

M5. Minería de Texto + webscraping

Clase 6. Transformers, LLMs y loros aleatorios



Un ejemplo

INPUT

Je suis étudiant

Traducción
Problema Sequence to Sequence

OUTPUT

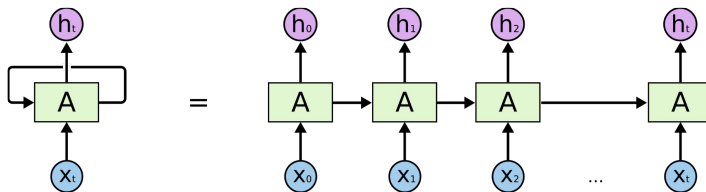
I am a student



Un ejemplo

INPUT

Je suis étudiant



OUTPUT

I am a student



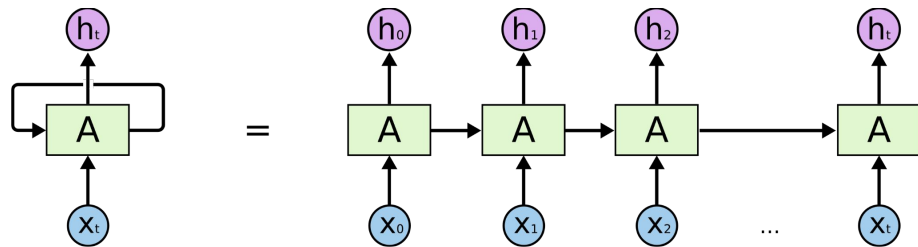
Un ejemplo

- El modelo presentado hasta ahora no tiene memoria: los inputs se presentan de manera independiente y no se tiene en cuenta relación entre ellos.
- Cuando leemos texto, esto no es así. Procesamos las letras, las palabras y las oraciones teniendo en cuenta la información que leímos previamente.
- Las **Recurrent Neural Networks** imitan esta lógica → primeros modelos de trabajo con texto.



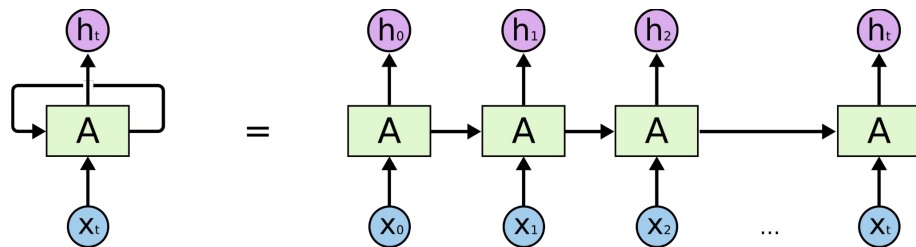
Un ejemplo - RNN

- Aprendizaje secuencial, tiene loop interno y va aprendiendo sobre lo que ya vio.
 - Sigue un loop interno. En cada iteración considera el estado actual del input y lo introduce (hidden state) para obtener output.



Un ejemplo - RNN

- Limitaciones
 - Es secuencial, loop que pasa de una etapa a la otra.
 - No hay una *paralelización* del aprendizaje, o sea, no capturan relaciones globales en un texto.



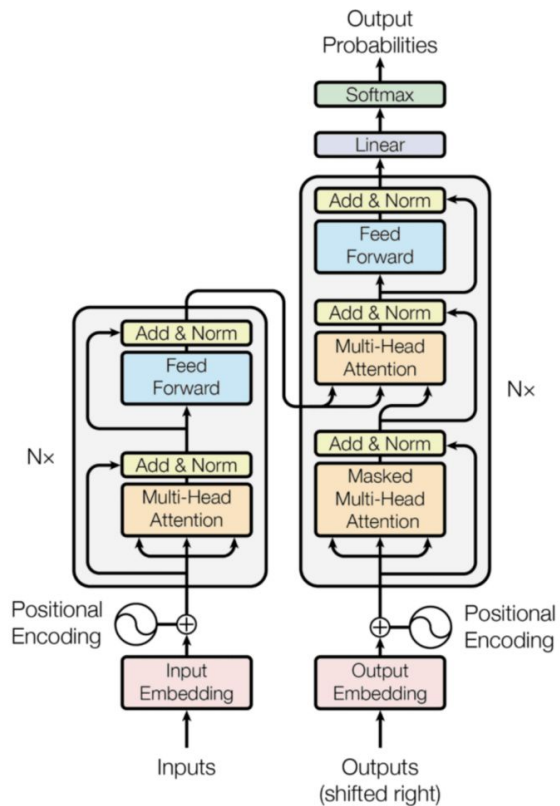
Un ejemplo



Transformers

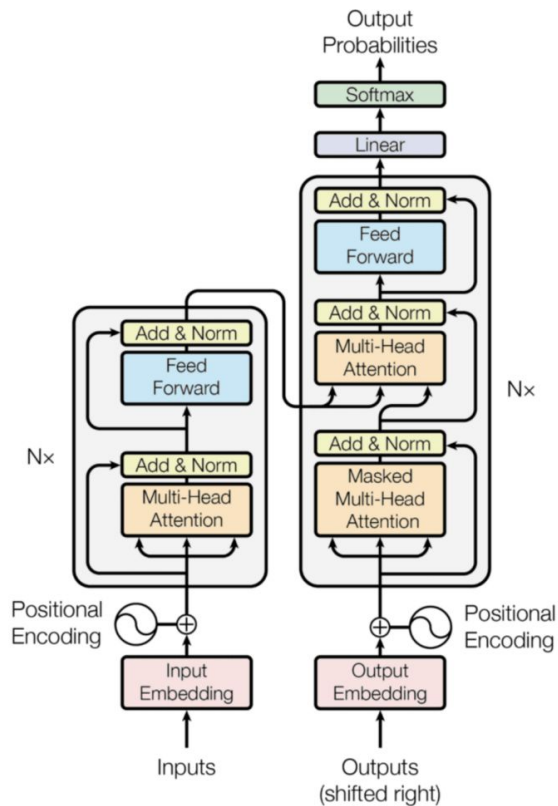
- Modelo paralelizable → puede procesar varias partes de una secuencia al mismo tiempo, lo que acelera considerablemente el entrenamiento y la inferencia.
- Capta las dependencias a largo plazo en el texto, lo que permite comprender mejor el contexto general y generar textos más coherentes.
- Utiliza mecanismos de **self-attention**.

