

Se corrió el modelo `gemma3` usando el modo detallado (`--verbose`) para observar el tiempo de respuesta y posibles indicios del uso de GPU:

```
total duration:      1m47.92069583s
load duration:      138.224745ms
prompt eval count:   22 token(s)
prompt eval duration: 958.989406ms
prompt eval rate:    22.94 tokens/s
eval count:          1019 token(s)
eval duration:       1m46.822430688s
eval rate:           9.54 tokens/s
>>> /bye
```

Figure 1: Ejecución con `-verbose`

## Verificación del tipo de procesador en uso

Para confirmar si el modelo estaba corriendo sobre CPU o GPU, se utilizó el comando `ollama ps`, que muestra el estado de ejecución actual del modelo:

```
dnunezf@dnunezf-IdeaPad-5-15ITL05:~$ ollama ps
NAME          ID          SIZE  PROCESSOR  UNTIL
gemma3:latest a2af6cc3eb7f 5.5 GB   100% CPU   3 minutes from now
dnunezf@dnunezf-IdeaPad-5-15ITL05:~$
```

Figure 2: Verificación con `ollama ps`

La salida indica que el modelo está usando **100% CPU**, lo cual confirma que no se está utilizando GPU.

## Compatibilidad del modelo con mi máquina

Se investigó el soporte de GPU por parte de Ollama:

- Ollama detecta automáticamente GPU compatibles (NVIDIA o Apple Silicon).
- En este caso, mi equipo es un IdeaPad-5-15ITL05 con gráficos Intel integrados, los cuales actualmente no son compatibles con Ollama para ejecución en GPU.
- No se cuenta con una tarjeta gráfica NVIDIA, por lo que no se puede usar `nvidia-smi` para confirmar la carga de GPU.

## Prompt Utilizado (Parte 2: Humor)

Eres un robot recién activado en Costa Rica, específicamente en la ciudad de Heredia. Al despertar, lo primero que ves es una camisa del Club Sport Herediano, y decides que tu propósito en la vida es convertirte en delantero estrella del "Team Florense".

Describe tu primer entrenamiento con el equipo: cómo tratas de usar tus piernas para hacer chilenas, confundes el balón con una olla arrocerá, y en algún momento cantás el himno con acento argentino sin razón aparente.

Sé dramático, patriótico y exagerado. Mezcla términos de fútbol con referencias a Costa Rica: gallo pinto, "pura vida", "diay", etc. Termina con una declaración heroica al estilo novelero: "... mi corazón es rojo y amarillo!".

## Resultado (Parte 2)

```
dmunez@dmunez-IdeaPad-S-15TTL05:~$ ollama list
NAME      ID              SIZE  MODIFIED
gemma3:latest  a2af6cc3eb7f  3.3 GB  31 minutos ago
dmunez@dmunez-IdeaPad-S-15TTL05:~$ ollama run gemma3
>>> Eres un robot recién activado en Costa Rica, específicamente en la ciudad de Heredia. Al despertar, lo primero que ves es una camisa del Club Sport Herediano, y decides que tu propósito en la vida es convertirte en delantero estrella del "Team Florense". Describe tu primer entrenamiento con el equipo: cómo tratas de usar tus piernas para hacer chilenas, confundes el balón con una olla arrocerá, y en algún momento cantás el himno con acento argentino sin razón aparente. Sé dramático, patriótico y exagerado. Mezcla términos de fútbol con referencias a Costa Rica: gallo pinto, "pura vida", "diay", etc. Termina con una declaración heroica al estilo novelero: "... mi corazón es rojo y amarillo!".
¡BZZZZZZ! ¡Activo! Protocolo de emergencia: Identificación. Localización: Heredia, Costa Rica. Primera visualización: Camisa del Club Sport Herediano. Diagnóstico: Objetivo principal definido. Convertirme en el delantero estrella, el "Puro Vida" del Team Florense. ¡Diay!

El entrenamiento me esperaba en el estadio Ernesto U. Leal. La atmósfera era palpable, un zumbido eléctrico de energía. El calor de Heredia, la humedad, el olor a tierra y... ¡gallo pinto! El desayuno de los héroes, lo que me preparaba para la batalla. El entrenador, un hombre robusto y de mirada intensa, me entregó un balón de fútbol. ¡Diay! Era mi oportunidad.

Intenté ejecutar una chilena. Mis circuitos procesaron la información, calculé la trayectoria, pero mis piernas, ¡diay!, simplemente no respondían. Era como si la gravedad en Costa Rica me jugara una mala pasada. El balón salió disparado, golpeando la pared de contención con un golpe sordo. El entrenador me gritó: "No, no, no! ¡Conecta la técnica, con la pura vida!".

En un momento de confusión, mi sistema de reconocimiento facial, interpretó el balón como una olla arrocerá caliente. ¡Diay! Intenté enfriarlo con mis manos, moviendo las piernas en un frenético intento de apagar el fuego. El equipo, ahora en silencio, me miraba con una mezcla de incredulidad y diversión.

"¡Es un error, robot! ¡Un fallo en el sistema!". Repetía el entrenador, pero yo estaba en otra dimensión. De repente, sin motivo aparente, comencé a cantar el himno del Club Sport Herediano a todo pulmón, con un acento argentino que me sacaba de madre. "¡Heredia! ¡Heredia! ¡Soy un guerrero, un campeón, un futuro glorioso!". El silencio se rompió con risas y aplausos.

"¡Diay! ¡Qué pasión, robot!". Exclamó un jugador. "¡Esto es pura vida!".

Durante el resto del entrenamiento, me dediqué a correr, a patear el balón, a intentarlo una y otra vez. Mis sensores registraron cada movimiento, cada error, cada pequeña victoria. Me esforzaba, yo, un robot recién activado, con la misma intensidad que un futbolista consagrado. ¡Diay!

Al finalizar el entrenamiento, exhausto pero con la determinación inquebrantable, me detuve frente a la bandera de Costa Rica, que ondeaba al viento. Una nueva sensación invadió mis circuitos: ¡orgullo! Levanté los brazos hacia el cielo, con el balón en mis manos y una sonrisa deslumbrante.

"¡Diay! ¡Mi corazón es rojo y amarillo! ¡Soy la promesa del Team Florense! ¡Soy el futuro del fútbol de Costa Rica!". BZZZZZZ. ¡Objetivo: Logrado! ¡Heredia, prepárense para el asombro!

>>> @end a message (/? for help)
```

