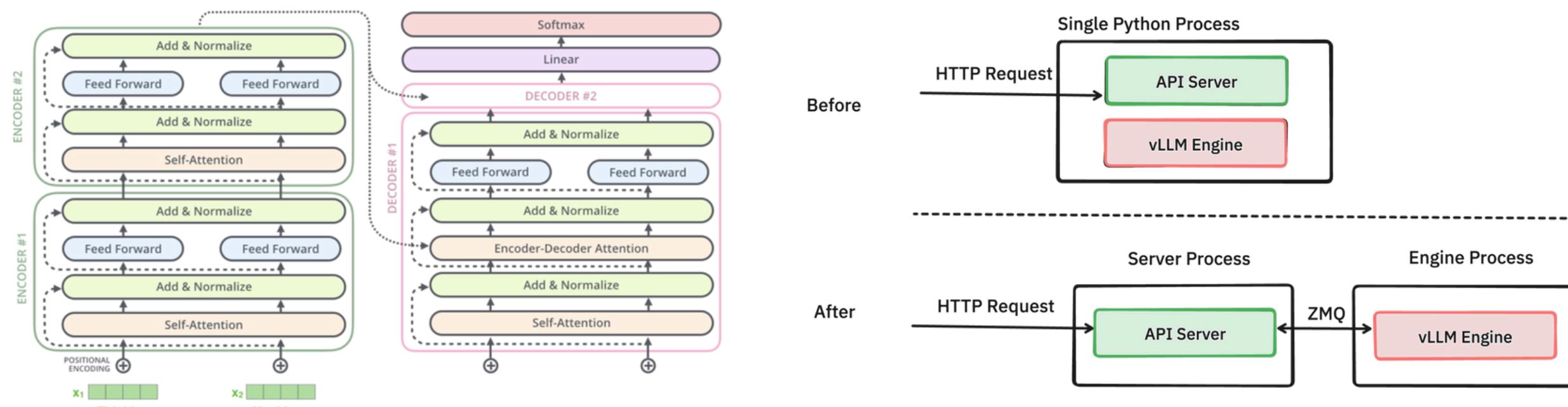


USING LLMS AT SCALE

Jimena Royo Letelier

Research Engineer
médialab - Sciences Po

RésIn project Winter School
October 18th, 2025



European
Polarisation
Observatory

SciencesPo
MÉDIALAB

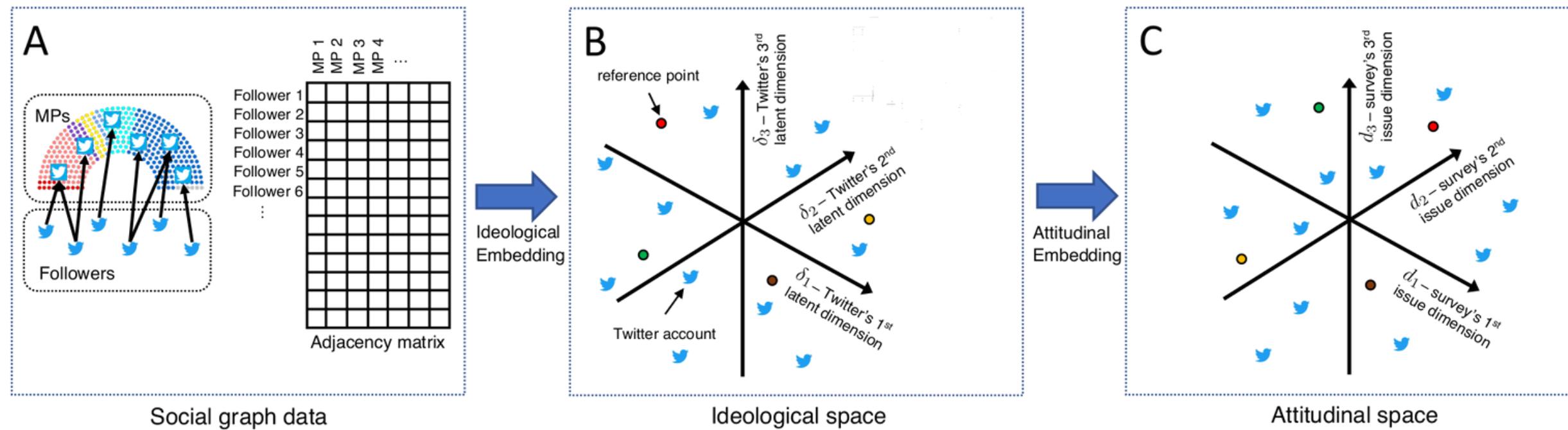
RÉSIN
RÉSEAU D'INGÉNIEUR·ES

USING LLMS AT SCALE

1. How big is fast ?
European Polarisation Observatory example
2. Research on using LLMs to annotate political-related content
3. LLM based annotations
4. LLMs optimizations and serving frameworks
5. Benchmarks and practical implementations
6. Advanced Topics
 - . Structured outputs
 - . Async serving
 - . OpenAI API compatibility
 - . Docker serving

Social Network's embeddings

European Polarisation Observatory



Unfolding the dimensionality structure of social networks
in ideological embeddings

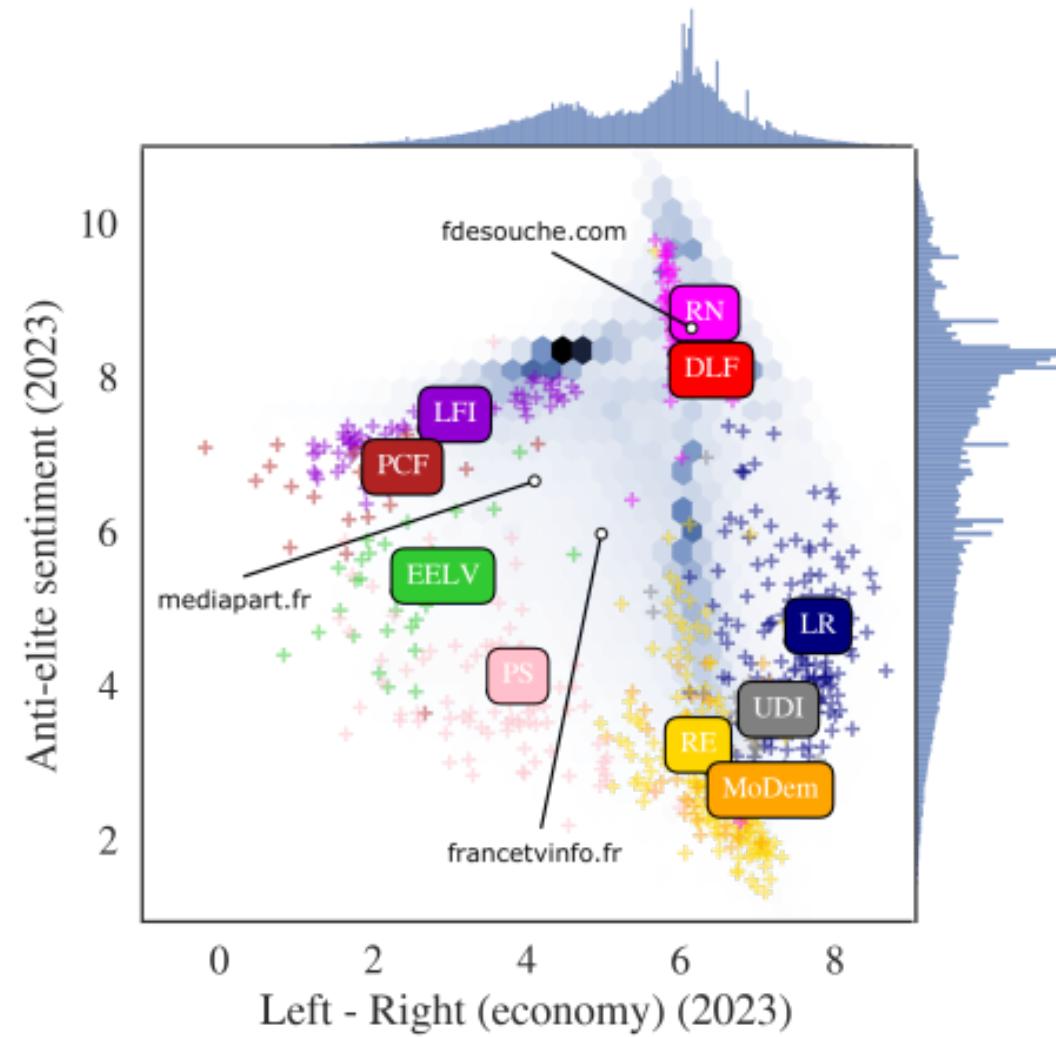
Ramaciotti et all

<https://hal.science/hal-03315759v3/document>

- Digital behavioral traces (typically following/follower networks) in social media platforms can be used to mine opinions at a massive scale.
- Observe ideologies and attitudes towards issues of the public debate (e.g., taxation, immigration, European integration, perception of elites).
- Study the ideological dynamics of users and media outlets

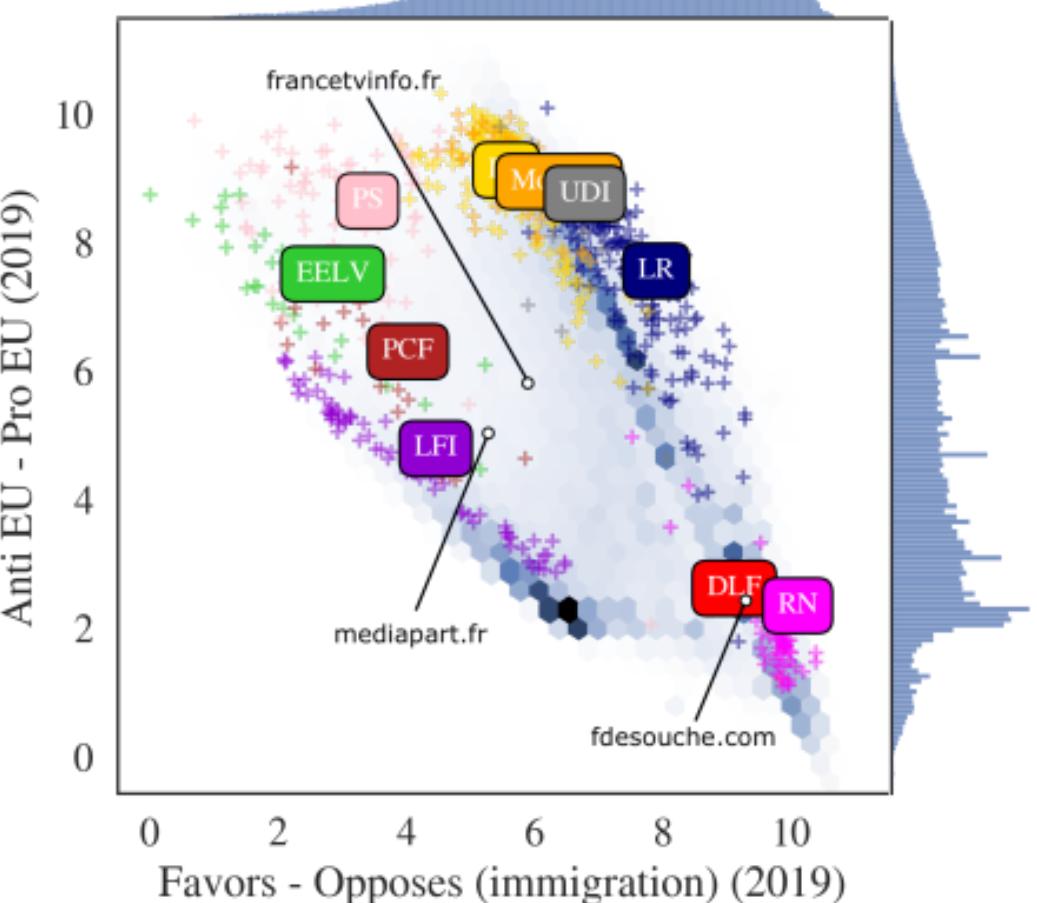
Social Network's embeddings

European Polarisation Observatory



Mapping the Digital Political Landscape: A
Multidimensional dataset of X users, Politicians, and
Media Outlets in Opinion Spaces