Transformers

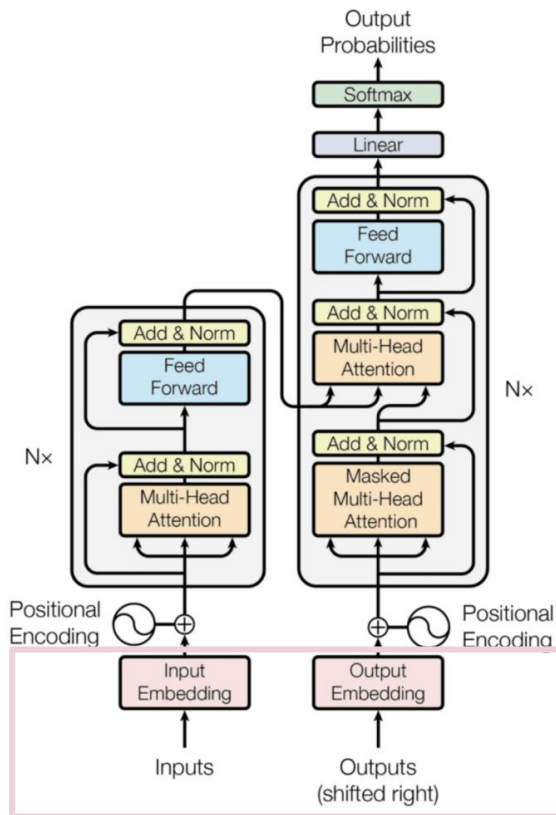


Transformers

Tres mecanismos importantes



Transformers

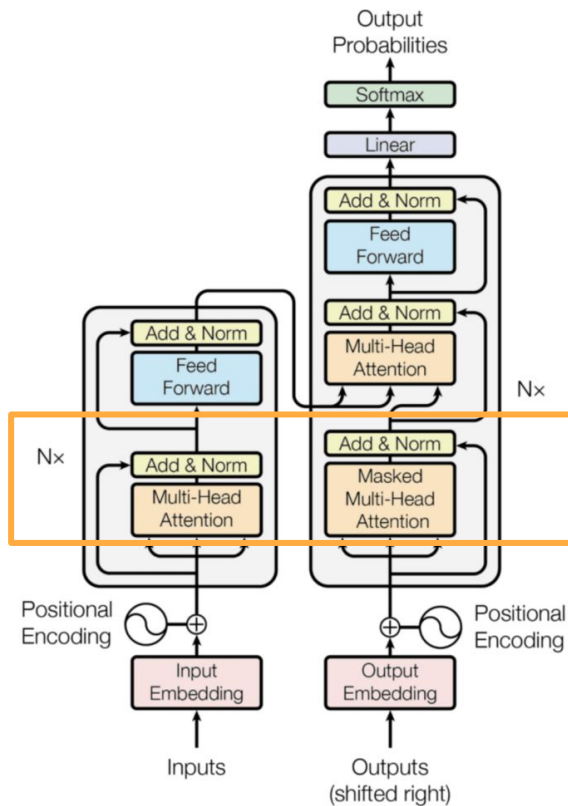


Tres mecanismos importantes

- Input/Output Embeddings



Transformers

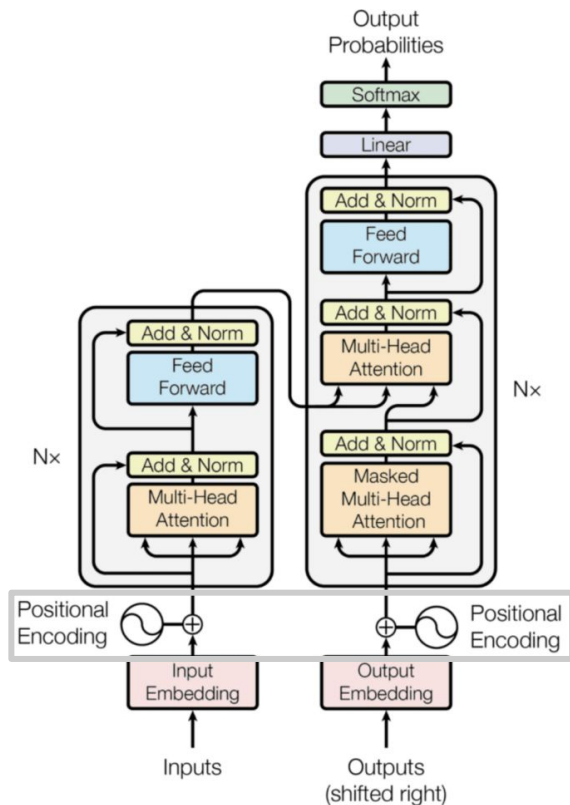


Tres mecanismos importantes

- Input/Output Embeddings
- Multi-head Attention



Transformers

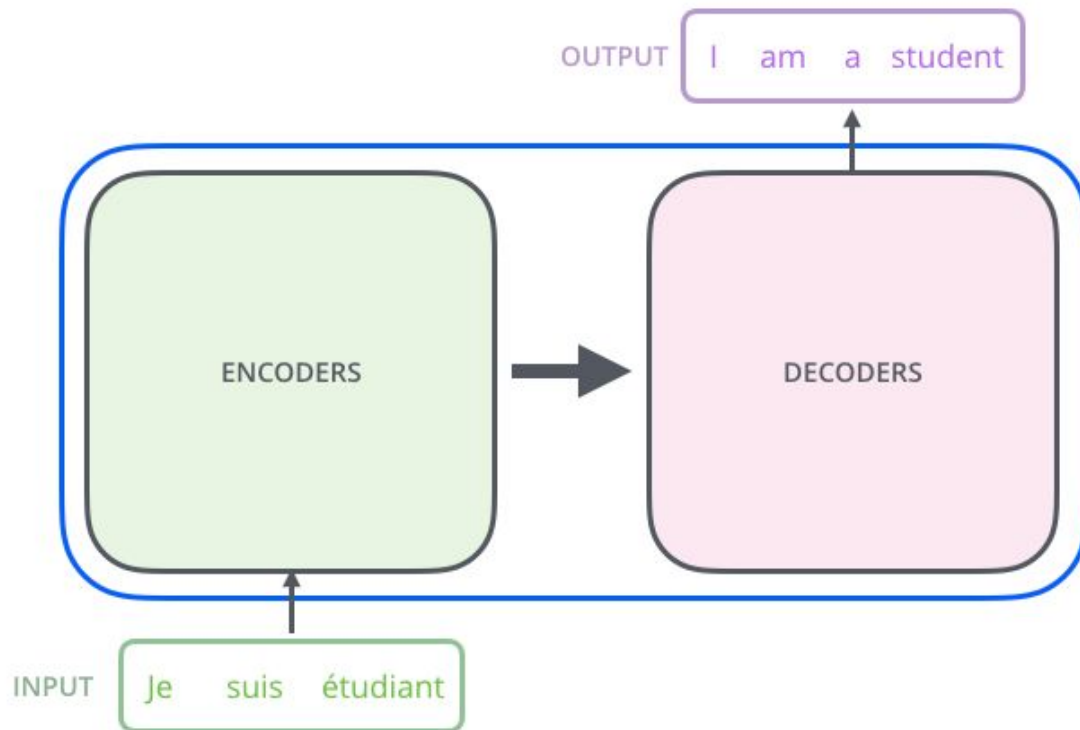


Tres mecanismos importantes

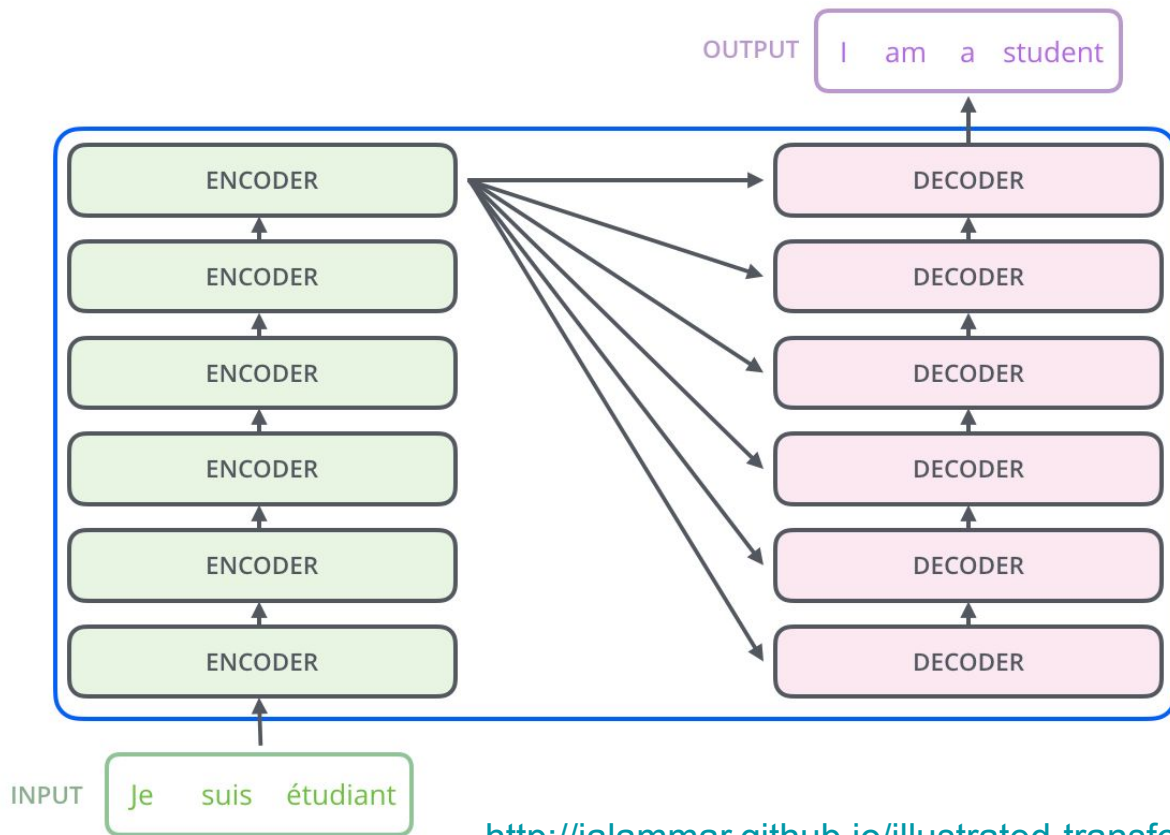
- Input/Output Embeddings
- Multi-head Attention
- Positional encoding



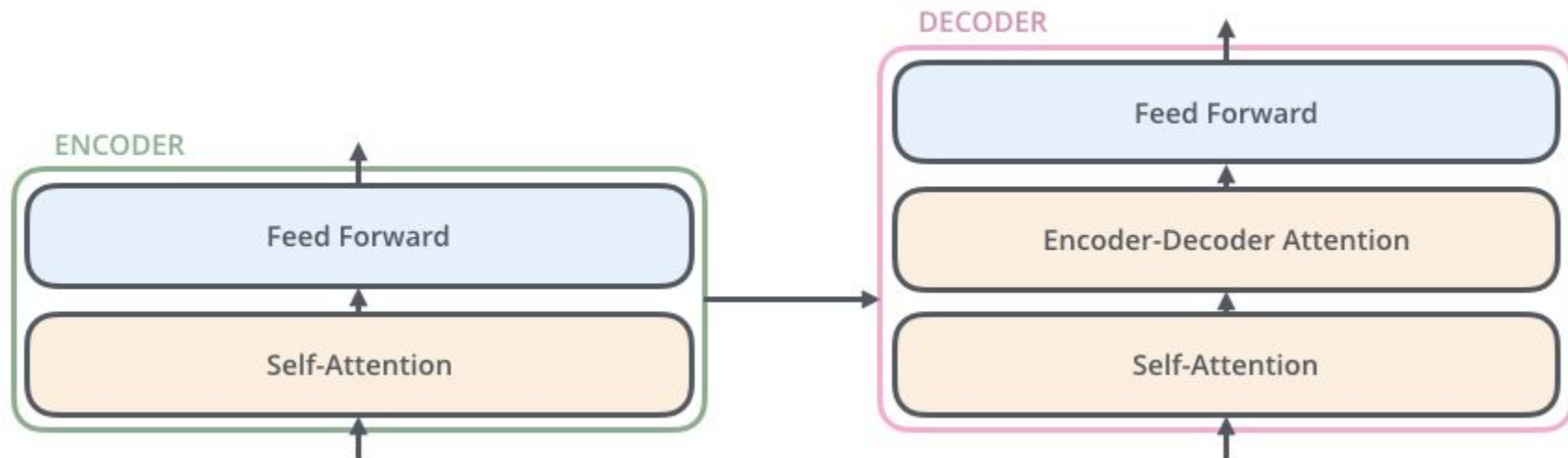
Abriendo la caja



Abriendo la caja



Abriendo la caja

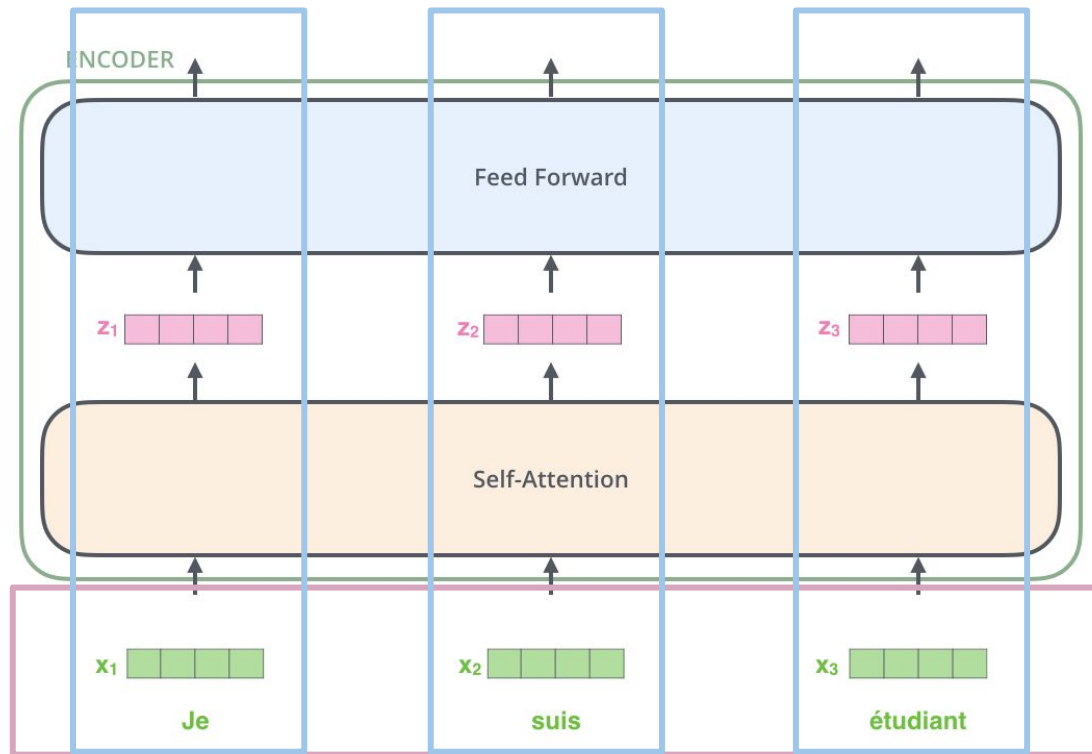


Abriendo la caja

Cada palabra “fluye” de forma paralela a través del encoder.

¿Cómo se recuperan las dependencias de palabras? =>
Self-Attention mechanism

Word Embedding
(d = hiperparámetro)
Se entrena con el modelo



Self-attention

“El perro no jugó con el niño porque él tenía pulgas”

- ¿A quién remite el término “él”? ¿Al perro o al niño?
- Para nosotros es evidente, pero para un modelo no.
- Cuando el modelo procesa la palabra "él", la atención propia le permite asociarla con “perro”.
- A medida que el modelo procesa cada palabra (cada posición en la secuencia de entrada), *self-attention* le permite buscar otras posiciones en la secuencia de entrada en busca de pistas que puedan ayudar a codificar mejor esta palabra.



Self-attention

- Cada input se asocia a tres vectores:
 - Query (Q), Key (K) y Value (V).
 - Los vectores surgen de multiplicar cada embedding de cada palabra por una matriz de pesos (W_Q , W_K y W_V) que se aprenden durante el entrenamiento.
- Se calculan las puntuaciones de similitud entre los vectores de Q y K.
 - Indican cuánta atención debe prestarse a cada elemento de la secuencia al procesar el elemento actual.
- Suma ponderada: Las puntuaciones de atención se utilizan para calcular una suma ponderada de los vectores. Esta suma ponderada representa el contexto o la información de toda la secuencia de entrada relevante para el elemento actual.



Self-attention

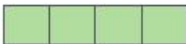
Son producto del entrenamiento...