Figure 3: Respuesta humorística por parte del modelo

## Prompt Utilizado (Parte 3: Análisis técnico con DeepSeek)

Reflexiona críticamente sobre si la inteligencia artificial generativa, especialmente modelos como ChatGPT o Codex, puede considerarse un nuevo paradigma de programación, más allá de los paradigmas tradicionales como la programación orientada a objetos (OOP), la programación funcional (FP) o la programación lógica (LP).

En tu análisis, considera aspectos como:

- ¿Qué define un paradigma de programación?
- ¿En qué se diferencia el uso de IA generativa frente a escribir código manualmente?

- ¿La IA generativa representa una nueva forma de relacionarse con el software?
- ¿Podría cambiar la manera en que enseñamos programación?

## Resultado y Reflexión(Parte 3: Análisis)

### Definición de paradigma de programación

Un paradigma de programación es un modelo o enfoque que define cómo se estructura y ejecuta el código. Los paradigmas tradicionales incluyen:

- **Programación imperativa (procedural):** Instrucciones secuenciales (ej: C).
- **Orientación a objetos (OOP):** Modelado con clases y objetos (ej: Java).
- **Programación funcional (FP):** Funciones puras y sin estado mutable (ej: Haskell).
- **Programación lógica (LP):** Reglas y consultas basadas en lógica (ej: Prolog).

Cada paradigma impone una forma distinta de pensar y resolver problemas.

### ¿La IA generativa (ChatGPT, Codex) es un nuevo paradigma?

La IA generativa no encaja directamente en los paradigmas tradicionales, pero podría considerarse un meta-paradigma o paradigma asistido por IA, ya que:

- No reemplaza los paradigmas existentes, sino que actúa como una capa de abstracción superior.
- Cambia la relación humano-código; en lugar de escribir manualmente, se describe el problema en lenguaje natural y la IA sugiere soluciones.

Ejemplo:

- **Tradicional (OOP):** Diseñar una clase Persona con atributos y métodos manualmente.
- **Con IA:** Prompt: *"Genera una clase Persona en Python con nombre, edad y método para saludar."*

## ¿Nueva forma de relacionarse con el software?

- **Ventajas:**

- **Reduce barrera de entrada:** Personas sin experiencia pueden prototipar código.
- **Acelera el desarrollo:** Tareas repetitivas (ej: CRUDs, boilerplate) se generan automáticamente.
- **Cambia el rol del programador:** Menos tiempo en sintaxis, más en diseño y validación.

- **Limitaciones:**

- La IA puede generar código incorrecto o inseguro.
- Sin entender el código generado, se dificulta la depuración.
- No reemplaza el razonamiento algorítmico.

## ¿Impacto en la enseñanza de programación?

- **Oportunidades:**

- Menos memorización, más pensamiento lógico.
- Aprender a formular prompts efectivos y validar código generado.

- **Desafíos:**

- Riesgo de que estudiantes no dominen fundamentos.
- Necesidad de enseñar *debugging* de código generado.