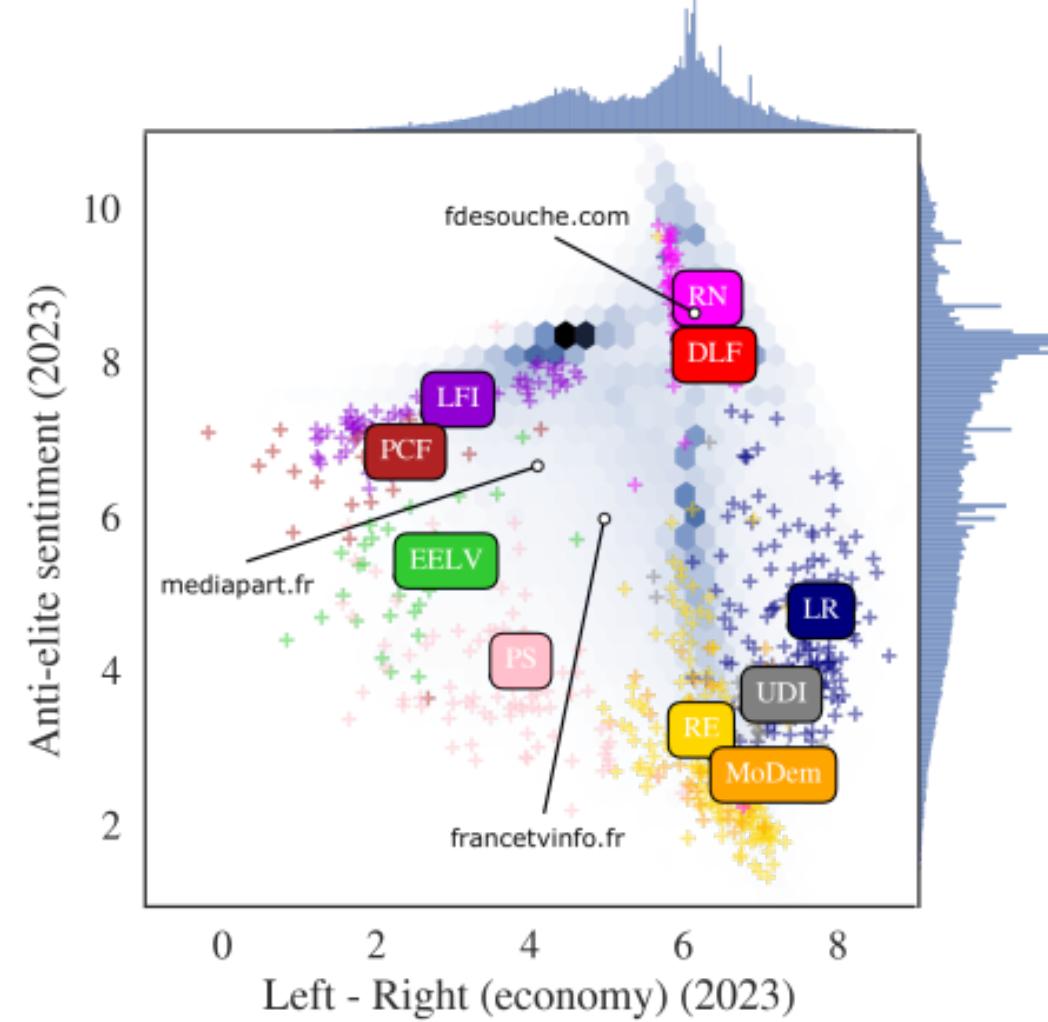
Vendeville et all
<https://hal.science/hal-05222448>



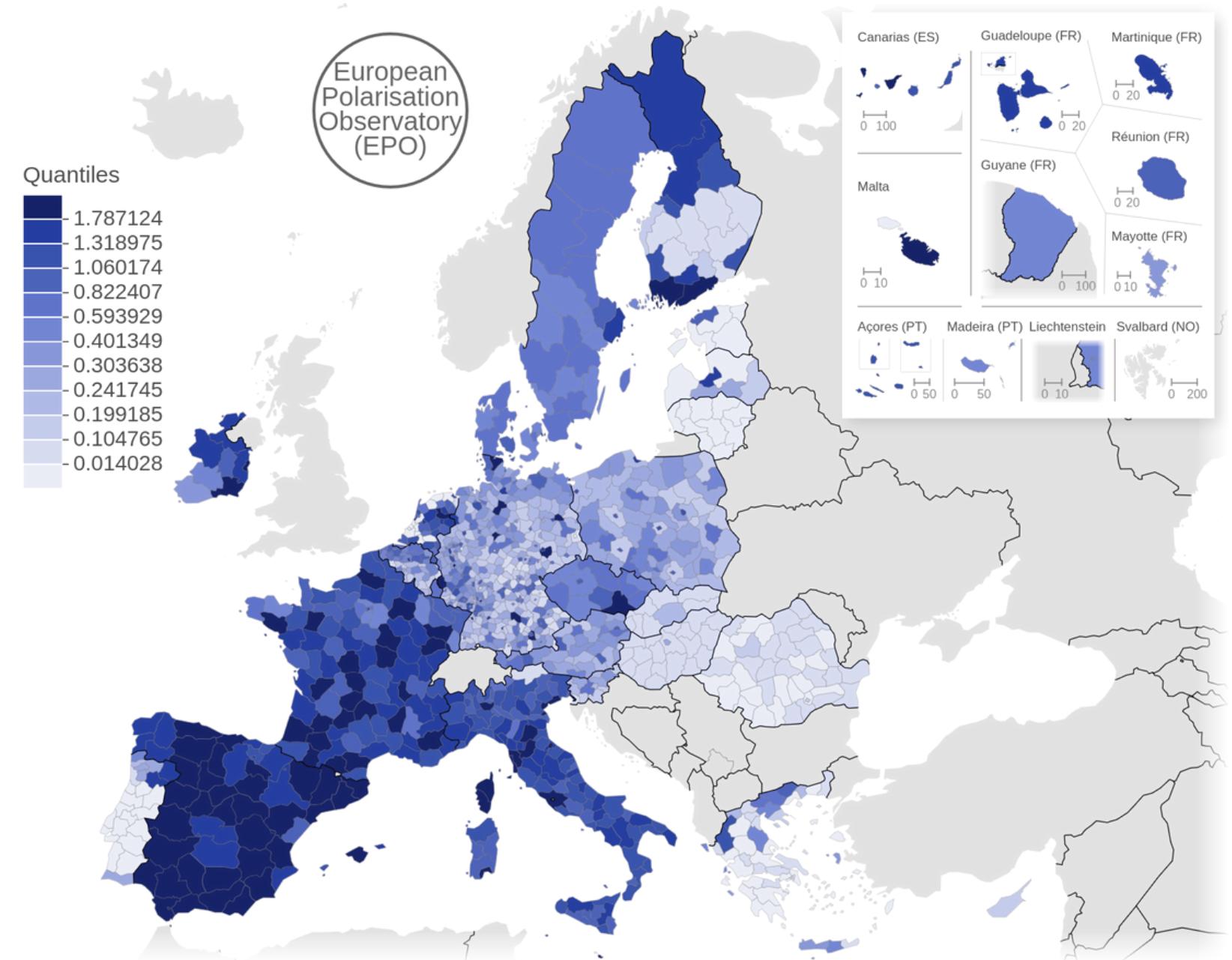
- Digital behavioral traces (typically following/follower networks) in social media platforms can be used to mine opinions at a massive scale.
- Observe ideologies and attitudes towards issues of the public debate (e.g., taxation, immigration, European integration, perception of elites).
- Study the ideological dynamics of users and media outlets

Social Network's embeddings

European Polarisation Observatory



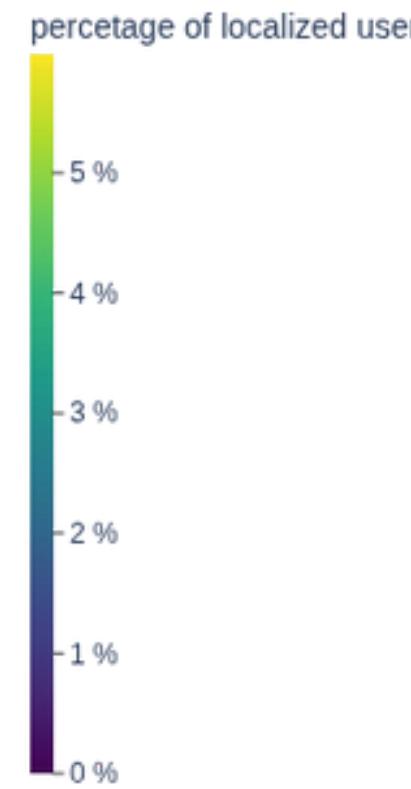
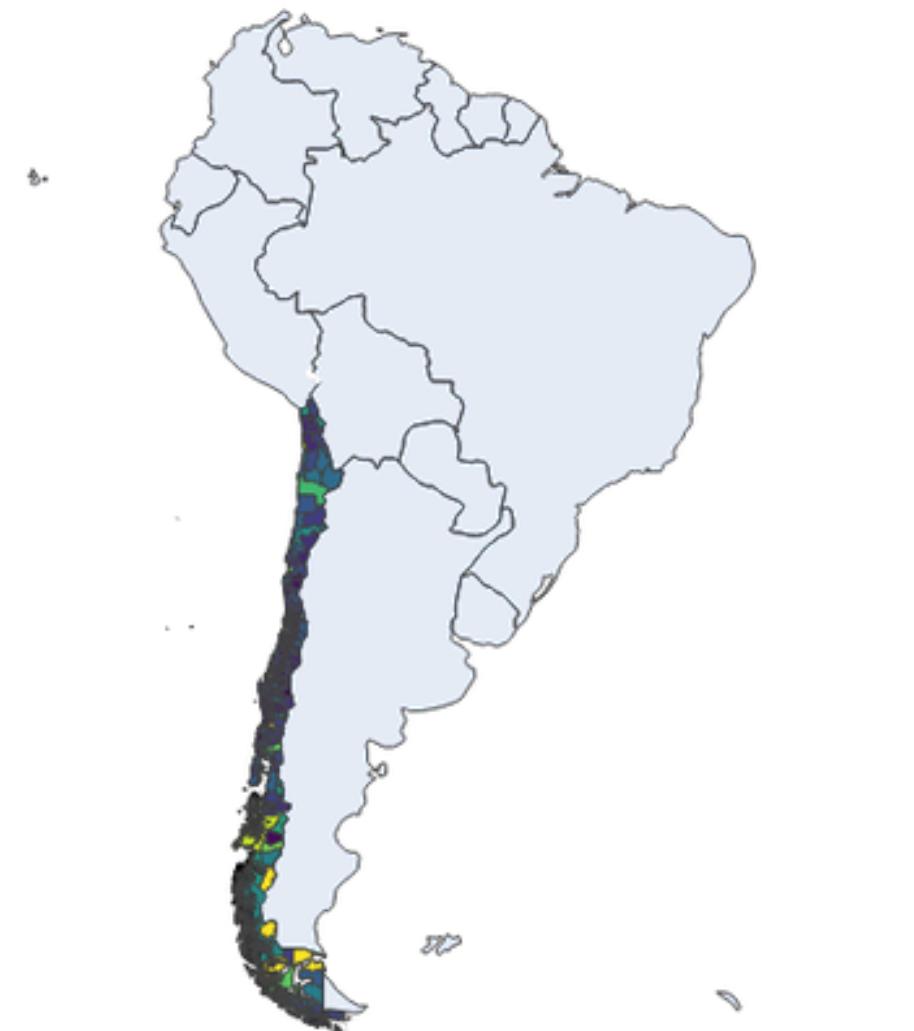
Percentage of X/twitter users
2024 NUTS 3 level - 2023 data collection



