Usando LLMs

Mariano Crosetti

Rosario, Argentina
Universidad Austral
Maestría en Explotación de Datos y Gestión del Conocimiento

"One ring to rule them all [...]" - J. R. R. Tolkien

Contenidos I

Contenidos

Modelo de Lenguaje - LM

Un modelo de lenguaje formalmente calcula:

Modelo de Lenguaje - LM

Un modelo de lenguaje formalmente calcula:

$$P(x_{t+1} | x_1 x_2 x_3 \ldots x_t)$$

- En criollo: calcula la probabilidad de una palabra **dada las anteriores**.
- Es un problema de

Modelo de Lenguaje - LM

Un modelo de lenguaje formalmente calcula:

$$P(x_{t+1} | x_1 x_2 x_3 \ldots x_t)$$

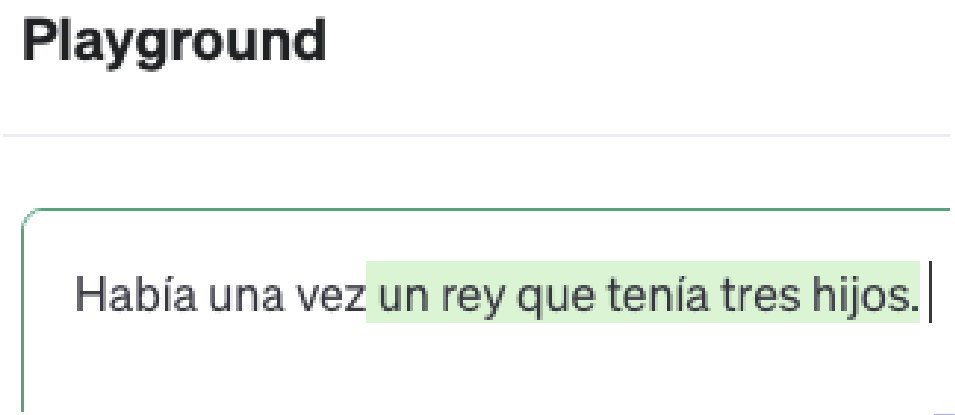
- En criollo: calcula la probabilidad de una palabra **dada las anteriores**.
- Es un problema de **clasificación** (las etiquetas son el conjunto de todas las palabras o vocabulario V).
- Y de aprendizaje

Modelo de Lenguaje - LM

Un modelo de lenguaje formalmente calcula:

$$P(x_{t+1} | x_1 x_2 x_3 \ldots x_t)$$

- En criollo: calcula la probabilidad de una palabra **dada las anteriores**.
- Es un problema de **clasificación** (las etiquetas son el conjunto de todas las palabras o vocabulario V).
- Y de aprendizaje **supervisado**: dado un corpus de texto, tenemos los pares (x,y) tomando prefijos del corpus.
- Como la señal de supervisión (etiquetas) son generadas automáticamente (no por humanos) se lo llama aprendizaje **auto-supervisado** (*self-supervised*).



Contenidos

Tokenización

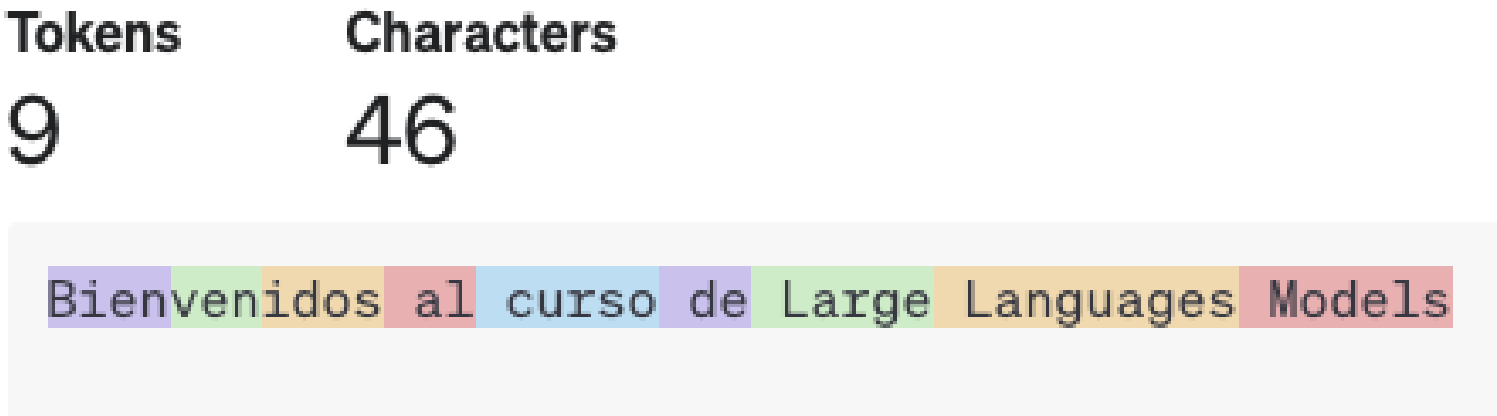
- **Token:** unidad básica de texto.
- El conjunto de todos los tokens constituye el vocabulario V con el que trabaja el modelo de lenguajes.
- Puede ser el conjunto de todas las palabras. Qué problema tiene esto?