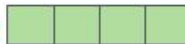
Input

Thinking

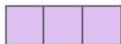
Machines

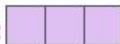
Embedding

X_1 

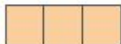
X_2 

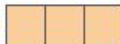
Queries

q_1 

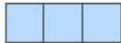
q_2 

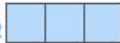
Keys

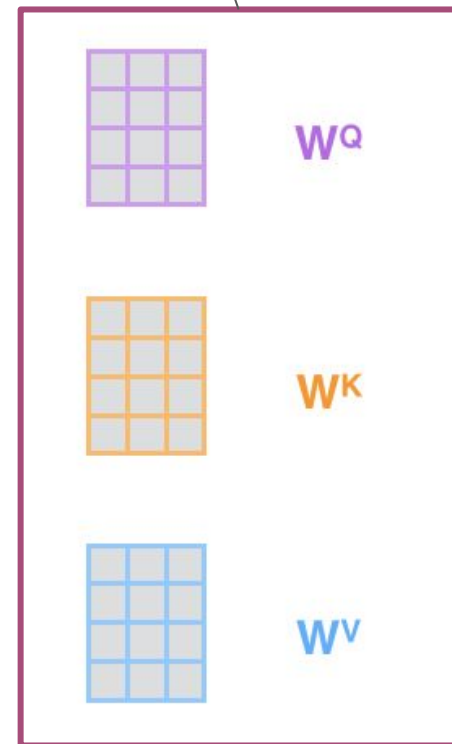
k_1 

k_2 

Values

v_1 

v_2 



Self-attention

- Atención multicabezal: La autoatención se aplica normalmente en paralelo varias veces con diferentes conjuntos de vectores Q, K y V aprendidos, creando múltiples "cabezas de atención".
- Esto permite al modelo centrarse en diferentes aspectos de los datos de entrada y capturar varios tipos de relaciones.

1) This is our input sentence*

2) We embed each word*

3) Split into 8 heads. We multiply X or R with weight matrices

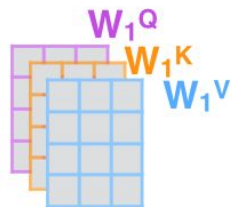
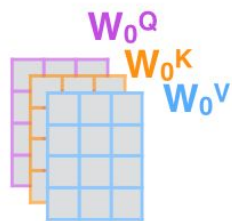
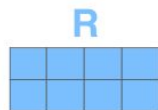
4) Calculate attention using the resulting $Q/K/V$ matrices

5) Concatenate the resulting Z matrices, then multiply with weight matrix W^O to produce the output of the layer

Thinking
Machines



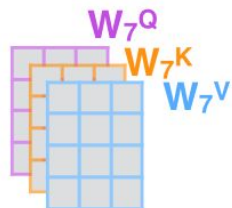
* In all encoders other than #0, we don't need embedding. We start directly with the output of the encoder right below this one



...

...

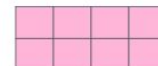
...



W^O



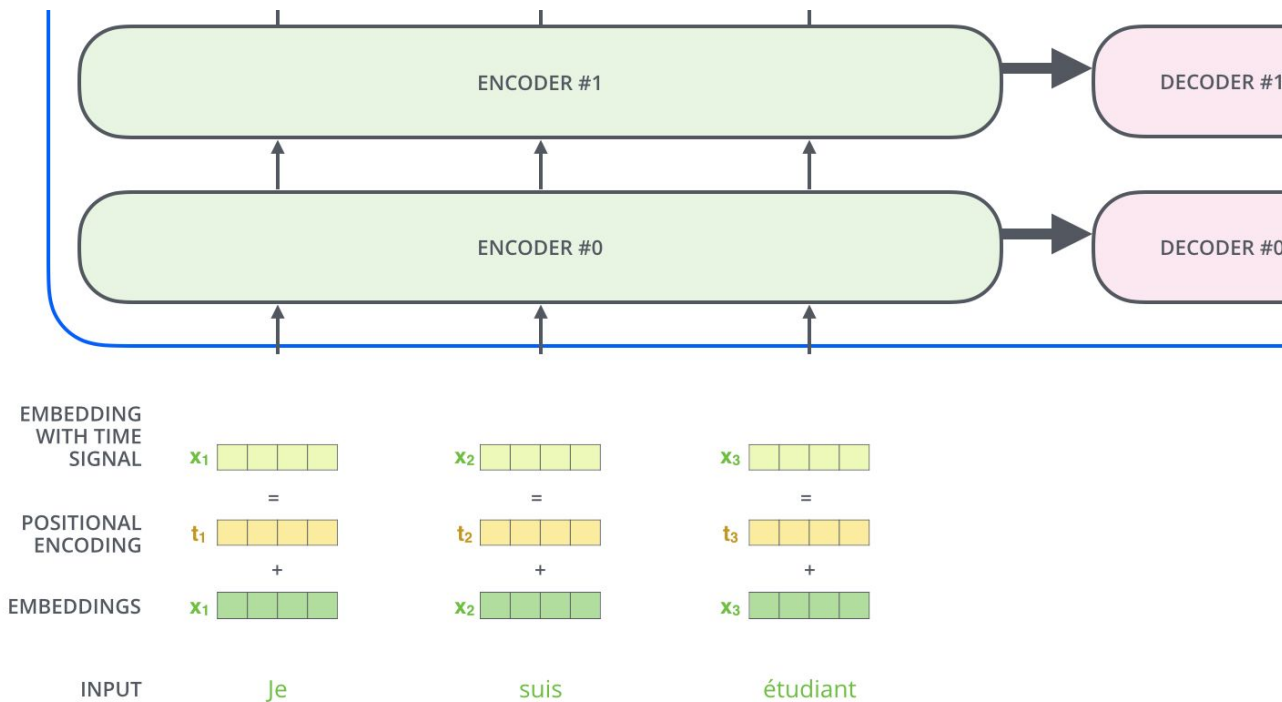
Z



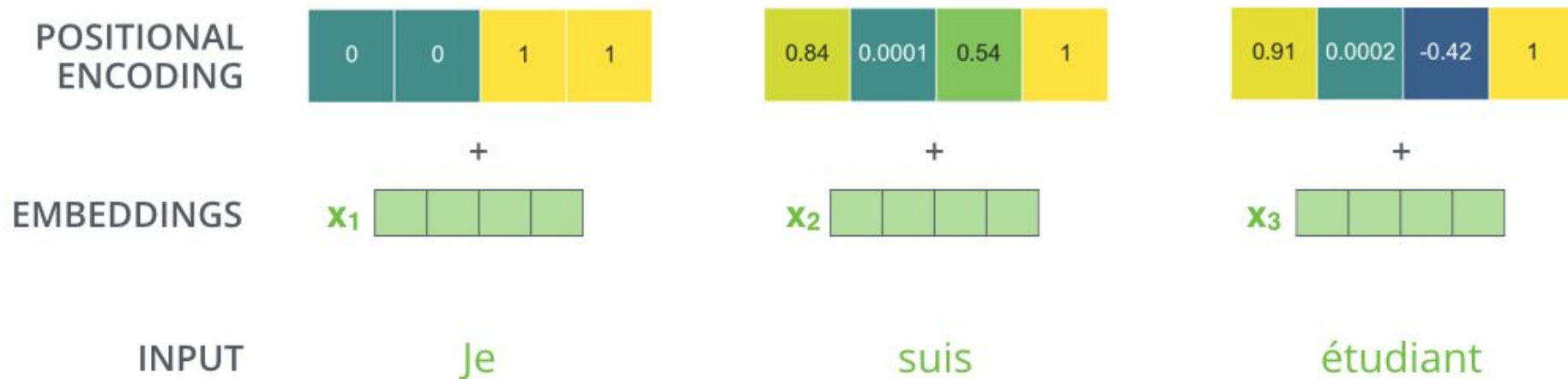
Positional encoding

Nos falta algo:
necesitamos poder
identificar el orden o la
posición de cada palabra
en la secuencia de input.

Para esto, el modelo
agrega un vector a cada
uno de los embeddings
de input



Positional encoding



La evolución de los transformers



Loros aleatorios...



La evolución de los transformers



Facebook Llama

"Smaller, more performant models such as Llama ... democratizes access in this important, fast-changing field."

February 24, 2023



Stanford Alpaca

"Alpaca behaves qualitatively similarly to OpenAI ... while being surprisingly small and easy /cheap to reproduce"

March 13, 2023



Databricks Dolly

"Dolly will help democratize LLMs, transforming them into a commodity every company can own and customize"

March 24, 2023



Mosaic MPT

"MPT-7B is trained from scratch on 1T tokens ... is open source, available for commercial use, and matches the quality of LLaMA-7B"

May 5, 2023



TII Falcon

"Falcon significantly outperforms GPT-3 for ... 75% of the training compute budget—and ... a fifth of the compute at inference time."

May 24, 2023

Non Commercial Use Only | **Commercial Use Permitted**



factor-data
EIDAES_UNSAM

La evolución de los LLM's



David McCandless, Tom Evans, Paul Barton
Information is Beautiful // May 2023



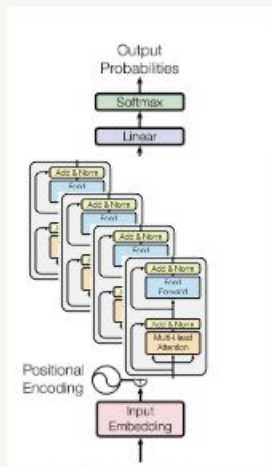
factor-data
EIDAE_SUNSAM

<https://sanchman21.medium.com/evolution-of-transformers-part-1-faac3f19d780>

La evolución de GPT

GPT/GPT-1

12 x

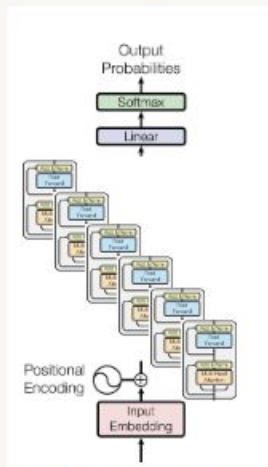


512 dimension
embeddings

GPT-2



48 x

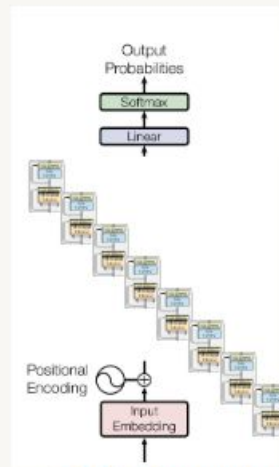


1024 dimension
embeddings

GPT-3



96 x



2048 dimension
embeddings



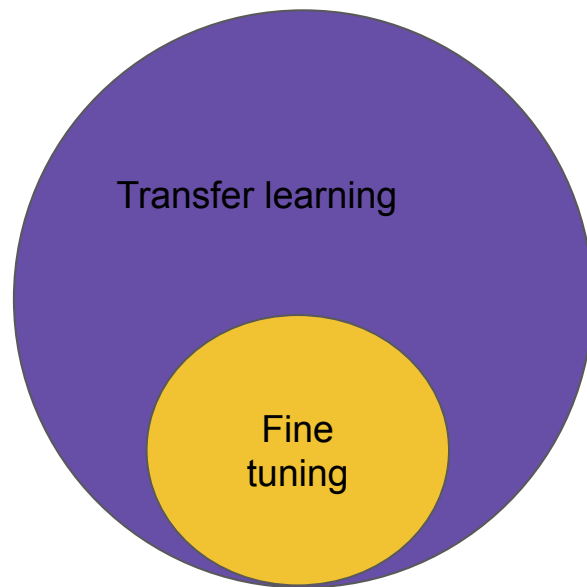
GPT ¿Qué es?