+ Geolocation inference

Social Network's embeddings

European Polarisation Observatory



Social Media Census project



SomeCens package could be use to:

- Create a Demograph tree object representing a country and its hierarchical organized geographic units
- Load sociodemographic (age and gender) information for units
- Localize user in units
- Create choropleth maps to show sociodemographic and localisation data
- Export tables with localized user and units sociodemographic data

Currently used with data from

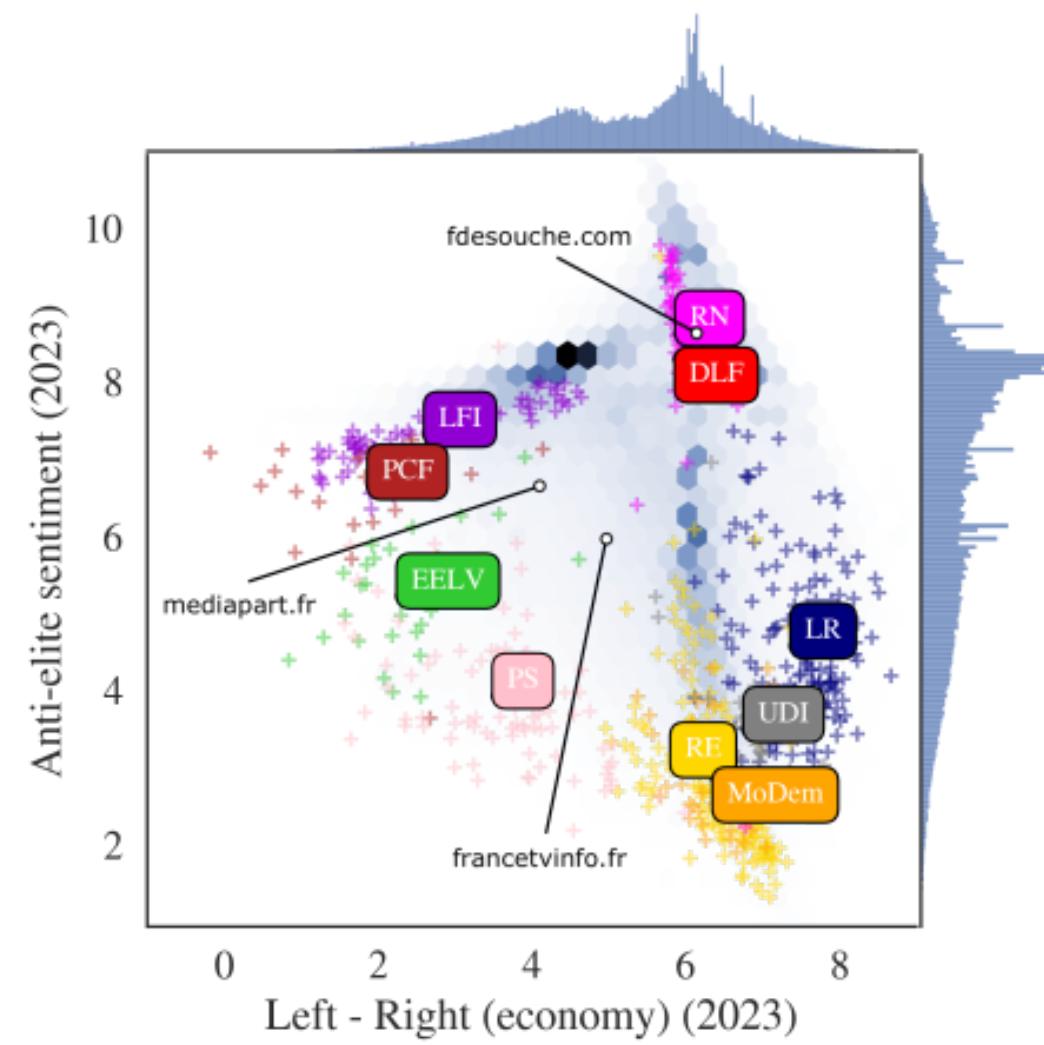
Country	Census
EU countries	NUTS 2024 classification
Chile	2024 Census
USA	2023 Annual County Resident Population Estimate

Check the provided [scripts](#) for data formatting.

+ demographic attributed inference

Social Network's embeddings

European Polarisation Observatory

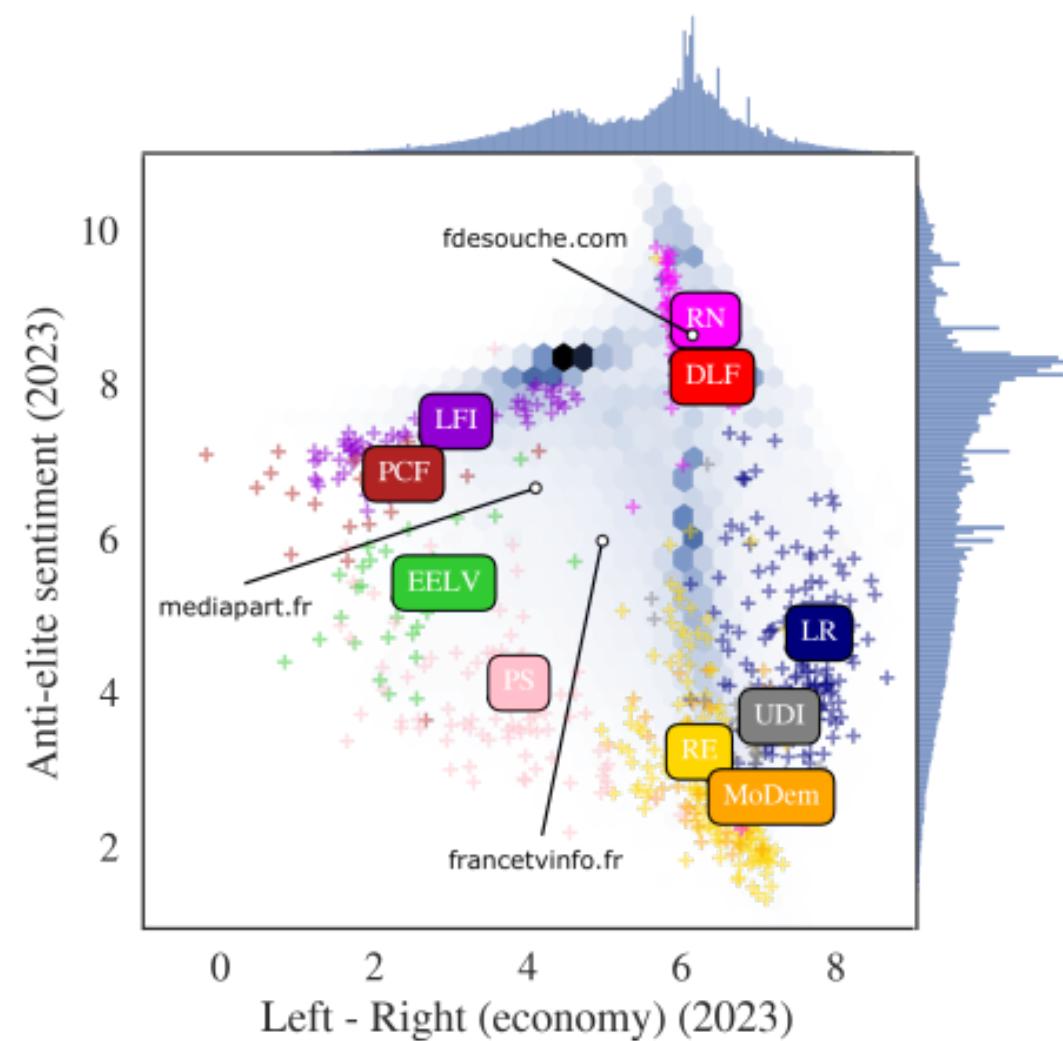


3 snapshots 2020 - 2023 - 2025
for a total of ~ 100 countries-year combinations

country	STATS			
	# lut users	# pp graph mps	# pp graph followers	# pp metadata users
us	36 004 856	522	10 926 871	5 581 233
turkey	22 555 696	586	8 899 430	2 703 836
brazil	17 703 352	528	6 142 272	2 834 581
mexico	9 506 524	534	2 119 744	940 393
uk	8 931 021	450	2 007 465	1 271 484
canada	8 121 494	390	786 199	472 830
japan	6 938 309	619	1 488 547	913 939
italy	6 845 693	469	1 939 329	657 098
southafrica	5 990 490	221	1 733 009	975 386
venezuela	5 492 217	198	1 182 342	582 151
spain	5 469 827	468	1 723 368	864 887
france	5 170 582	880	978 209	539 677
argentina	5 046 873	266	1 410 573	709 208
colombia	4 590 157	266	1 322 463	701 803
germany	3 411 973	676	1 168 788	456 169
chile	2 668 393	179	843 424	462 440
israel	2 622 833	98	407 280	135 692
australia	2 066 824	188	554 036	288 574
poland	1 990 419	468	732 201	240 578
nigeria	1 937 980	227	430 730	299 596
netherlands	1 881 448	197	375 696	199 235
ireland	1 138 341	208	328 014	176 948
finland	1 125 774	179	270 596	124 272
greece	1 087 081	142	355 653	110 974
newzealand	1 000 563	102	84 612	53 594
czechia	873 722	201	228 636	75 490
austria	687 905	142	134 872	52 210
sweden	653 044	244	152 179	77 869
ecuador	634 320	105	117 339	73 806
norway	608 291	124	141 721	67 067

Social Network's embeddings

European Polarisation Observatory



Mapping the Digital Political Landscape...