Tokenización

- **Token:** unidad básica de texto.
- El conjunto de todos los tokens constituye el vocabulario V con el que trabaja el modelo de lenguajes.
- Puede ser el conjunto de todas las palabras. Qué problema tiene esto?
  - Asume que V es fijo y aprendible durante el entrenamiento.
  - Las palabras que no hayan aparecido durante el entrenamiento no tienen embeddings calculados. Lo cual es problemático para tratar: variaciones (deliciooooso), errores de tipeo, términos nuevos ("twittear").
  - Lenguajes morfológicamente complejos requieren entrenar word embeddings de las declinaciones como si fueran palabras independiente (por ejemplo conjugaciones "jugandoz "jugar").
- **Solución a lo anterior?**

Tokenización (cont.)

- **Solución:** que la unidad básica de texto (o tokens) sea a nivel "sub-palabra".
- Cómo elegimos las subpalabras?
  - Tokenizadores para cada lenguaje que capturen la morfología del lenguaje.
  - Tokenizadores probabilísticos (ver algoritmo *Byte-pair*). Ahora los tokens son cadenas de texto frecuentes.

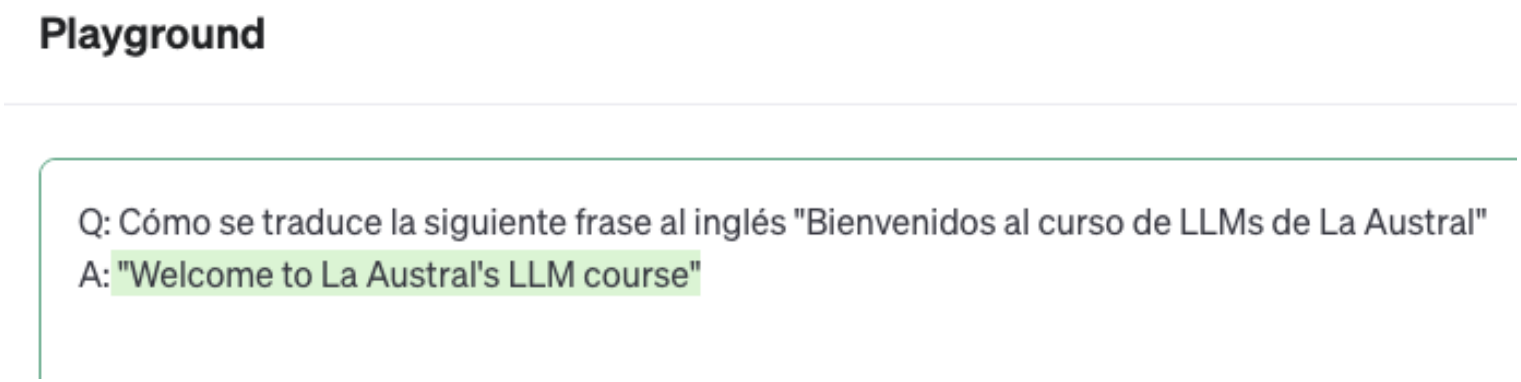


Contenidos

Large Language Models

Cuándo hacemos modelos suficientemente grandes (1 billón de parámetros), aprende capacidades que nos hace pensar que realmente está capturando:

- Sintaxis y gramática del lenguaje
- Conocimiento
- Lógica y razonamiento



Mmm... quizás podemos usar esto para resolver problemas!