- GPT (2018): pre-entrenado en “BookCorpus” dataset => 117 millones de parámetros. Modelo autorregresivo -----
- GPT-2 (2019): pre-entrenado en un dataset WebText (43TB de datos).
Diferentes tamaños de GPT2: 117M (pequeño), 345M (mediano), 774M (grande), and 1.5B (extra grande).
- PT-3 (2020): pre-entrenado sobre WebText2 (45TB de texto)
- GPT-3 (2022): 175 mil millones de parámetros.
- GPT-4 (2023): 1.76 billones de parámetros



Transfer learning y fine tuning

- Transfer learning: aplicar un modelo general pre-entrenado a una tarea nueva pero relacionada.
- Fine tuning: usar un modelo pre-entrenado pero re-entrenándolo (es decir, modificando algunos, varios o todos sus parámetros) para una tarea nueva.

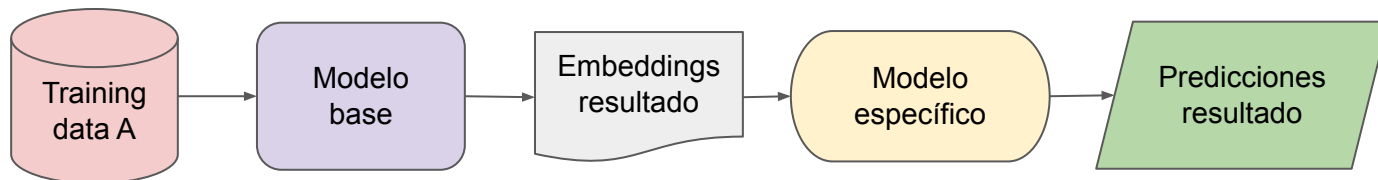


Transfer learning y fine tuning



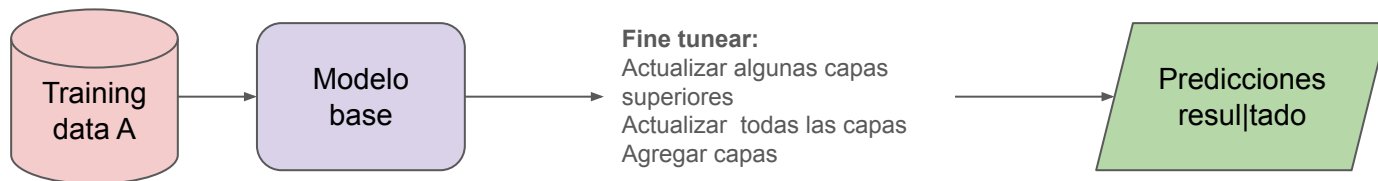
Ejemplo:

T5, GPT4



Ejemplo:

Usar BERT para generar embeddings para usarlos como input de un random forest



Ejemplo:

Hacer fine tuning sobre Llama para extraer información estructurada de texto no estructurado

Transfer learning y fine tuning

- Full fine tuning: generalmente supone producir un modelo por tarea
 - Puede generar “catastrophic forgetting”: que el modelo se olvide de algunas tareas para las que fue pre-entrenado
 - Además suele ser costoso en tiempo y poder de cómputo. ¿Cómo evitarlo?
- ~~● Parameter Efficient Tuning (PEFT):
LoRA, soft tuning, etc.~~
 - X-Shot learning

X-shot learning

- Simplemente supone proveer ejemplos (x-ejemplos) de la nueva tarea
- Muy importante el diseño del prompt:
 - Prompt engineering
 - Prompt design
 - Prompt development
 - hard/discrete prompt tuning
- No requiere modificar los pesos del modelo

```
pipeline(  
    ""For each tweet, describe its sentiment:  
    [Tweet]: "I hate it when my phone battery dies."  
    [Sentiment]: Negative  
    ###  
    [Tweet]: "My day has been 👍"  
    [Sentiment]: Positive  
    ###  
    [Tweet]: "This is the link to the article"  
    [Sentiment]: Neutral  
    ###  
    [Tweet]: "This new music video was incredible"  
    [Sentiment]: """)
```

Instruction

Few-shot examples

Prompt



X-shot learning

Ventajas

- No requiere grandes datasets etiquetados de forma ad-hoc
- No hace falta crear copias del modelo original
- Hay cierto carácter “intuitivo” en el diseño de un prompt

Desventajas

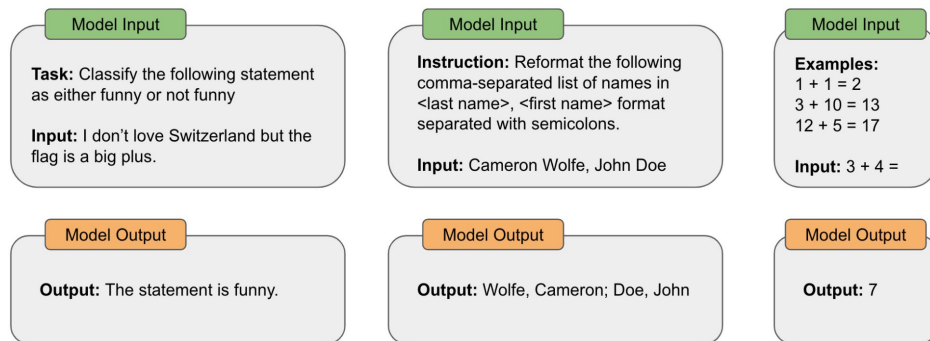
- El prompt engineering es manual
- Los prompts son específicos de cada modelo
- El largo del contexto es una limitación:
 - Si agregamos más ejemplos, hay menos espacio para las instrucciones
 - Contextos más largos => mayor tiempo de respuesta
 - A veces los LLMs “olvidan” las partes intermedias ([Liu et al -2023-](#))



Prompt engineering

- La ingeniería de prompts se refiere a la formulación de instrucciones para Modelos de Lenguaje de gran tamaño (LLMs) con el objetivo de realizar tareas específicas. Estas instrucciones guían el comportamiento del modelo y determinan la calidad del resultado.

Practical Prompt Engineering



Prompt engineering

Importancia

- Guía al modelo en el análisis del texto.
- Influye en la salida del modelo.
- Es una habilidad crucial para dirigir el análisis de textos

Concepto

- Las instrucciones para el modelo representan la forma en que un concepto social científico se codifica.
- Podemos pensar a la ingeniería de prompts como un método cualitativo que busca capturar algún aspecto de la realidad social.

Prompt engineering

Importancia

- **Definir la tarea:** Establecer claramente el objetivo del análisis de texto. ¿Qué tipo de información específica o “insight” se quiere extraer del texto?
- **Determinar el resultado deseado:** Identificar el tipo de respuesta esperada (información factual, opiniones subjetivas, etc.).
- **Considerar longitud y especificidad:** Equilibrar entre un prompt conciso y uno exploratorio según los objetivos del análisis.



Prompt engineering

- **Incluir instrucciones o contexto:** Proveer instrucciones o información contextual que pueda ser relevante. Esto puede incluir “pedirle” al LLM que considere ciertos aspectos del texto
- **Basarse en investigaciones previas:** Utilizar instrucciones de codificación humana como referencia.
- **Hacer los resultados analizables:** Asegurar que la salida del modelo sea consistente y sistemática. Por ejemplo, si la salida esperada es una escala de 1 a 4, podría incluirse “[Contestar en formato “0, 1, 2, 3”. No expliques tu respuesta.]’
- **Iterar y probar:** Experimentar con diferentes formulaciones y ajustar según los resultados obtenidos.



Chain of Thought (CoT) prompting

- Desglosar un problema o una pregunta compleja en una serie de pasos más pequeños y manejables. La idea es que al dividir la tarea en subproblemas y resolver cada uno de ellos secuencialmente, el modelo puede producir respuestas más precisas y coherentes.
- ¿Cómo funciona?
 - Descomposición de la tarea: Se divide la tarea principal en varios pasos intermedios que son más fáciles de resolver.
 - Razones explícitas: Se pide al modelo que genere una "cadena de pensamiento" en lugar de simplemente dar una respuesta directa. Esto significa que el modelo explica su proceso de pensamiento paso a paso.
 - Generación de la respuesta: Después de desglosar el problema y razonar a través de cada paso, el modelo genera la respuesta final.



Chain of Thought (CoT) prompting

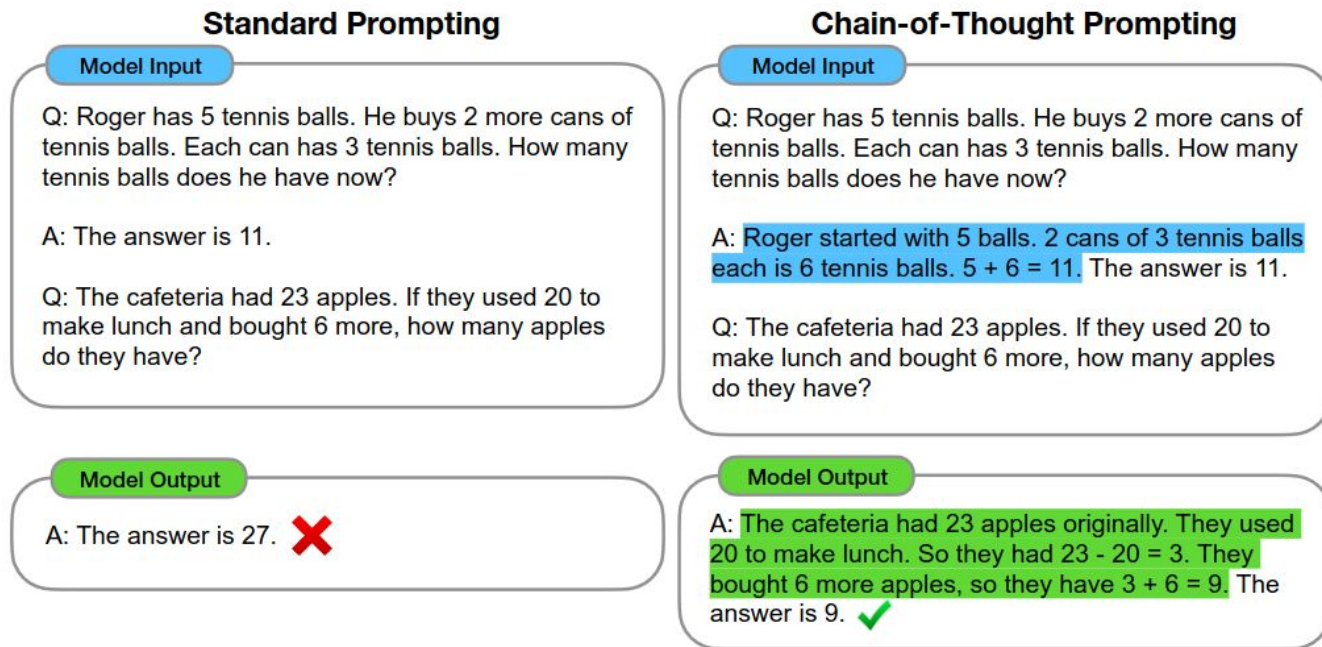


Figure 1: Chain-of-thought prompting enables large language models to tackle complex arithmetic, commonsense, and symbolic reasoning tasks. Chain-of-thought reasoning processes are highlighted.

Chain of Thought (CoT) prompting

Ventajas

- Mejor precisión: Al abordar problemas complejos en pasos más pequeños, se mejora la precisión de las respuestas.
- Transparencia: Proporciona una explicación paso a paso, lo que hace que el proceso de pensamiento del modelo sea más transparente.
- Mejor manejo de problemas complejos: Ayuda al modelo a manejar tareas y preguntas complejas que de otro modo serían difíciles de resolver de una sola vez.