Vendeville et all

<https://hal.science/hal-05222448>

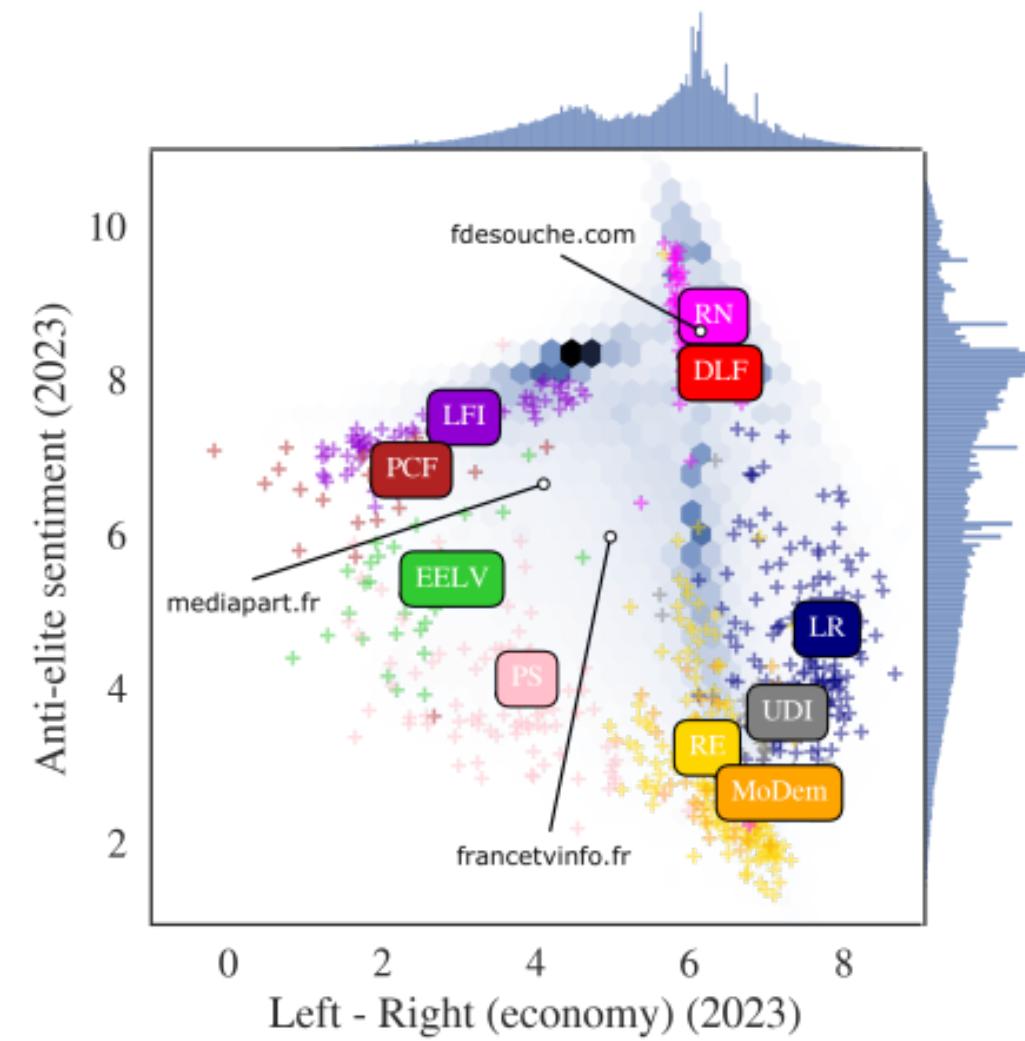
Data is available at
<https://doi.org/10.21410/7E4/QPECFF>

Row n°2
label redistributionist choices Redistributionist Not-Redistributionist Unknown user_prompt Please classify the following Twitter profile bio as "Redistributionist", "Not-Redistributionist" or "Unknown" according to whether the author of the text (who is from \${country}) advocates for more taxation, more regulation, more government spending, or increasing welfare state or not. Be concise and answer only "Redistributionist", "Not-Redistributionist" or "Unknown": "\${userbio}"
Row n°3
label anti_redistributionist choices Not-Redistributionist Redistributionist Unknown user_prompt Please classify the following Twitter profile bio as "Not-Redistributionist", "Redistributionist" or "Unknown" according to whether the author of the text (who is from \${country}) advocates for less taxation, less regulation, less government spending, or reducing welfare state or not. Be concise and answer only "Anti-Redistributionist", "Not-Anti-Redistributionist" or "Unknown": "\${userbio}"
Row n°4
label populist choices Populist Not-Populist Unknown user_prompt Please classify the following Twitter profile bio as "Populist", "Not-Populist" or "Unknown" according to whether the author of the text (who is from \${country}) holds populist views or not. Populist views include, among other, believing that society is split between the people and elites, or that political elites are corrupt. Be concise and answer only "Populist", "Not-Populist" or "Unknown": "\${userbio}"
Row n°5
label elite choices Elite Not-Elite Unknown user_prompt Please classify the following Twitter profile bio as "Elite", "Not-Elite" or "Unknown" according to whether the author of the text (who is from \${country}) belongs to an elite group, including political or economic elites. Be concise and answer only "Elite", "Not-Elite" or "Unknown": "\${userbio}"

lLM-based annotations for validation

Social Network's embeddings

European Polarisation Observatory



Mapping the Digital Political Landscape...

Vendeville et all

<https://hal.science/hal-05222448>

Data is available at
<https://doi.org/10.21410/7E4/QPECFF>

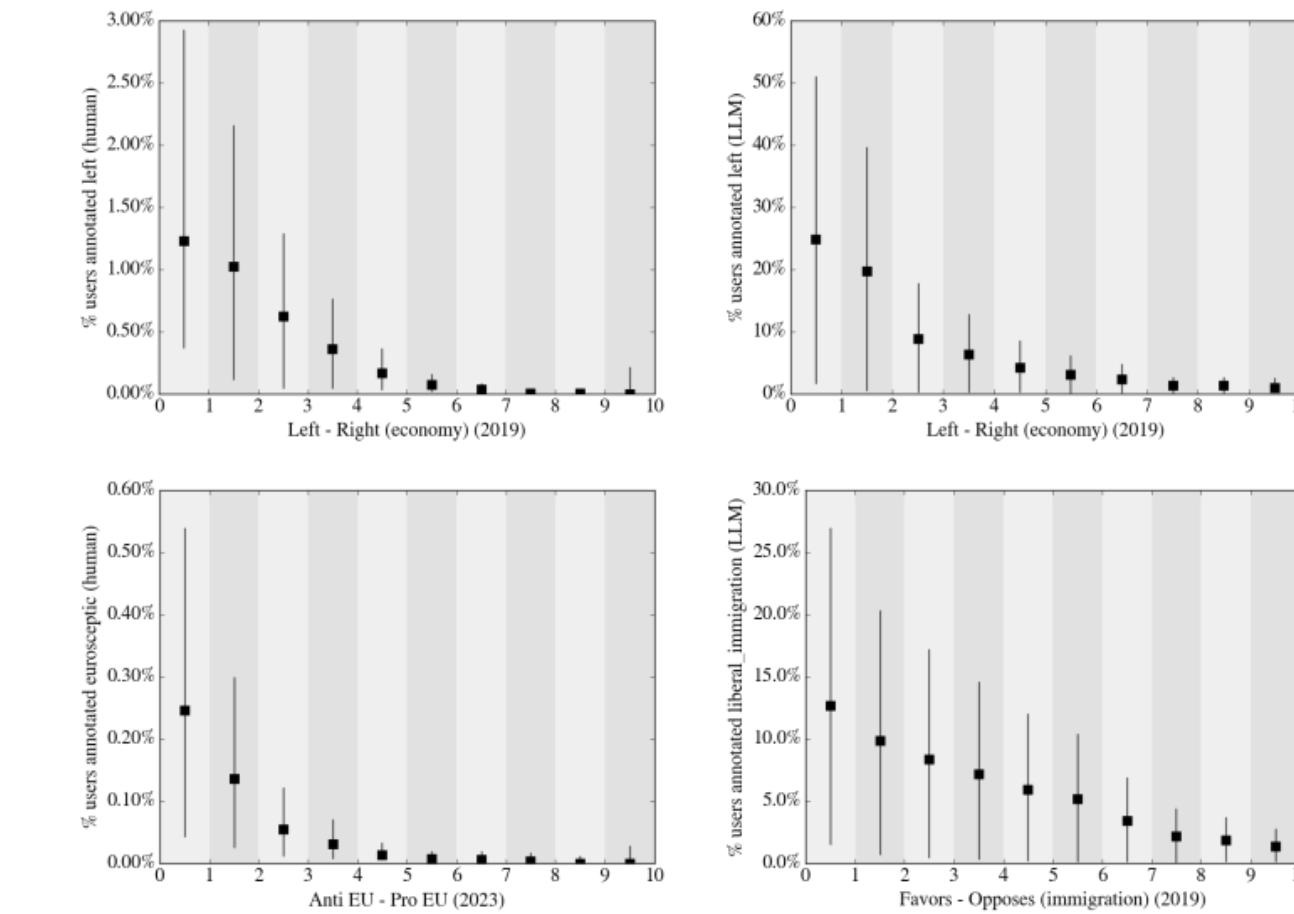
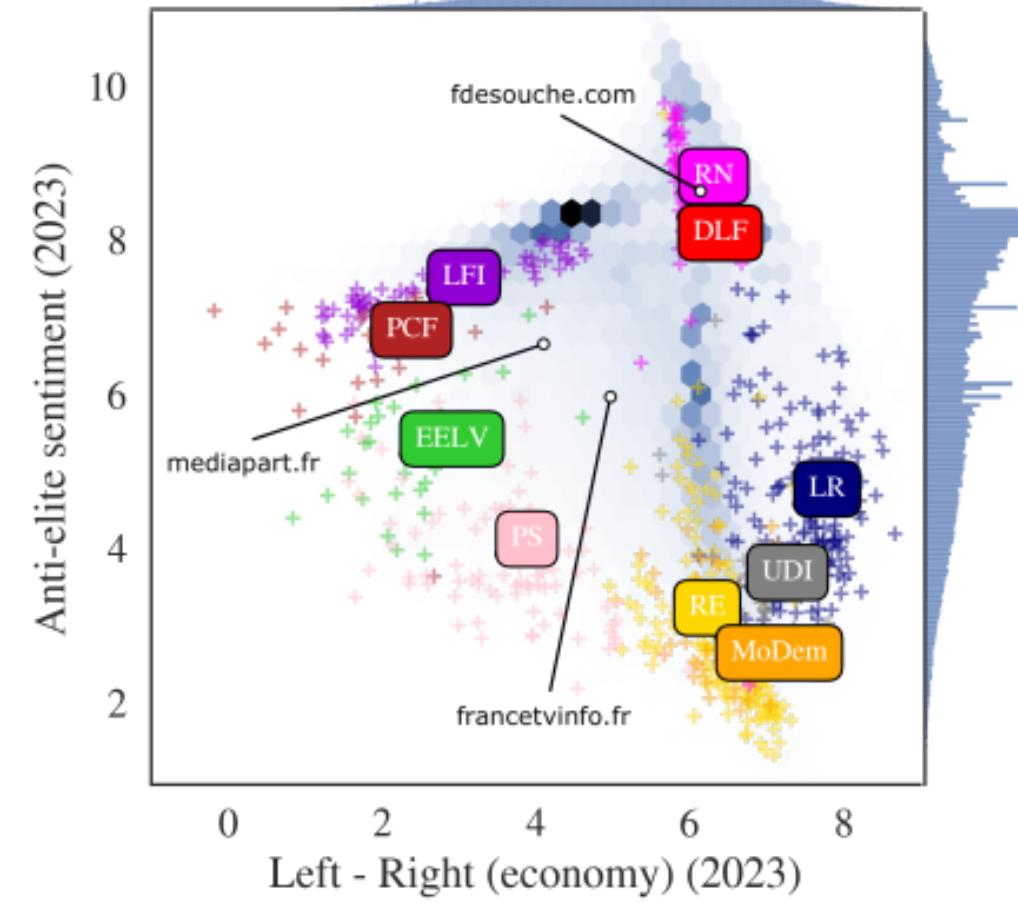