ChatGPT

OpenAI le dió una vuelta de tuerca a la última idea de la sección anterior y entrenó un modelo de lenguaje conversacional:

- Algunos mensajes son del “usuario” que lo consulta.
- Y otros del “asistente”.
- Está entrenado en conversaciones donde el “usuario” da instrucciones al “asistente”, quién trata de resolverlas.
- No es más que un modelo de lenguaje dónde el rol (usuario y asistente) están codificados de alguna manera:

```
<usuario>
Cuánto es 2 + 2?
</usuario>
<asistente>
2 + 2 es 4
</asistente>
```

Se creó así el ChatGPT que todos conocemos.

ChatGPT

Se puede usar el asistente como fuente de inteligencia para resolver problemas de NLP:

- Clasificación:
- Information Extraction
- Summarization
- Translation
- Question Answer

USER

Classify the sentiment of the movie review in "positive" or "negative.
Only output any of the words "positive" or "negative"

Text:
Might end up being the biggest disappointment that I will see in 2009. I seem to be the rare person who disliked Park's Oldboy, but I think that his "Lady Vengeance" and "Sympathy for Mr. Vengeance" are among the best films I've seen in the 2000's decade. Therefore, I really was looking forward to see this, especially as it got such positive reviews. Instead, I found the film clichéd, and broke little, if any new ground to the vampire genre. And while I can appreciate a bit of gallows humor in movies like this, I felt Park did this at very inopportune times.

Others have compared/contrasted this to "Let the Right One In," and I have to say that "Let the Right One In" was far superior to this one, and was a fresh take on the vampire genre. Sadly, Park's take was a tired one.

ASSISTANT {f}

negative

ChatGPT

Ejemplo de Information Retrival:

USER

Extract from the text the following properties of the described Person:
- Birth date
- Full name
- Country of precedence

Extract it in a JSON with the fields "birth_date", "fullname", "contry_of_precedence"

Text:
Napoleón Bonaparte (nacido Napoleone Buonaparte; Ajaccio, 15 de agosto de 1769-Santa Elena, 5 de mayo de 1821) más tarde conocido por su nombre regio Napoleón I, fue un militar y político francés de origen italiano nacido en Córcega que saltó a la fama durante la Revolución francesa y dirigió exitosas campañas durante las Guerras revolucionarias. Fue el líder de facto de la República Francesa como primer cónsul desde 1799 hasta 1804, y después emperador de los franceses desde

ASSISTANT

```json
{
  "birth_date": "1769-08-15",
  "fullname": "Napoleón Bonaparte",
  "contry_of_precedence": "Francia"
}
```