Chain of Thought (CoT) prompting

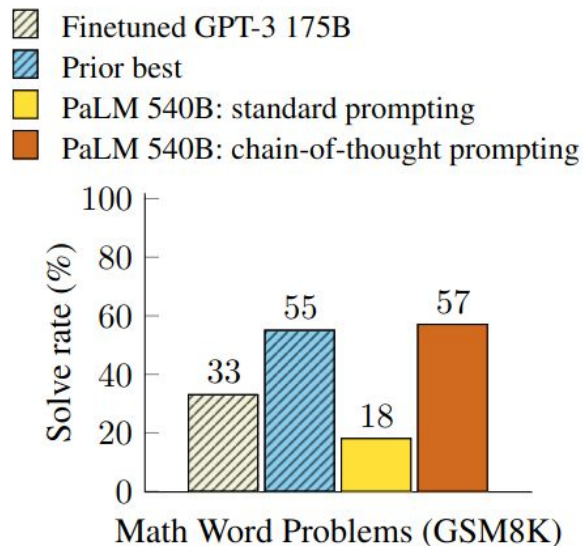


Figure 2: PaLM 540B uses chain-of-thought prompting to achieve new state-of-the-art performance on the GSM8K benchmark of math word problems. Finetuned GPT-3 and prior best are from Cobbe et al. (2021).

Chain of Thought (CoT) prompting

Fuente: [Kojima et al \(2023\)](#)

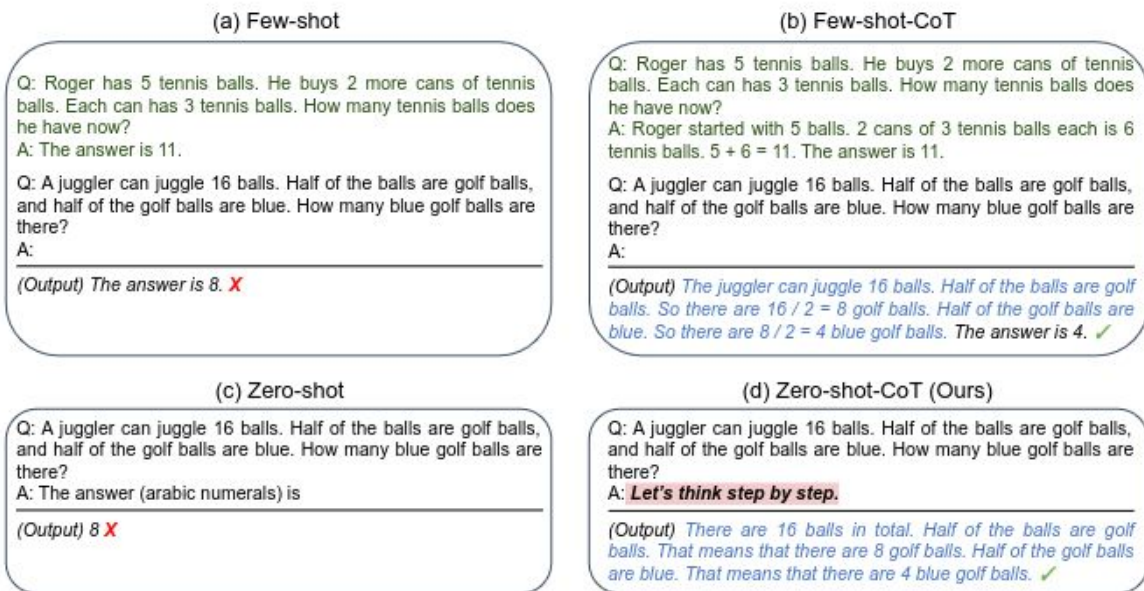
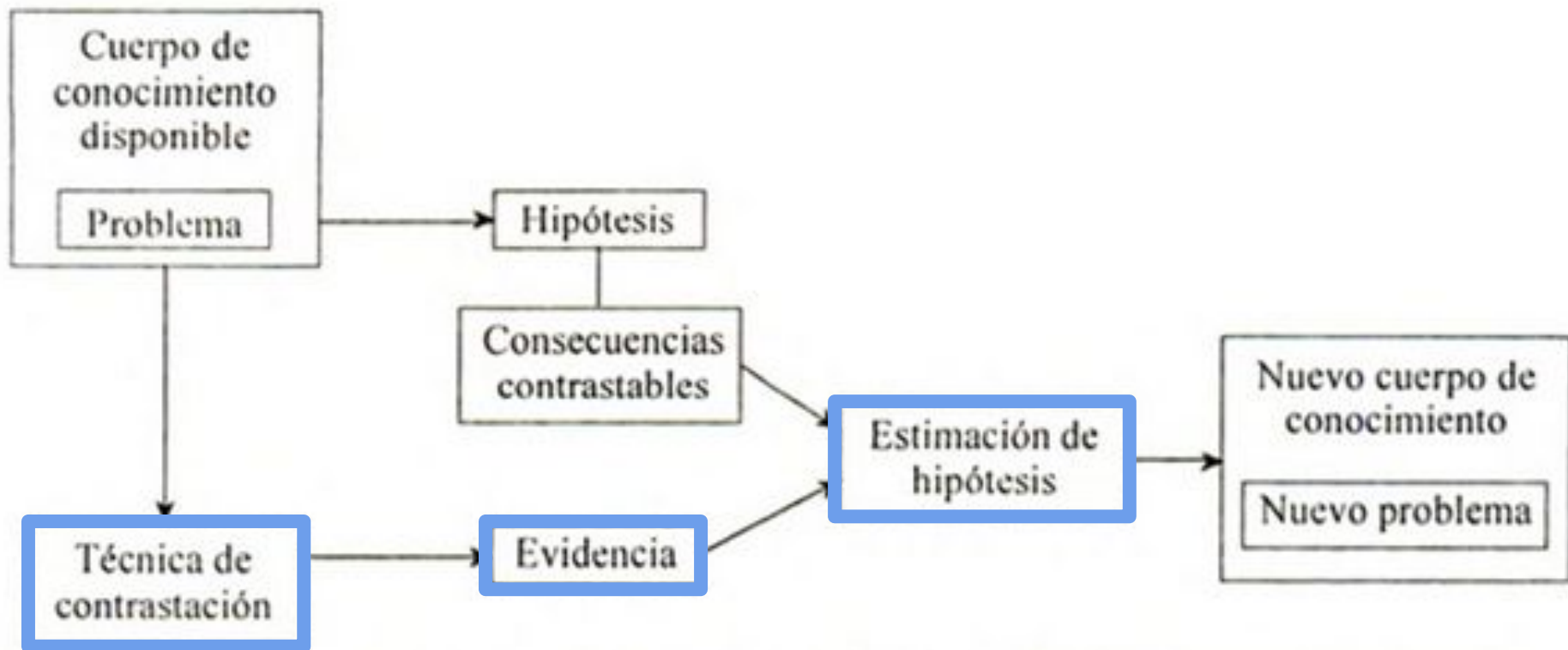


Figure 1: Example inputs and outputs of GPT-3 with (a) standard Few-shot ([Brown et al., 2020]), (b) Few-shot-CoT ([Wei et al., 2022]), (c) standard Zero-shot, and (d) ours (Zero-shot-CoT). Similar to Few-shot-CoT, Zero-shot-CoT facilitates multi-step reasoning (blue text) and reach correct answer where standard prompting fails. Unlike Few-shot-CoT using step-by-step reasoning examples **per task**, ours does not need any examples and just uses the same prompt “Let’s think step by step” *across all tasks* (arithmetic, symbolic, commonsense, and other logical reasoning tasks).

Chain of Thought (CoT) prompting

Desventajas

- Requiere más recursos computacionales: implica producir y procesar mayor cantidad de texto.
- Potencial para errores acumulativos: cada paso de la cadena de pensamiento puede introducir errores, y estos errores pueden acumularse, llevando a una respuesta final incorrecta.
- Limitaciones en la capacidad del modelo: algunos modelos pueden no ser lo suficientemente avanzados para manejar efectivamente CoT, lo que puede limitar su utilidad en ciertos contextos.



Proceso de investigación y automatización

- Definición del problema
 - Formulación del problema
 - Revisión bibliográfica
- Tareas vinculadas a la recolección de datos
 - Construcción de instrumentos
- Tareas específicas vinculadas al procesamiento de información
 - Exploración de texto / “Subrayado” de entrevistas
 - Codificación de preguntas abiertas
 - Código de análisis (R, Python, etc.)

Generative AI for Economic Research: Use Cases and Implications for Economists*

by Anton Korinek[†]

September 2023. This is a living paper - [check back soon](#) for updates.

Abstract

Generative AI, in particular large language models (LLMs) such as ChatGPT, has the potential to revolutionize research. I describe dozens of use cases along six domains in which LLMs are starting to become useful as both research assistants and tutors: ideation and feedback, writing, background research, data analysis, coding, and mathematical derivations. I provide general instructions and demonstrate specific examples of how to take advantage of each of these, classifying the LLM capabilities from experimental to highly useful. I argue that economists can reap significant productivity gains by taking advantage of generative AI to automate micro tasks. Moreover, these gains will grow as the performance of AI systems across all of these domains will continue to improve. I also speculate on the longer-term implications of AI-powered cognitive automation for economic research. The online resources associated with this paper offer instructions for how to get started and will provide regular updates on the latest capabilities of generative AI that are useful for economists.

JEL Codes: A10, B41, J23, O3

*Accepted, *Journal of Economic Literature*. The online resources associated with this paper provide instructions for how to get started with using Generative AI in economic research. They will also provide regular updates on the latest capabilities of Generative AI that are useful for economists. They will soon be available on the journal website <https://www.aeaweb.org/journals/JEL> and at <https://www.aeaweb.org/resources/>. An earlier version of this paper was circulated under the title “Language Models and Cognitive Automation for Economic Research.”

[†]Anton Korinek is a Professor at the University of Virginia, David M. Rubenstein Fellow at the Brookings Institution, the Economics of AI Lead at the Centre for the Governance of AI (GovAI), a Research Affiliate of the NBER and CEPR, and a Senior Advisor of the Partnership on AI. Financial support from Brookings and from Longview Philanthropy is gratefully acknowledged. I thank Julian Hazell, Sid Srinivasan, and participants at a number of seminars and webinars for helpful conversations on the topic. Moreover, I thank Max Schmidman, Don Suh, Natasha Seindler, and Davis Talaferro for excellent research assistance, and GPT-3/4 and Claude/Claude 2 for inspiration and editorial assistance. Moreover, I thank the editor, David Romer, and four anonymous referees for their thoughtful comments.



Proceso de investigación y automatización

- Brainstorming / Ideas / Feedback

- Microtarefas

- escritura, edición, traducción, resumen, etc.
- corrección, escritura de código
- reformato de datos, etc.

Category	Task	Usefulness
Ideation and Feedback	Brainstorming	●
	Feedback	○
	Providing counterarguments	○
Writing	Synthesizing text	●
	Editing text	●
	Evaluating text	●
	Generating catchy titles & headlines	●
	Generating tweets to promote a paper	●
Background Research	Summarizing Text	●
	Literature Research	○
	Formatting References	●
	Translating Text	●
	Explaining Concepts	○
Coding	Writing code	○
	Explaining code	○
	Translating code	●
	Debugging code	○
Data Analysis	Creating figures	○
	Extracting data from text	●
	Reformatting data	●
	Classifying and scoring text	○
	Extracting sentiment	○
Math	Simulating human subjects	○
	Setting up models	○
	Deriving equations	○
	Explaining models	○

The third column reports my subjective rating of LLM capabilities as of September 2023:

○: experimental; results are inconsistent and require significant human oversight

○: useful; requires oversight but will likely save you time

●: highly useful; incorporating this into your workflow will save you time

Table 2: Summary of LLM capabilities and rating of usefulness



Proceso de investigación y automatización

- Definición del problema
 - Formulación del problema
 - Revisión bibliográfica
 - Formulación de hipótesis
- Tareas vinculadas a la recolección de datos
 - Construcción de instrumentos
 - Recolección de datos
- Tareas específicas vinculadas al procesamiento de información
 - Exploración de texto / “Subrayado” de entrevistas
 - Codificación de preguntas abiertas
 - Código de análisis (R, Python, etc.)