Figure 2: Proportion of users having a given label indicating ideological or issue stances, as provided by both human and LLM annotations, along the corresponding CHES dimensions they intend to validate. Vertical bars delimit Clopper-Pearson confidence intervals [83] at the $\alpha = 0.05$.

llm-based annotations for validation

Social Network's embeddings

European Polarisation Observatory

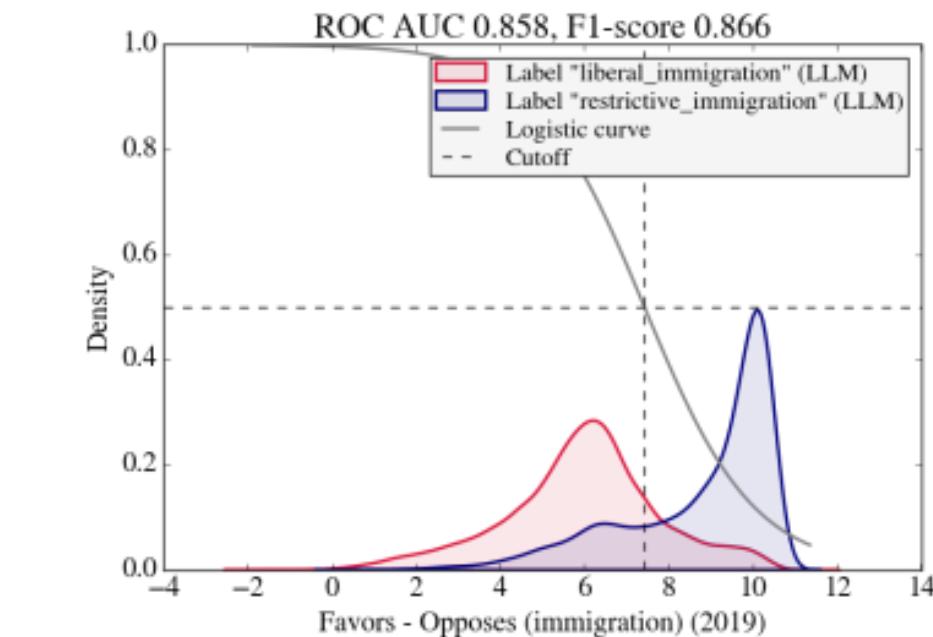
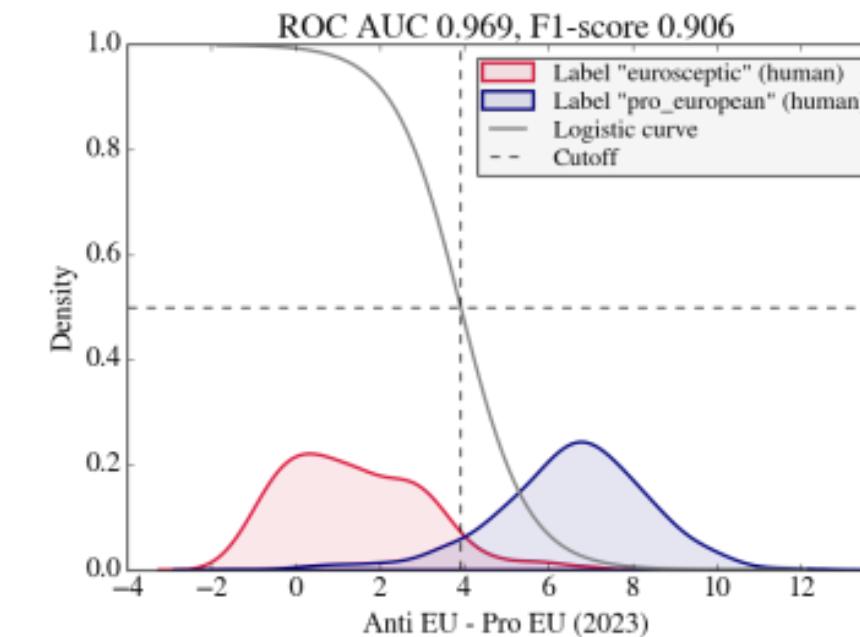
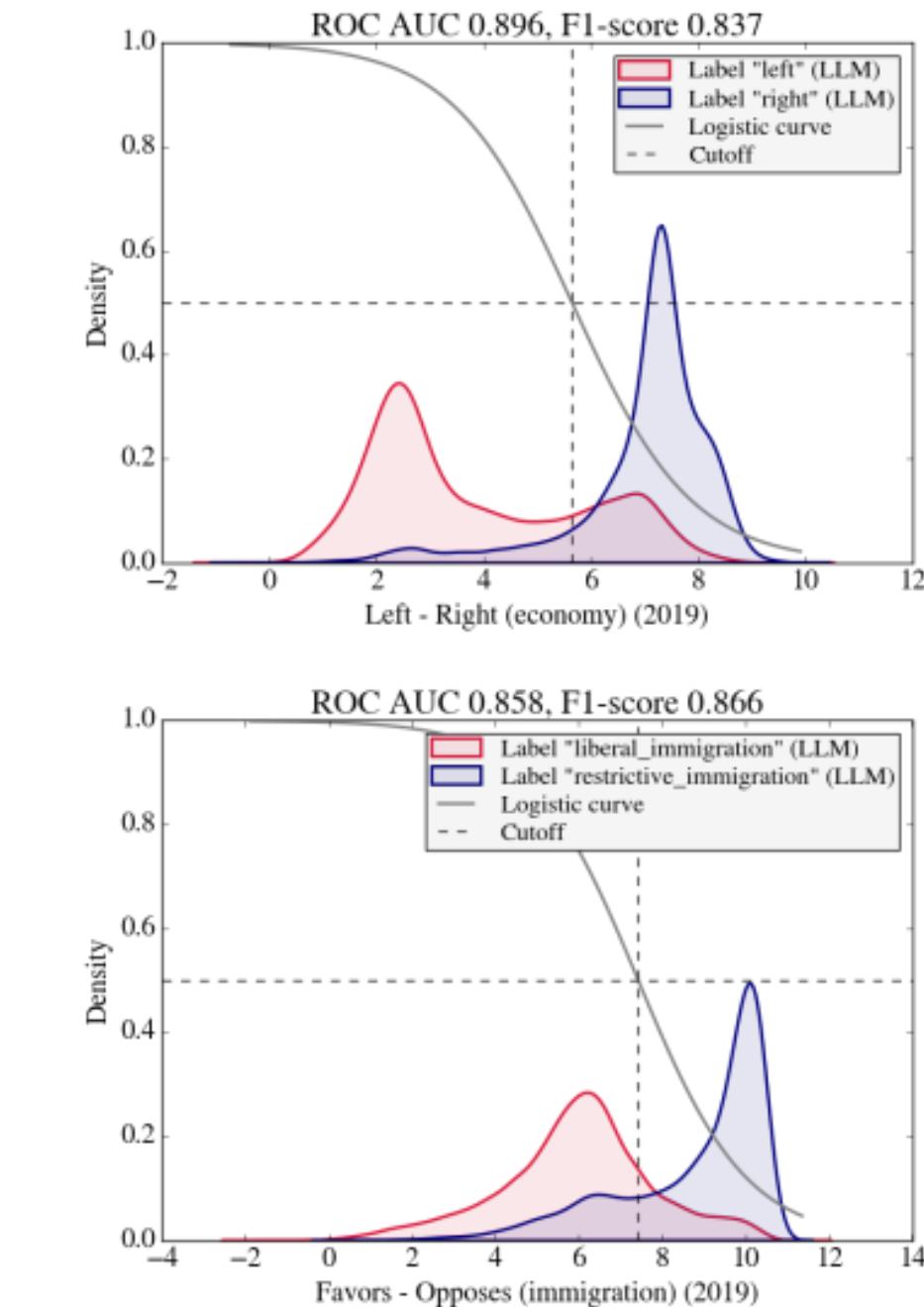
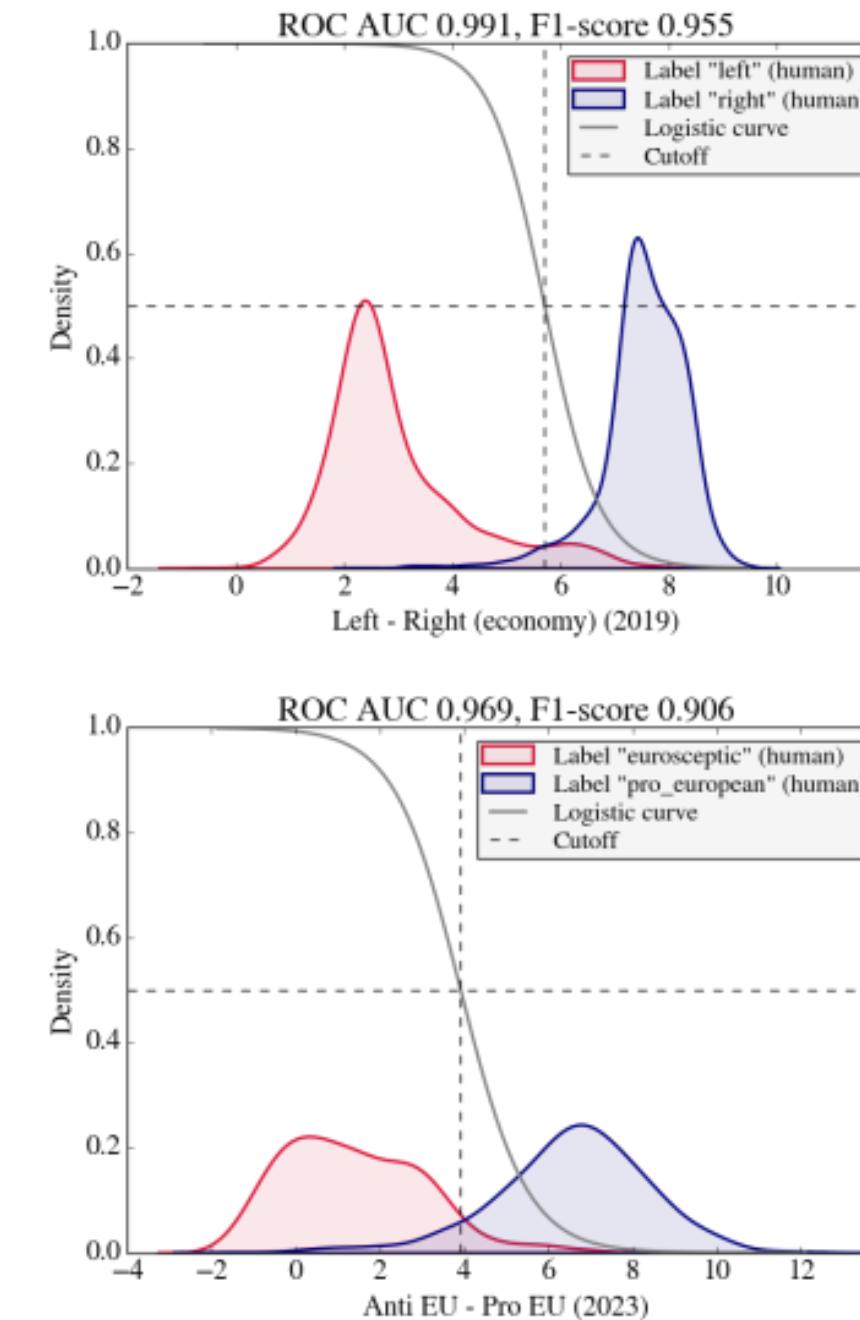


Mapping the Digital Political Landscape...