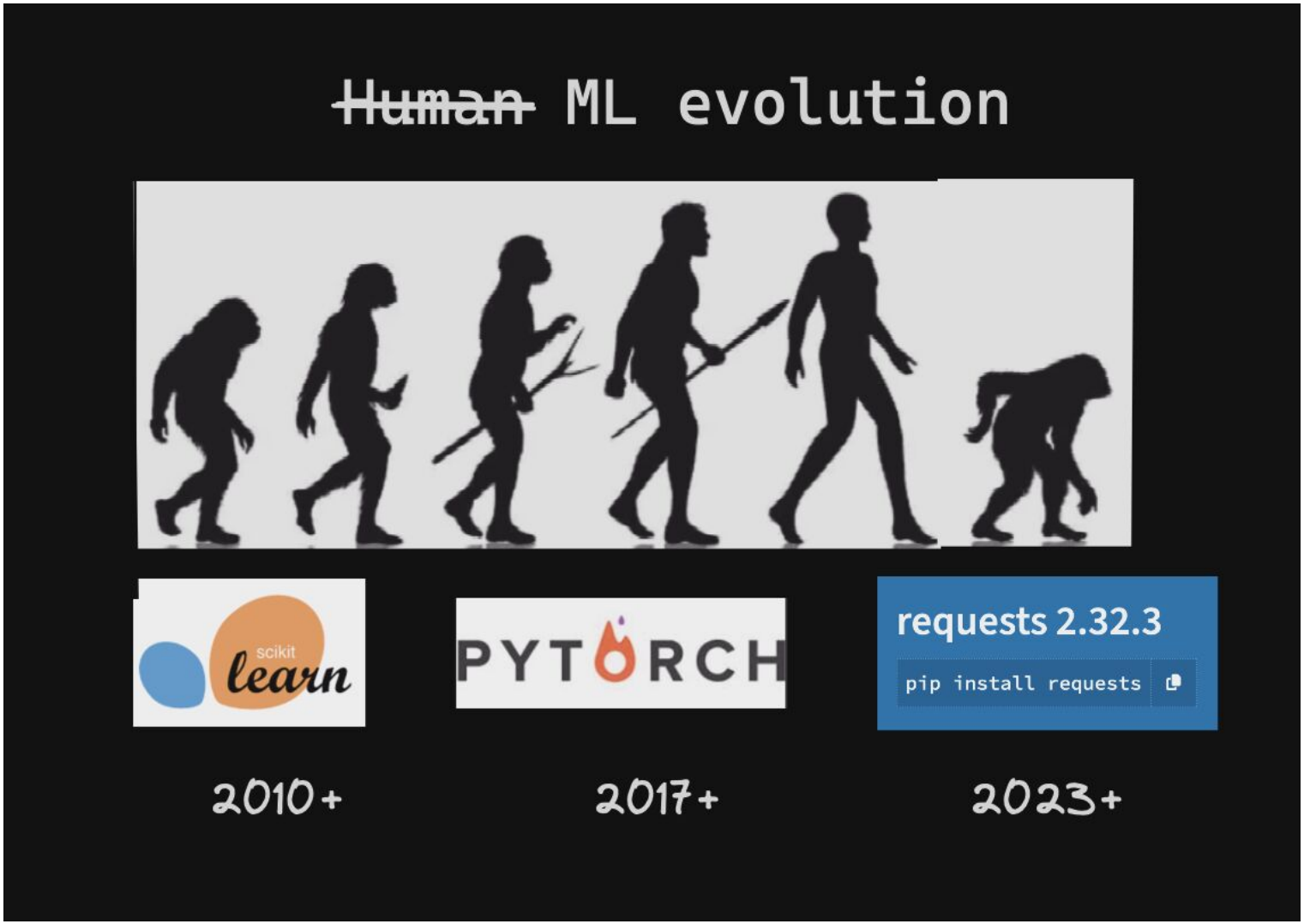
OpenAI API

OpenAI provee una API que permite que nuestros programas sean los üsuarios"del ChatGPT.

```
from openai import OpenAI
instr = 'Cual es la traducion del siguiente texto al ingles "{}"?\nProduci
    un JSON con el campo "answer"'
def translate_es_to_en(string_to_translate):
    client = OpenAI()
    response = client.chat.completions.create(
        model="gpt-3.5-turbo",
        messages=[
            {"role": "user", "content": instr.format(string_to_translate)},
        ]
    )
    return response.choices[0].message.content
```

OpenAI API

Esto nos permite crear muchísimas aplicaciones!



Particularidades de OpenAI API

- JSON Mode: fuerza al modelo a producir JSON. Aún así, hay que incluir también una instrucción que le pida explícitamente que el output sea JSON.
- system role: diseñado para proveer al asistente información sobre como comportarse y así condicionar su manera de responder.
- seed: permite contolar la aleatoriedad. Poniendo una misma seed los resultados deberían ser deterministas.
- stop: Para setear una secuencia de tokens que interrumpen la producción.
- temperature: Para ponderar tokens más allá del más probable.
- n: Cantidad de sampleos a hacer.
- logprobs | toplogprobs: si retornor las probabilidades de los tokens elegidos.
- frecuency penalty | presence_penalty: Para penalizar tokens que ya aparecieron
- models: Para elegir el modelo.

Dudas?

Dudas? Síganme en `https://marianocrosetti.com`

MARIANO
CROSETTI

–

NLP & Computer Vision
SWE Distributed Systems
ICPC Coach & LATAM Champion

LEARN

WORK

READ

CHILL