Conducting Qualitative Interviews with AI

Abstract

Qualitative interviews are one of the fundamental tools of empirical social science research and give individuals the opportunity to explain how they understand and interpret the world, allowing researchers to capture detailed and nuanced insights into complex phenomena. However, qualitative interviews are seldom used in economics and other disciplines inclined toward quantitative data analysis, likely due to concerns about limited scalability, high costs, and low generalizability. In this paper, we introduce an AI-assisted method to conduct semi-structured interviews. This approach retains the depth of traditional qualitative research while enabling large-scale, cost-effective data collection suitable for quantitative analysis. We demonstrate the feasibility of this approach through a large-scale data collection to understand the stock market participation puzzle. Our 395 interviews allow for quantitative analysis that we demonstrate yields richer and more robust conclusions compared to qualitative interviews with traditional sample sizes as well as to survey responses to a single open-ended question. We also demonstrate high interviewee satisfaction with the AI-assisted interviews. In fact, a majority of respondents indicate a strict preference for AI-assisted interviews over human-led interviews. Our novel AI-assisted approach bridges the divide between qualitative and quantitative data analysis and substantially lowers the barriers and costs of conducting qualitative interviews at scale.

JEL-Codes: C830, C900, D140, D910, Z130.

Keywords: artificial intelligence, interviews, large language models, qualitative methods, stock market participation.

Felix Chopra

*University of Copenhagen / Denmark
felix.chopra@econ.ku.dk*

Ingar Haaland

*NHH Norwegian School of Economics
Bergen / Norway
ingar.haaland@nhh.no*

This version: September 15, 2023

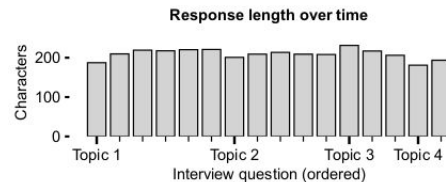
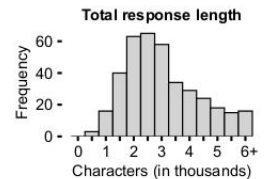
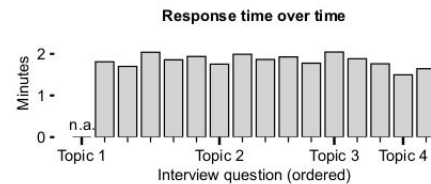
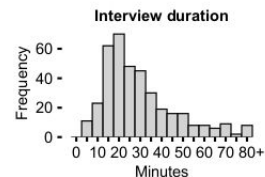
We thank Peter Andre, Christopher Roth, and Johannes Wohlfart for helpful discussions. IRB approval was obtained from the ethics committee of NHH Norwegian School of Economics. The activities of the Center for Economic Behavior and Inequality (CEBI) are financed by the Danish National Research Foundation, Grant DNRF134. Financial support from the Research Council of Norway through its Centre of Excellence Scheme (FAIR project No 262075) is gratefully acknowledged.



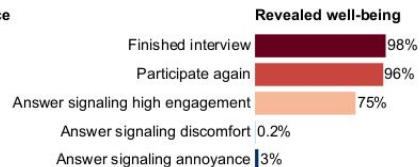
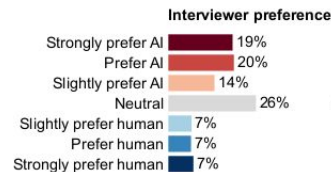
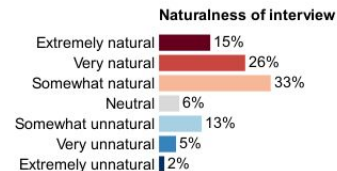
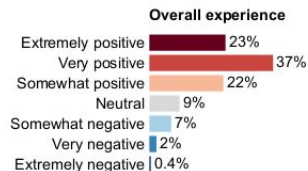
Proceso de investigación y automatización

- Definición del problema
 - Formulación del problema
 - Revisión bibliográfica
 - Formulación de hipótesis
- Tareas vinculadas a la recolección de datos
 - Construcción de instrumentos
 - Recolección de datos
- Tareas específicas vinculadas al procesamiento de información
 - Exploración de texto / “Subrayado” de entrevistas
 - Codificación de preguntas abiertas
 - Código de análisis (R, Python, etc.)

A. Effort and engagement



B. Interview experience



Proceso de investigación y automatización

- Definición del problema
 - Formulación del problema
 - Revisión bibliográfica
 - Formulación de hipótesis
- Tareas vinculadas a la recolección de datos
 - Construcción de instrumentos
 - Recolección de datos
- Tareas específicas vinculadas al procesamiento de información
 - Exploración de texto / “Subrayado” de entrevistas
 - Codificación de preguntas abiertas
 - Código de análisis (R, Python, etc.)



Original Manuscript

Large Language Models Outperform Expert Coders and Supervised Classifiers at Annotating Political Social Media Messages

Social Science Computer Review
2024, Vol. 0(0) 1–15
© The Author(s) 2024



Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/08944393241286471
journals.sagepub.com/home/ssc



Petter Törnberg^{1,2}

Abstract

Instruction-tuned Large Language Models (LLMs) have recently emerged as a powerful new tool for text analysis. As these models are capable of zero-shot annotation based on instructions written in natural language, they obviate the need of large sets of training data—and thus bring potential paradigm-shifting implications for using text as data. While the models show substantial promise, their relative performance compared to human coders and supervised models remains poorly understood and subject to significant academic debate. This paper assesses the strengths and weaknesses of popular fine-tuned AI models compared to both conventional supervised classifiers and manual annotation by experts and crowd workers. The task used is to identify the political affiliation of politicians based on a single X/Twitter message, focusing on data from 11 different countries. The paper finds that GPT-4 achieves higher accuracy than both supervised models and human coders across all languages and country contexts. In the US context, it achieves an accuracy of 0.934 and an inter-coder reliability of 0.982. Examining the cases where the models fail, the paper finds that the LLM—unlike the supervised models—correctly annotates messages that require interpretation of implicit or unspoken references, or reasoning on the basis of contextual knowledge—capacities that have traditionally been understood to be distinctly human. The paper thus contributes to our understanding of the revolutionary implications of LLMs for text analysis within the social sciences.

Keywords

text annotation, Large Language Models, text as data, Twitter, political messages

Proceso de investigación y automatización

Original Manuscript

Large Language Models Outperform Expert Coders and Supervised Classifiers at Annotating Political Social Media Messages

Petter Törnberg^{1,2} 

Abstract

Instruction-tuned Large Language Models (LLMs) have recently emerged as a powerful new tool for text analysis. As these models are capable of zero-shot annotation based on instructions written in natural language, they obviate the need of large sets of training data—and thus bring potential paradigm-shifting implications for using text as data. While the models show substantial promise, their relative performance compared to human coders and supervised models remains poorly understood and subject to significant academic debate. This paper assesses the strengths and weaknesses of popular fine-tuned AI models compared to both conventional supervised classifiers and manual annotation by experts and crowd workers. The task used is to identify the political affiliation of politicians based on a single X/Twitter message, focusing on data from 111 different countries. The paper finds that GPT-4 achieves higher accuracy than both supervised models and human coders across all languages and country contexts. In the US context, it achieves an accuracy of 0.934 and an inter-coder reliability of 0.982. Examining the cases where the models fail, the paper finds that the LLM—unlike the supervised models—correctly annotates messages that require interpretation of implicit or unspoken references, or reasoning on the basis of contextual knowledge—capacities that have traditionally been understood to be distinctly human. The paper thus contributes to our understanding of the revolutionary implications of LLMs for text analysis within the social sciences.

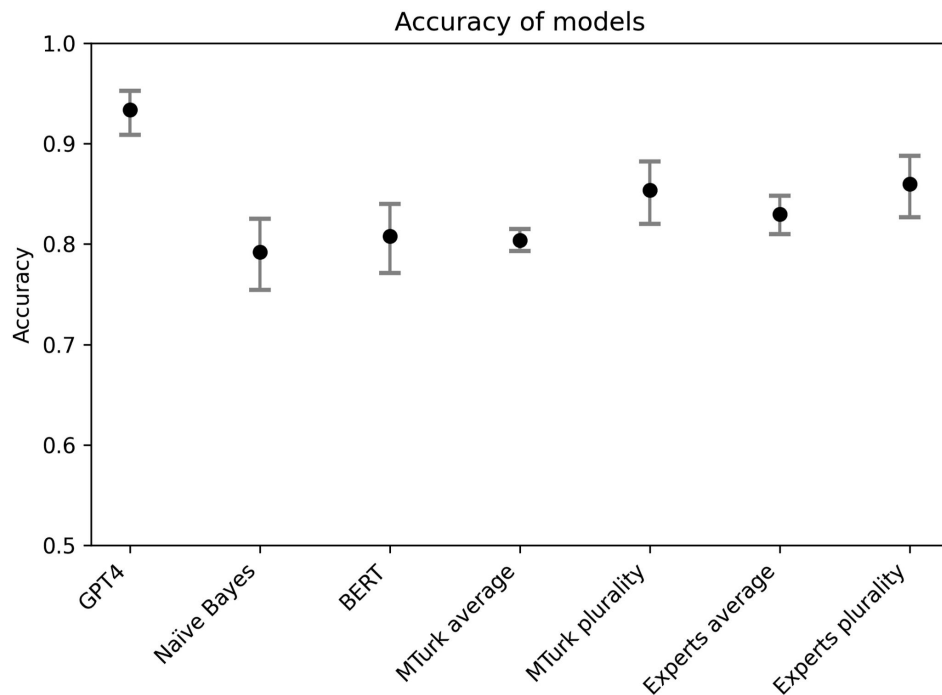
Keywords

text annotation, Large Language Models, text as data, Twitter, political messages

Social Science Computer Review
2024, Vol. 0(0) 1–15
© The Author(s) 2024



Article reuse guidelines:
sagepub.com/journals-permissions
DOI: 10.1177/08944393241286471
journals.sagepub.com/home/ssc