Vendeville et all

<https://hal.science/hal-05222448>

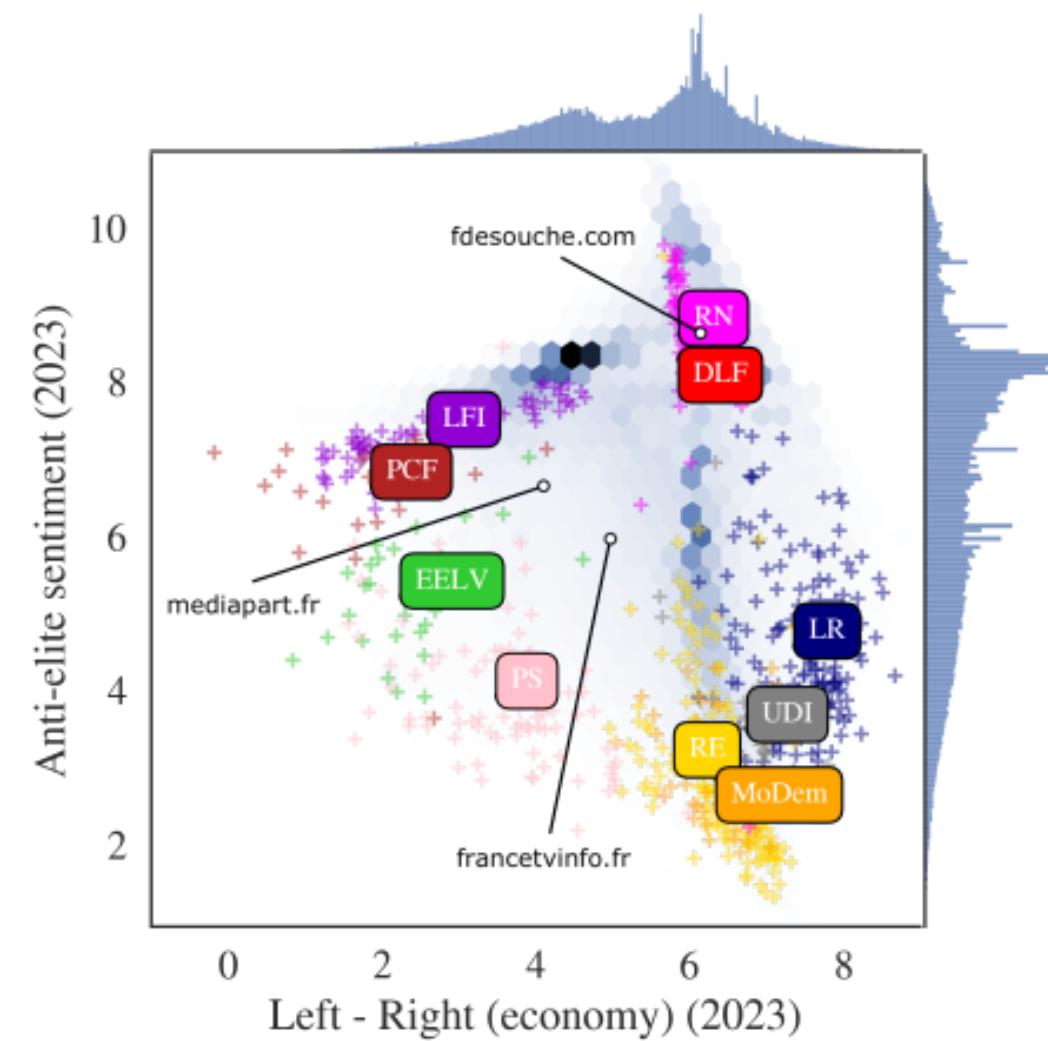
Data is available at
<https://doi.org/10.21410/7E4/QPECFF>



llm-based annotations for validation

Social Network's embeddings

European Polarisation Observatory



Mapping the Digital Political Landscape...

Vendeville et all

<https://hal.science/hal-05222448>

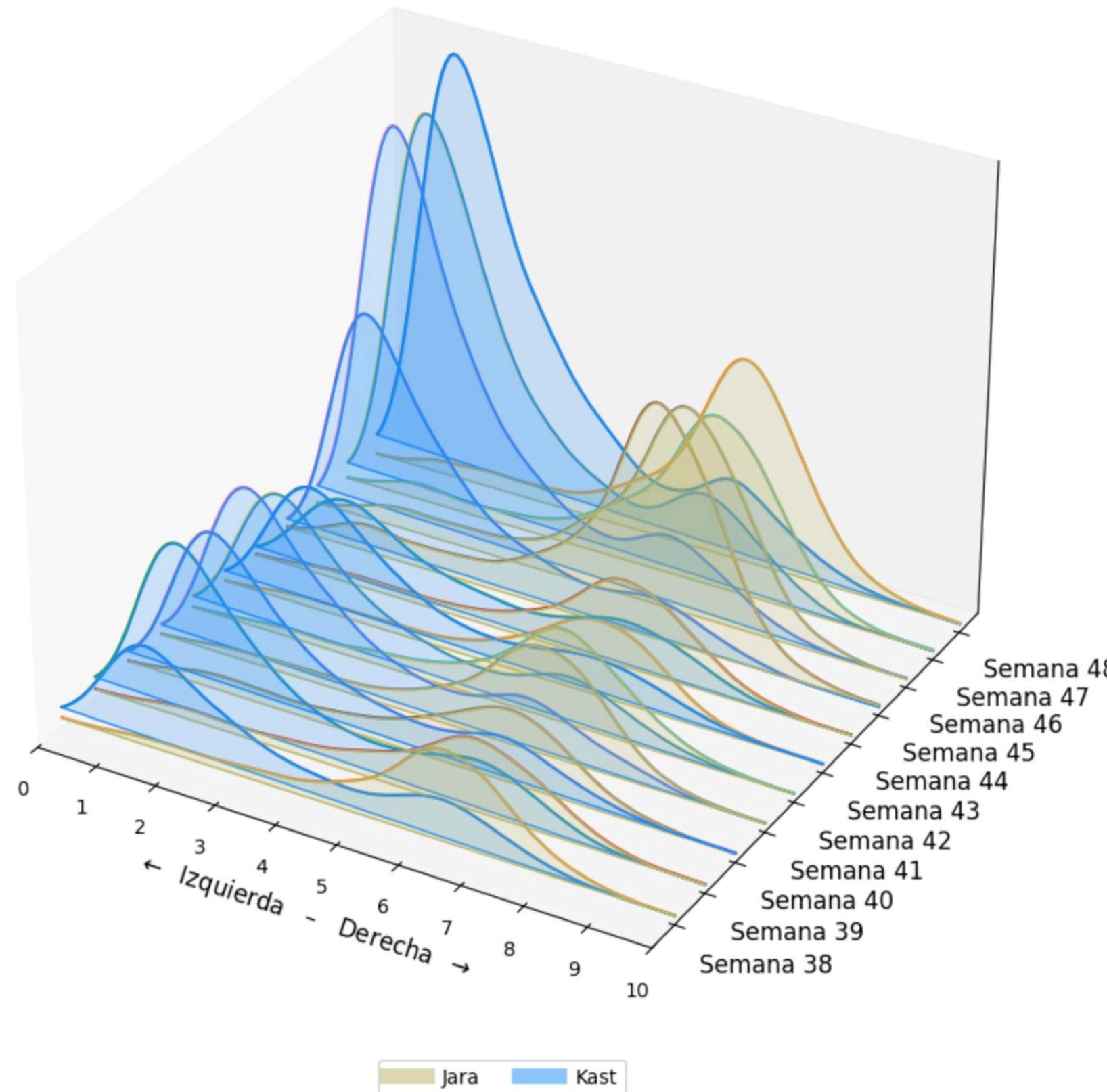
Data is available at
<https://doi.org/10.21410/7E4/QPECFF>



Ilm-based annotations for validation

Social Network's embeddings

European Polarisation Observatory



Chilean December 2025 presidential elections



Pedro Ramaciotti ✅ • 1st

Chair of AI in Computational Social Sciences French National Research Cent...

1d • 🔍

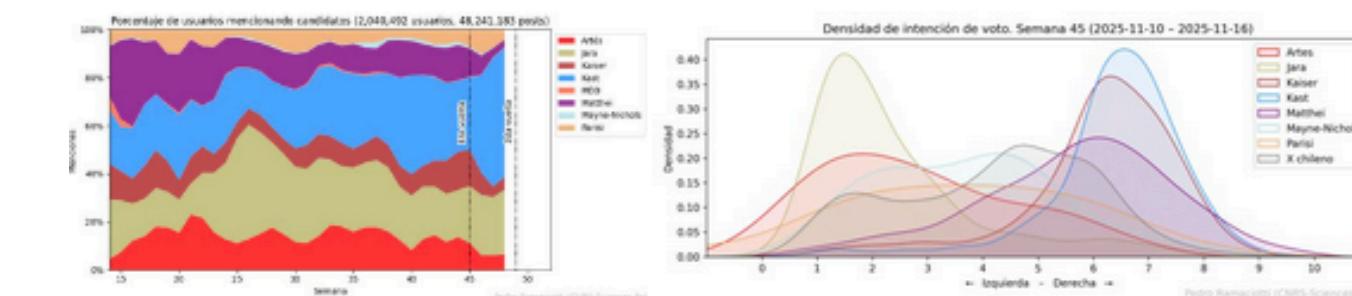
...

En el Observatorio Europeo de la Polarización en París seguimos la elección chilena. Analizamos un panel de 2 millones de usuarios X/Twitter (aprox. 60% del total en Chile, y 50M posts) durante 30 semanas usando inferencia ideológica e IA (LLMs) para anotar sus preferencias.

Usamos ideology scaling para posicionar usuarios en escala de izq. a derecha, y usamos LLMs para anotar por quien dicen querer votar, a quien apoyan, y a quien critican. Terminada la veda sobre datos de preferencias electorales publicaremos análisis detallados.