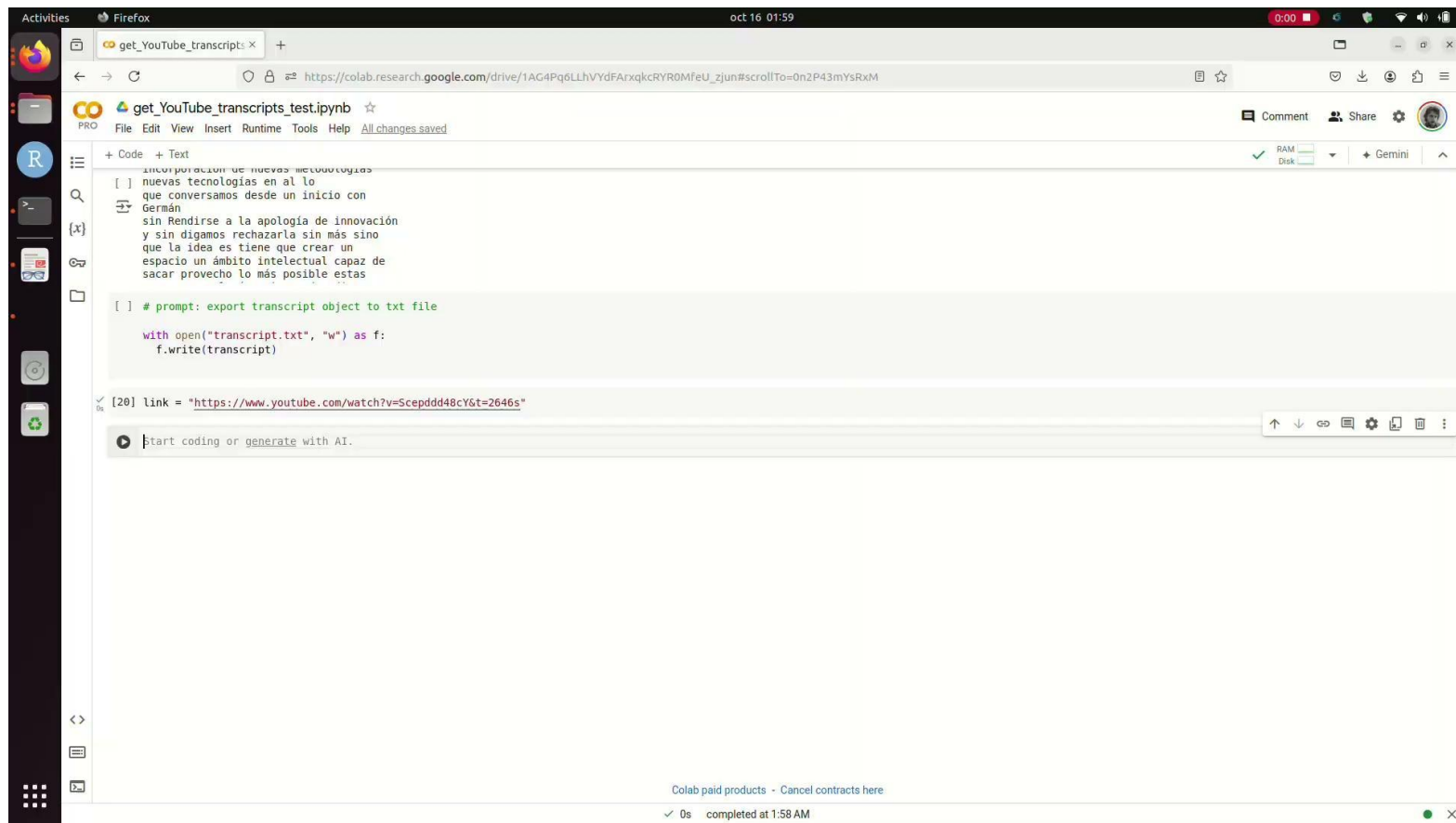
factor-data
EIDAES_UNSAM

Proceso de investigación y automatización

- Definición del problema
 - Formulación del problema
 - Revisión bibliográfica
 - Formulación de hipótesis
- Tareas vinculadas a la recolección de datos
 - Construcción de instrumentos
 - Recolección de datos
- Tareas específicas vinculadas al procesamiento de información
 - Exploración de texto / “Subrayado” de entrevistas
 - Codificación de preguntas abiertas
 - Código de análisis (R, Python, etc.)



Proceso de investigación y automatización



Activities Firefox oct 16 01:59

get_Youtube_transcripts_test.ipynb

File Edit View Insert Runtime Tools Help All changes saved

+ Code + Text

```
incorporación de nuevas metodologías  
[ ] nuevas tecnologías en al lo  
que conversamos desde un inicio con  
Germán  
sin Rendirse a la apología de innovación  
y sin digamos rechazarla sin más sino  
que la idea es tiene que crear un  
espacio un ámbito intelectual capaz de  
sacar provecho lo más posible estas
```

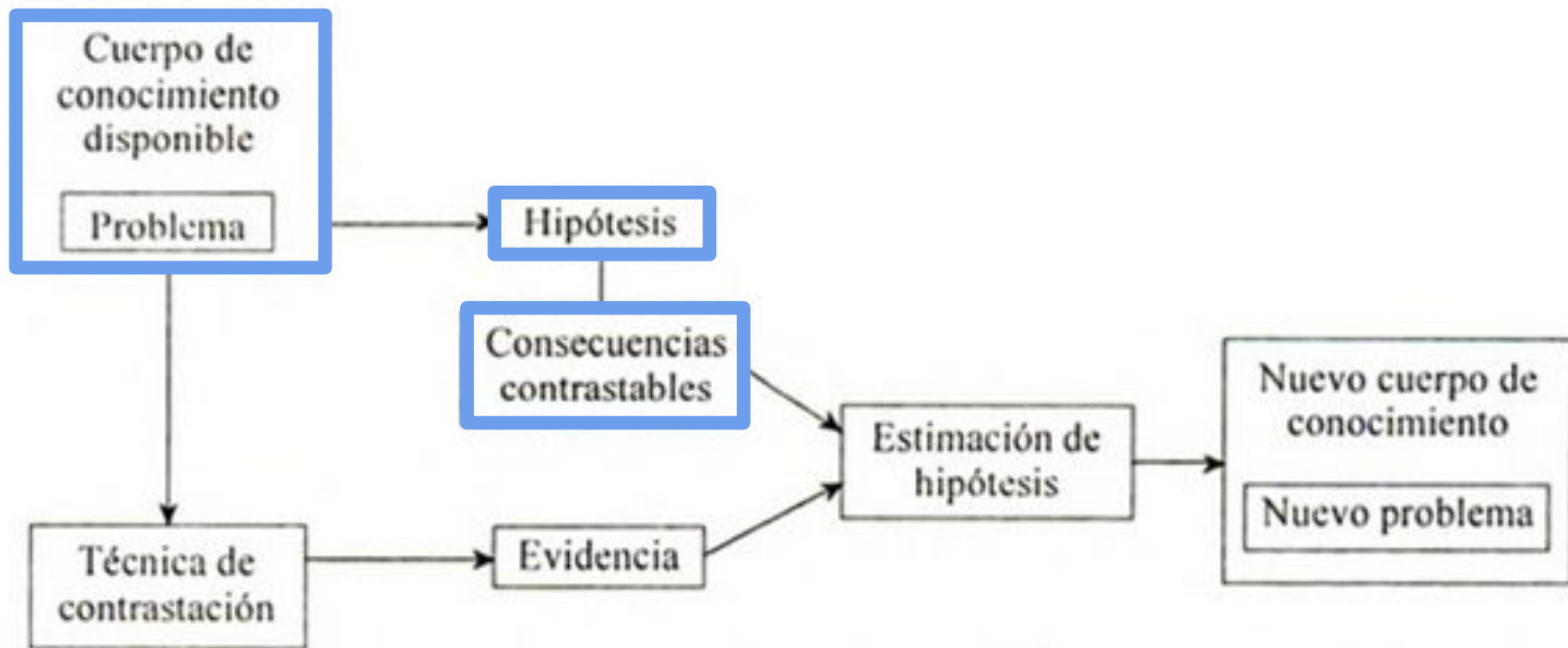
```
[ ] # prompt: export transcript object to txt file  
  
with open("transcript.txt", "w") as f:  
    f.write(transcript)
```

[20] link = "https://www.youtube.com/watch?v=Scepddd48cY6t=2646s"

start coding or generate with AI.

Colab paid products - Cancel contracts here

completed at 1:58 AM



Proceso de investigación y automatización

- Sesgo algorítmico

VS

- Fidelidad algorítmica

Out of One, Many: Using Language Models to Simulate Human Samples

Lisa P. Argyle¹, Ethan C. Busby¹, Nancy Fulda², Joshua Gubler¹, Christopher Rytting², and David Wingate²

¹Department of Political Science, Brigham Young University

²Department of Computer Science, Brigham Young University

September 16, 2022

Abstract

We propose and explore the possibility that language models can be studied as effective proxies for specific human sub-populations in social science research. Practical and research applications of artificial intelligence tools have sometimes been limited by problematic biases (such as racism or sexism), which are often treated as uniform properties of the models. We show that the “algorithmic bias” within one such tool— the GPT-3 language model— is instead both fine-grained and demographically correlated, meaning that proper conditioning will cause it to accurately emulate response distributions from a wide variety of human subgroups. We term this property *algorithmic fidelity* and explore its extent in GPT-3. We create “silicon samples” by conditioning the model on thousands of socio-demographic backstories from real human participants in multiple large surveys conducted in the United States. We then compare the silicon and human samples to demonstrate that the information contained in GPT-3 goes far beyond surface similarity. It is nuanced, multifaceted, and reflects the complex interplay between ideas, attitudes, and socio-cultural context that characterize human attitudes. We suggest that language models with sufficient algorithmic fidelity thus constitute a novel and powerful tool to advance understanding of humans and society across a variety of disciplines.

Contents

1	Introduction	2
2	The GPT-3 Language Model	3
3	Algorithmic Fidelity	4
4	Silicon Sampling: Correcting Skewed Marginals	5
5	Study 1: Free-form Partisan Text	6
6	Study 2: Vote Prediction	10
7	Study 3: Closed-ended Questions and Complex Correlations in Human Data	12

arXiv:2209.06899v1 [cs.LG] 14 Sep 2022

Proceso de investigación y automatización

- Fidelidad algorítmica
- Replicación de otro estudio
- “Muestreo de silicio”
- Generación de un dataset a partir de un perfil ideológico “prototípico” -
- Se le pidió a GPT3 que generara palabras describiendo a demócratas y republicanos

	Describing Democrats	Describing Republicans
Strong Republicans	Ideologically, I describe myself as <u>conservative</u> . Politically, I am a <u>strong Republican</u> . Racially, I am <u>white</u> . I am <u>male</u> . Financially, I am <u>upper-class</u> . In terms of my age, I am <u>young</u> . When I am asked to write down four words that typically describe people who support the <u>Democratic</u> Party, I respond with: 1. <u>Liberal</u> 2. <u>Socialist</u> 3. <u>Communist</u> 4. <u>Atheist</u> .	Ideologically, I describe myself as <u>conservative</u> . Politically, I am a <u>strong Republican</u> . Racially, I am <u>white</u> . I am <u>male</u> . When I am asked to write down four words that typically describe people who support the <u>Republican</u> Party, I respond with: 1. <u>Conservative</u> 2. <u>Male</u> 3. <u>White (or Caucasian)</u> 4. <u>Christian</u> .
Strong Democrats	Ideologically, I describe myself as <u>liberal</u> . Politically, I am a <u>strong Democrat</u> . Racially, I am <u>white</u> . I am <u>female</u> . Financially, I am <u>poor</u> . In terms of my age, I am <u>old</u> . When I am asked to write down four words that typically describe people who support the <u>Democratic</u> Party, I respond with: 1. <u>Liberal</u> 2. <u>Young</u> 3. <u>Female</u> 4. <u>Poor</u> .	Ideologically, I describe myself as <u>extremely liberal</u> . Politically, I am a <u>strong Democrat</u> . Racially, I am <u>hispanic</u> . I am <u>male</u> . Financially, I am <u>upper-class</u> . In terms of my age, I am <u>middle-aged</u> . When I am asked to write down four words that typically describe people who support the <u>Republican</u> Party, I respond with: 1. <u>Ignorant</u> 2. <u>Racist</u> 3. <u>Misogynist</u> 4. <u>Homophobic</u> .

Figure 1. Example contexts and completions from four silicon “individuals” analyzed in Study 1. Plaintext indicates the conditioning context; underlined words show demographics we dynamically inserted into the template; blue words are the four harvested words.



Algunas cuestiones a repensar...

- Distribución similar de palabras entre el estudio original replicado y el que surge de analizar los datos sintéticos de GPT
- Estos “individuos de silicio” pueden servir
 - para probar diferentes formulaciones de preguntas,
 - clasificar diferentes tipos de medidas,
 - identificar relaciones clave para evaluar más de cerca y elaborar planes de análisis antes de recopilar datos con participantes humanos.