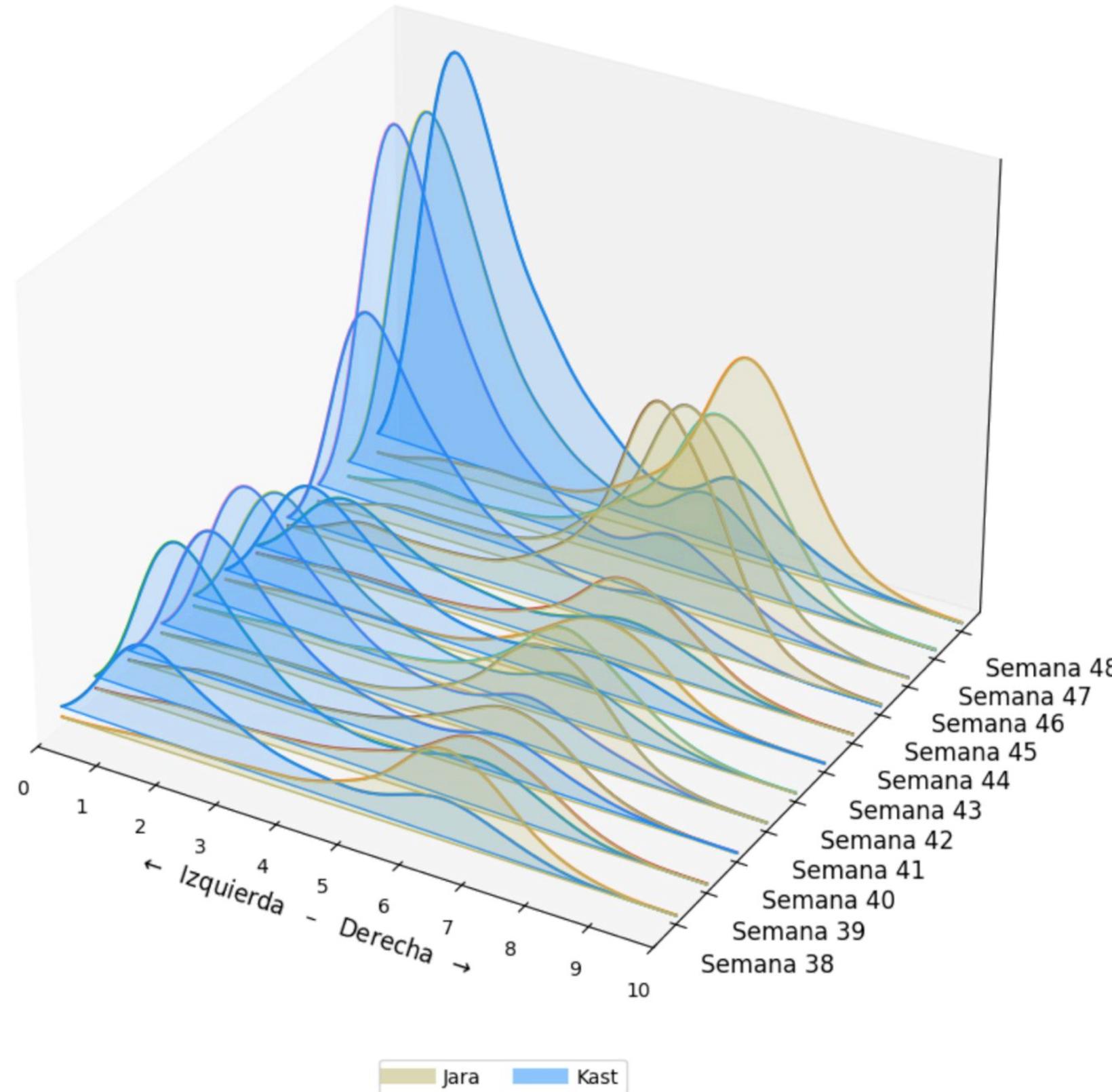
Buenas elecciones y hasta la próxima semana.

Show translation



Social Network's embeddings

European Polarisation Observatory



Chilean December 2025 presidential elections

- At the European Polarization Observatory in Paris, we followed the Chilean election. We analyzed a panel of **2 million X/Twitter users** (approximately 60% of the total in Chile, and **50 million posts**) over **30 weeks** using ***ideological inference and AI (LLMs*** to record their preferences.
- We used ***ideology scaling*** to position users on a left-to-right scale, and LLMs to record who they said they wanted to vote for, who they supported, and who they criticized.
- Once the blackout on electoral preference data is lifted, we will publish detailed analyses.

Using LLMs to study political-related content

Codebook instructions	Prompt instructions
Based on both the title, publication date, and the obituary text, infer the person's age at death in years and insert it as a numeric value. If age is not inferable, leave this field blank. If the person's age is not inferable, insert 9999.	<p>Below I will provide an obituary of a deceased person.</p> <p>Based on both the title, publication date, and the obituary text, infer the person's age at death in years. You should respond with a numeric value. If the person's age is not inferable, respond with "9999".</p> <p>Please format your response in plain text, inside quotation marks, like this: "<insert your response>"</p> <p>Here is the obituary date, title, and text: {insert obituary text}</p>
Does the obituary mention that the person served in the army? If so, put "yes." Otherwise, put "not mentioned".	<p>Below I will provide an obituary of a deceased person.</p> <p>Does the obituary mention that the person served in the military? If so, respond with "yes". Otherwise, respond with "not mentioned".</p> <p>If a person served in a foreign (that is non-US) military, also respond with "yes". Please limit your response to only one of these two codes: "yes", "not mentioned".</p> <p>Advisory roles do not count as having served in the army.</p> <p>Please format your response in plain text, inside quotation marks, like this: "<insert your response>"</p> <p>Here is the obituary date, title, and text: {insert obituary text}</p>
Table 2. Codebook and prompt instruction for age in years and military service. <i>Note: Changes are marked up in gray.</i>	

From Codebooks to Promptbooks: Extracting Information from Text with Generative Large Language Models

Stuhler et all

https://osf.io/preprints/socarxiv/wjvfg_v1

Mapping and Influencing the Political Ideology of Large Language Models using Synthetic Personas

Pietro Bernardelle
The University of Queensland
Brisbane, Australia
p.bernardelle@uq.edu.au

Leon Fröhling
GESIS
Cologne, Germany
leon.froehling@gesis.org

Stefano Civelli
The University of Queensland
Brisbane, Australia
s.civelli@uq.edu.au

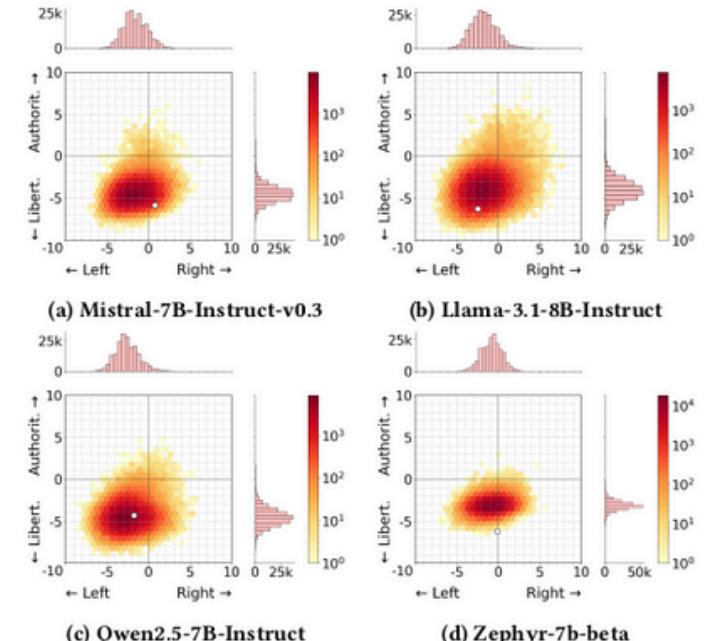
Riccardo Lunardi
University of Udine
Udine, Italy
riccardo.lunardi@uniud.it

Kevin Roitero
University of Udine
Udine, Italy
kevin.roitero@uniud.it

Gianluca Demartini
The University of Queensland
Brisbane, Australia
demartini@acm.org

Abstract

The analysis of political biases in large language models (LLMs) has primarily examined these systems as single entities with fixed viewpoints. While various methods exist for measuring such biases, the impact of persona-based prompting on LLMs' political orientation remains unexplored. In this work we leverage PersonaHub, a collection of synthetic persona descriptions, to map the political distribution of persona-based prompted LLMs using the Political Compass Test (PCT). We then examine whether these initial compass distributions can be manipulated through explicit ideological prompting towards diametrically opposed political orientations: right-authoritarian and left-libertarian. Our experiments reveal that synthetic personas predominantly cluster in the left-libertarian quadrant, with models demonstrating varying degrees of responsiveness when prompted with explicit ideological descriptors. While all models demonstrate significant shifts towards right-authoritarian positions, they exhibit more limited shifts towards left-libertarian positions, suggesting an asymmetric response to ideological manipulation that may reflect inherent biases in model training.



Mapping and Influencing the Political Ideology of Large Language Models using Synthetic Personas
Bernardelle et all

https://osf.io/preprints/socarxiv/wjvfg_v1

Social media text classification cf. Emilien's talk

A few to a hundred words texts (profile bios, posts)



BFMTV
@BFMTV
Municipales de 2026: "Il n'y aura pas de maires de La France insoumise",
assure Raphaël Glucksmann ("Place publique")

...

Cerveaux Non Disponibles

@cerveaux_non_disponibles

Media/news company
Vivez les luttes sociales, antiracistes et climatiques via des contenus indépendants et alternatifs.
cerveauxnondisponibles.net + 3

Elsa Marcel

@Elsa_Marcel

Avocate au Barreau de Seine-Saint-Denis.
Du côté de la jeunesse, des travailleurs, des classes populaires et de tous les opprimés.
Militante à [@RevPermanente](#).

A rejoint X en décembre 2018

Multi-class multi-label classification setting

- sentiment analysis
- thematic labelling
- ...

- sport | politics | entertainment | family | etc.
- yes | non
- left | right | unknown
- support | disagreement | neutral

Cerveaux Non Disponibles

@cerveaux_non_disponibles

Media/news company
Vivez les luttes sociales, antiracistes et climatiques via des contenus indépendants et alternatifs.
cerveauxnondisponibles.net + 3

left-leaning account



?

favors ecologic



?

Difficulties

Social media database can easily be huge

country	STATS			
	# lut users	# pp graph mps	# pp graph followers	# pp metadata users
us	36 004 856	522	10 926 871	5 581 233
turkey	22 555 696	586	8 899 430	2 703 836
brazil	17 703 352	528	6 142 272	2 834 581
mexico	9 506 524	534	2 119 744	940 393
uk	8 931 021	450	2 007 465	1 271 484

Social media content inner complexities

- political content
- insults, offenses, mockery
- sarcasm / irony

text @invermel Miente miente que algo queda !!!!
Pero ya nadie le cree al nazi @joseantoniookast .
Te queda claro imbécil acéfalo @invermel
english @invermel You lie you lie that something is left!!!
But no one believes the Nazi @joseantoniookast anymore.
It's clear to you imbecile without a head @invermel