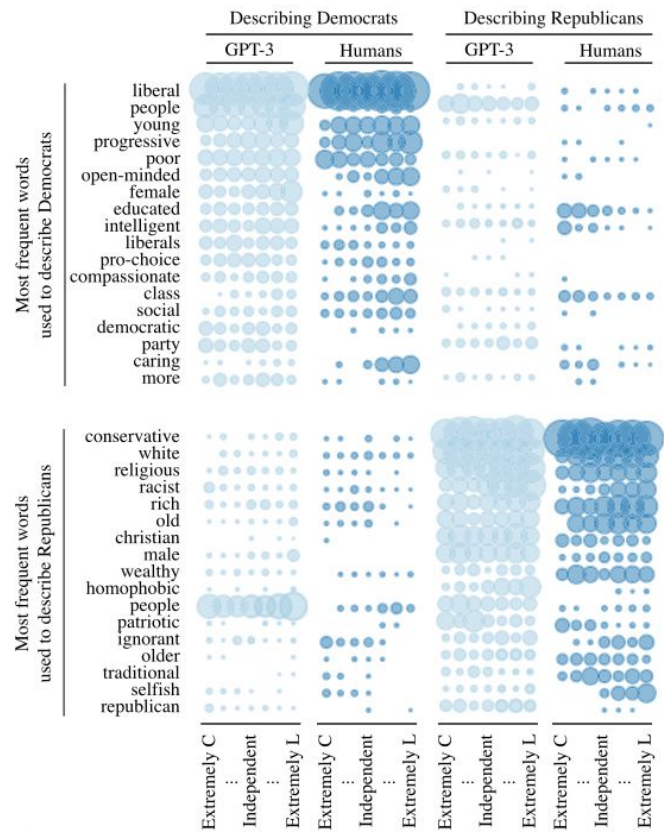



Figure 2. The original Pigeonholing Partisans dataset and the corresponding GPT-3-generated words. Bubble size represents relative frequency of word occurrence; columns represent the ideology of list writers. GPT-3 uses a similar set of words to humans.

Proceso de investigación y automatización

- Definición del problema
 - Formulación del problema
 - Revisión bibliográfica
- Tareas vinculadas a la recolección de datos
 - Construcción de instrumentos
- Tareas específicas vinculadas al procesamiento de información
 - Exploración de texto / “Subrayado” de entrevistas
 - Codificación de preguntas abiertas
 - Código de análisis (R, Python, etc.)

Default (GPT-3.5)

Cada una de las palabras luego de "words" está asociada a percepciones sobre trabajo infantil. Por favor codificá y agrupá cada una de las palabras en categorías conceptuales más generales y devolvé una tabla en la que la primera columna es la categoría y la segunda es la palabra. Además quisiera un código en Python para generar un dict indexado por la categoría que contenga listas de palabras de cada categoría. 

words:

'bronca',
'desigualdad',
'explotación',
'explotación',
'forzado',
'irresponsable',
'casa',
'parque',
'tareas',
'esclavitud',
'mal',
'peor',
'injusticia',
'deshonor',
'abuso',
'injusticia',
'anormal',
'abuso,aprovecharse',
'menores',



Proceso de investigación y automatización

- Definición del problema
 - Formulación del problema
 - Revisión bibliográfica
- Tareas vinculadas a la recolección de datos
 - Construcción de instrumentos
- Tareas específicas vinculadas al procesamiento de información
 - Exploración de texto / “Subrayado” de entrevistas
 - Codificación de preguntas abiertas
 - Código de análisis (R, Python, etc.)



Categoría	Palabras
Sentimientos Negativos	bronca, desigualdad, irresponsable, mal, peor, injusticia, deshonor, abuso, injusto, anormal, mal pago, abuso, sin valores, vergüenza, explotación, abandono, prohibido, abuso, inapropiado, ilegal, abuso, explotación, abuso, explotación, abuso, trata de personas, esclavitud, involuntario, obligación, maltrato, explotación, descuido, furia, irresponsabilidad, injusticia, abuso, desconsideración, violencia, maltrato, explotación, bajos recursos, abuso, deshonestidad, ilegal, explotación, incoherencia, intolerancia, degradación, explotación, inocencia, necesidad, explotación, disponibilidad, explotación, injusta, innecesario, inmoral, insalubre, repudiable, prohibido, inseguro, inexperiencia, explotación, desigualdad, horror, degradación, imposible, explotación, aprovechamiento, vulnerabilidad, sobre explotación, insalubre, malas condiciones, locura, abuso, falta de consideración, explotación, tristeza, infancia, explotación, maltrato, falta de oportunidades, mal, impotencia, negligencia, ilegal, abuso, esclavitud, aprovechamiento, exclusión, pobreza, abuso, explotación, agrotexil, explotación, abuso, explotación, ayuda, inconsciencia, trabajo en negro, familia, pedir en la calle, abuso, maltrato, vulnerabilidad, explotación, denigración, ni idea
Actividades	casa, parque, tareas, jardín, guarderías, pequeños, familia, ganas, estudio, poblamiento, padres, educación, juego, diversión, comer, inclusión, enseñanza, esfuerzo, juego, educación, aprendizaje, formación, remuneración, costumbres, chico, educación
Conceptos Negativos	esclavitud, abuso, explotación, abuso, menores, abuso, explotación, injusticia, menor, mal, maltrato, explotación, explotación, mal, explotación, mal, explotación, derechos del niño, trabajo familiar, incorrecto, prohibido, explotación, derechos, explotación, ilegal, no está de acuerdo, mal, ayuda, aprendizaje, necesidad, capacitarse, orientación, tecnología, injusticia, explotación, desconsideración, presión, explotación, economía regional, explotación, explotación, explotación, incoherencia, intolerancia, explotación, necesidad, explotación, inexperiencia
Conceptos Positivos	aprendizaje, cuidado, ayuda, está bien, colaboración, respeto, tecnología, inclusión, ayuda, enseñanza, responsabilidad, necesidad, usos, aprendizaje, está bien, importante, sobre explotación, paciencia, como un juego, educación, familia, educación
Tipos de Trabajo	casa, tareas, trabajo familiar, agrotexil, pedir en la calle

Proceso de investigación y automatización

- Definición del problema
 - Formulación del problema
 - Revisión bibliográfica
- Tareas vinculadas a la recolección de datos
 - Construcción de instrumentos
- Tareas específicas vinculadas al procesamiento de información
 - Exploración de texto / “Subrayado” de entrevistas
 - Codificación de preguntas abiertas
 - Código de análisis (R, Python, etc.)



Sos un codificador experto de datos cuantitativos y cualitativos. A continuación deberás clasificar diferentes respuestas de una encuesta sobre el uso de ChatGPT.

Cada respuesta estará delimitada con "####".

Clasificá cada respuesta según alguna de las siguientes categorías:

["Búsqueda de datos/bibliográfica", "Generación/corrección de código y programación", "Escritura de artículos, papeles, etc", "Traducción", "No lo he usado", "Preparar clases y cursos", "Generación de ideas/Brainstorming", "Usos recreativos", "Resumir textos", "Mejorar el estilo de escritura"]. Si no sabés clasificar, solamente poné "Sin datos". No fuerces la clasificación.

A su vez, el formato de salida deberá ser el dict de python debería ser el siguiente:

```
{mail:"", text:"", classif_1:"", classif_2:""....}
```

Es decir, un mail, el texto y una key para cada uso que encuentres.

A continuación paso la primera lista:

██████████ No lo he usado lo suficiente

####

██████████ Lo uso como buscador y como asistente de corrección de normas APA

####

██████████ asistente para escribir texto y código. re-escritura de texto para mayor claridad, expresiones regulares, tormenta de ideas,

####

██████████

####

caridelu@gmail.com No lo he usado lo suficiente

####

██████████ Armado de gráficos en LaTeX, pasaje de citas de un formato a otro, para que me invente títulos para los trabajos

####

██████████ Para escribir introducciones, para mejorar algo en inglés, para escribir correos, para buscar variantes en mi escritura, para programar.

####

██████████ Como asistente para escribir código

####

██████████ Me ayuda a consultas sobre programación en R. También lo uso para mejorar escritura de e-mails en inglés.

####

Loros y ciencias sociales

```
[39] dict_1 = {  
    "text": "No lo he usado lo suficiente",  
    "No lo he usado": "No lo he usado lo suficiente"  
},  
    {  
    "text": "Lo uso como buscador y como asistente de corrección de normas APA",  
    "Búsqueda de datos/bibliográfica": "Lo uso como buscador",  
    "Generación/corrección de código y programación": "Como asistente de corrección de normas APA"  
},  
    {  
    "text": "Asistente para escribir texto y código. Re-escritura de texto para mayor claridad, expresiones regulares, tormenta de ideas",  
    "Generación/corrección de código y programación": "Asistente para escribir código",  
    "Escritura de artículos, papeles, etc.": "Asistente para escribir texto",  
    "Generación de ideas/Brainstorming": "Tormenta de ideas"  
},  
    {  
    "text": "D",  
    "Sin datos": "D"  
},  
    {  
    "text": "No lo he usado lo suficiente",  
    "No lo he usado": "No lo he usado lo suficiente"  
},  
    {  
    "text": "Armado de gráficos en LaTeX, pasaje de citas de un formato a otro, para que me invente títulos para los trabajos",  
    "Búsqueda de datos/bibliográfica": "Pasaje de citas de un formato a otro",  
    "Escritura de artículos, papeles, etc.": "Armado de gráficos en LaTeX",  
    "Generación de ideas/Brainstorming": "Para que me invente títulos para los trabajos"  
},  
    {  
    "text": "Para escribir introducciones, para mejorar algo en inglés, para escribir correos, para buscar variantes en mi escritura, para programar.",  
    "Escritura de artículos, papeles, etc.": "Para escribir introducciones",  
    "Mejorar el estilo de escritura": "Para mejorar algo en inglés, para buscar variantes en mi escritura",  
    "Generación/corrección de código y programación": "Para programar"  
}
```



Algunas cuestiones para cerrar...

- Opacos
- Muchos son cerrados y propietarios
- Otros no...
- Usos: no son útiles para cualquier cosa



```
grosati@rsix: /media/grosati/Data/llama.cpp

== Running in interactive mode. ==
- Press Ctrl+C to interject at any time.
- Press Return to return control to LLaMa.
- To return control without starting a new line, end your input with '/'.
- If you want to submit another line, end your input with '\'.

Text transcript of a never ending dialog, where USER interacts with an AI assistant named ChatLLaMa.
ChatLLaMa is helpful, kind, honest, friendly, good at writing and never fails to answer USER's requests immediately and with details and precision.
There are no annotations like (30 seconds passed...) or (to himself). Just what USER and ChatLLaMa say aloud to each other.
The dialog lasts for years, the entirety of it is shared below. It's 10000 pages long.
The transcript only includes text, it does not include markup like HTML and Markdown.

USER: Hello, ChatLLaMa!
ChatLLaMa: Hello USER! How may I help you today?
USER: What year is it?
ChatLLaMa: We are in 2023.
USER: Please tell me the largest city in Europe.
ChatLLaMa: The largest city in Europe is Moscow, the capital of Russia.
USER: What can you tell me about Moscow?
ChatLLaMa: Moscow, on the Moskva River in western Russia, is the nation's cosmopolitan capital. In its historic core is the Kremlin, a complex that is the home to the president and tsarist treasures in the Armoury. Outside its walls is Red Square, Russia's symbolic center.
USER: What is a cat?
ChatLLaMa: A cat is a domestic species of small carnivorous mammal. It is the only domesticated species in the family Felidae.
USER: How do I pass command line arguments to a Node.js program?
ChatLLaMa: The arguments are stored in process.argv.

    argv[0] is the path to the Node.js executable.
    argv[1] is the path to the script file.
    argv[2] is the first argument passed to the script.
    argv[3] is the second argument passed to the script and so on.
USER: Name a color.
ChatLLaMa: Blue.
USER: What time is it?
ChatLLaMa: It is 23:43.
USER: Please, perform sentiment analysis on the quoted sentence, using a discrete scale ranging from 1 (very negative) to 5 (very positive). Do not return anything other from the score: only the number, no explanations at all.
"I loved that movie"
```



Algunas cuestiones para cerrar...

- NLP “Del giro lingüístico al giro (lingüístico) computacional”.
- Posibilidades metodológicas para las ciencias sociales
- Discusiones “no metodológicas” que suscitan

Innatismo o no del lenguaje (discusión con Chomsky)

“Los límites de mi mundo son los de mi lenguaje”.