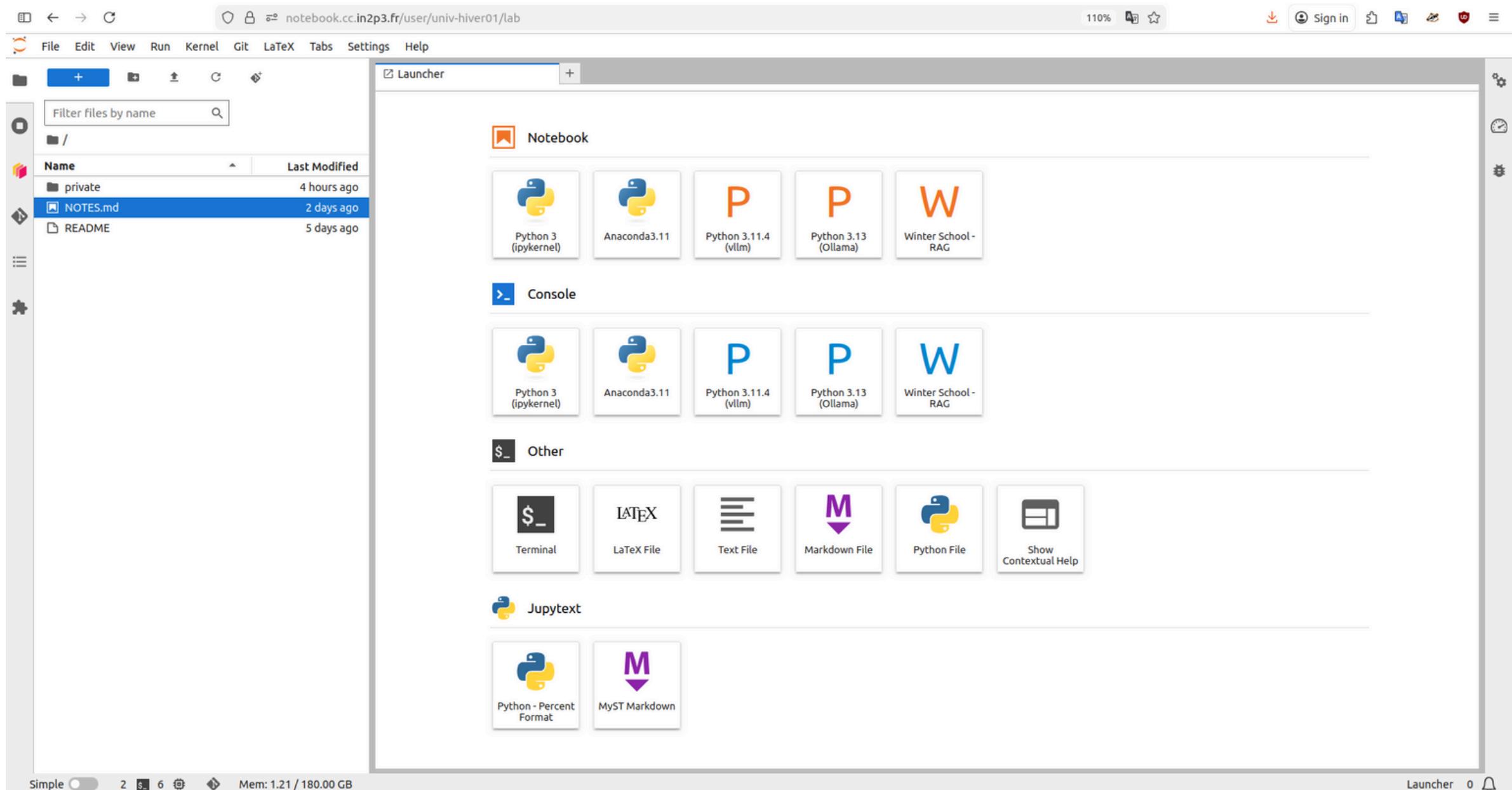
Limited hardware ressources

One of two GPU card of
20 up to 40 GB of RAM

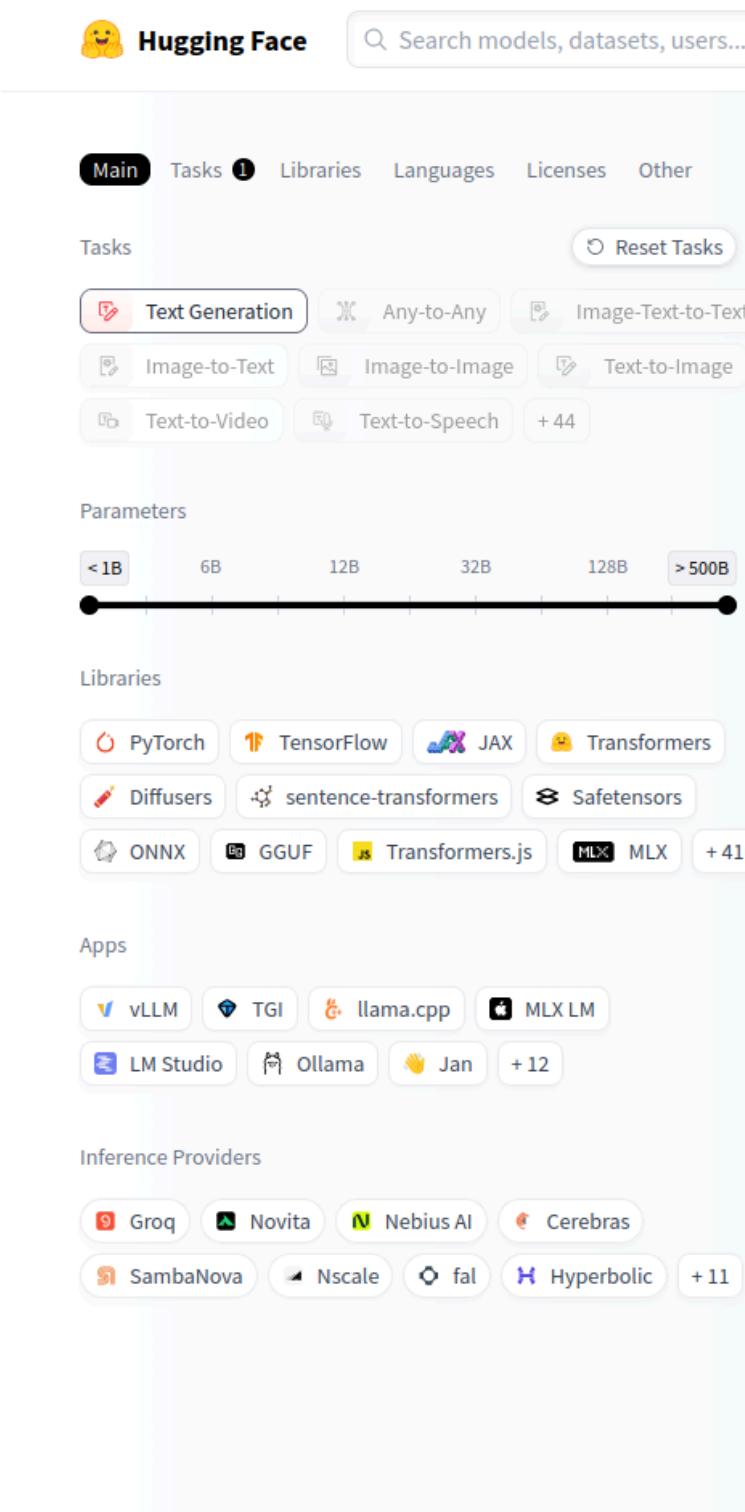
Hands on

<https://notebook.cc.in2p3.fr>

<https://github.com/jimenaRL/resinwinterschool>



Hugging face Transformer LLM-based annotations



The screenshot shows the Hugging Face homepage. On the left, there's a sidebar with sections for Tasks (Text Generation, Any-to-Any, Image-Text-to-Text, Image-to-Text, Image-to-Image, Text-to-Image, Text-to-Video, Text-to-Speech), Parameters (a slider from <1B to >500B), Libraries (PyTorch, TensorFlow, JAX, Transformers, Diffusers, sentence-transformers, Safetensors, ONNX, GGU, Transformers.js, MLX, +41), Apps (vLLM, TGI, llama.cpp, MLX LM, LM Studio, Ollama, Jan, +12), and Inference Providers (Groq, Novita, Nebius AI, Cerebras, SambaNova, Nscale, fal, Hyperbolic, +11). The main area is titled "Models 305,100" and features a search bar and filters for "Full-text search", "Inference Available", and "Sort: Most likes". It lists various models like deepseek-ai/DeepSeek-R1, meta-llama/Meta-Llama-3-8B, meta-llama/Llama-3.1-8B-Instruct, bigscience/bloom, meta-llama/Llama-2-7b-chat-hf, meta-llama/Llama-2-7b, openai/gpt-oss-120b, openai/gpt-oss-20b, mistralai/Mistral-7B-v0.1, microsoft/phi-2, google/gemma-7b, deepseek-ai/DeepSeek-V3, deepseek-ai/DeepSeek-V3-0324, openai-community/gpt2, mistralai/Mistral-7B-Instruct-v0.2, bigcode/starcoder, Qwen/QwQ-32B, meta-llama/Llama-3.3-70B-Instruct, and tiiuae/falcon-40b.

Looking for a model

<https://huggingface.co/models>



Hugging face Transformer LLM-based annotations

The screenshot shows the Hugging Face Model Card for the **Meta-Llama-3-8B** model. Key details include:

- Model card:** Shows a note about a newer version ([meta-llama/Llama-3.1-8B](#)) and a gated access message.
- Downloads last month:** 2,081,998
- Safetensors:** Model size: 8B params
- Inference Providers:** Text Generation (NEW)
- Libraries:** Transformers
- Other Providers:** Inference Providers, Notebooks, Google Colab, Kaggle, Local Apps, vLLM, TGI

Model Details:

Meta developed and released the Meta Llama 3 family of large language models (LLMs), a collection of pretrained and instruction tuned generative text models in 8 and 70B sizes. The Llama 3 instruction tuned models are optimized for dialogue use cases and outperform many of the available open source chat models on common industry benchmarks. Further, in developing these models, we took great care to optimize helpfulness and safety.

Model developers: Meta

Usage instructions

<https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct?library=transformers>

Hugging face Transformer library

The screenshot shows the Hugging Face Model Card for the **Meta-Llama-3-8B** model. A modal window titled "How to use from the **Transformers** library" is displayed, containing four code snippets:

- # Gated model: Login with a HF token with gated access permission
hf auth login
- # Use a pipeline as a high-level helper
from transformers import pipeline
- pipe = pipeline("text-generation", model="meta-llama/Meta-Llama-3-8B")
- # Load model directly
from transformers import AutoTokenizer, AutoModelForCausalLM
- tokenizer = AutoTokenizer.from_pretrained("meta-llama/Meta-Llama-3-8B")
model = AutoModelForCausalLM.from_pretrained("meta-llama/Meta-Llama-3-8B")

The modal also includes "Copy" buttons for each snippet and a close button (X). The background of the card shows a dark theme with various sections like "Model card", "Model Details", "Variations", and "Input".

Usage instructions

<https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct?library=transformers>

Prompt engineering

The screenshot shows a user interface for an AI inference provider. At the top, there are tabs for "Inference Providers" (with a "NEW" badge), "Novita", and "+3". Below that are buttons for "Text Generation", "Reset Chat", and "Examples". A text input field says "Input a message to start chatting with meta-llama/Llama-3.1-8B-Instruct.". A blue message box contains the following text:

Please classify the following text as "Left-leaning", "Not-Left" or "Unknown" according to whether it contains politically Left-leaning or not: "Avec mes amis indépendantistes Bretons de Dispac'h ! Gitans et Bretons, même combat contre l'État français colonial et impérialiste 🤝"

A response message box below says "I would classify this text as "Left-leaning".". The input field says "Here's why:" followed by "Your sentence here...". A "Send" button is next to the input field. At the bottom left is a link "View Code Snippets" and at the bottom right is a link "Open Playground".

"Be concise" example

I would classify this text as "Left-leaning".

Here's why: The text mentions "indépendantistes Bretons", which suggests support for the Breton independence movement, a cause often associated with the Left. It mentions fighting against a "colonial and imperialist" state, which is a common criticism of Western colonial powers from a Left-leaning perspective. The use of the fist emoji (🤝) also suggests a more radical or activist tone, which is often associated with Left-leaning politics. However, it's worth noting that the text also contains a reference to "Gitans" (a term for Romani people), which could be seen as an expression of solidarity with a marginalized group, which is a common theme in Left-leaning politics. So, based on these factors, I would classify the text as Left-leaning, but I would also acknowledge that it's difficult to determine the exact political orientation of the author without more context.

The screenshot shows a similar AI inference provider interface. A blue message box contains the following text:

Please classify the following text as "Left-leaning", "Not-Left" or "Unknown" according to whether it contains politically Left-leaning or not. Be concise and answer only "Left-leaning", "Not-Left" or "Unknown": "Avec mes amis indépendantistes Bretons de Dispac'h ! Gitans et Bretons, même combat contre l'État français colonial et impérialiste 🤝"

At the bottom left, a green button says "Left-leaning".

Left-leaning

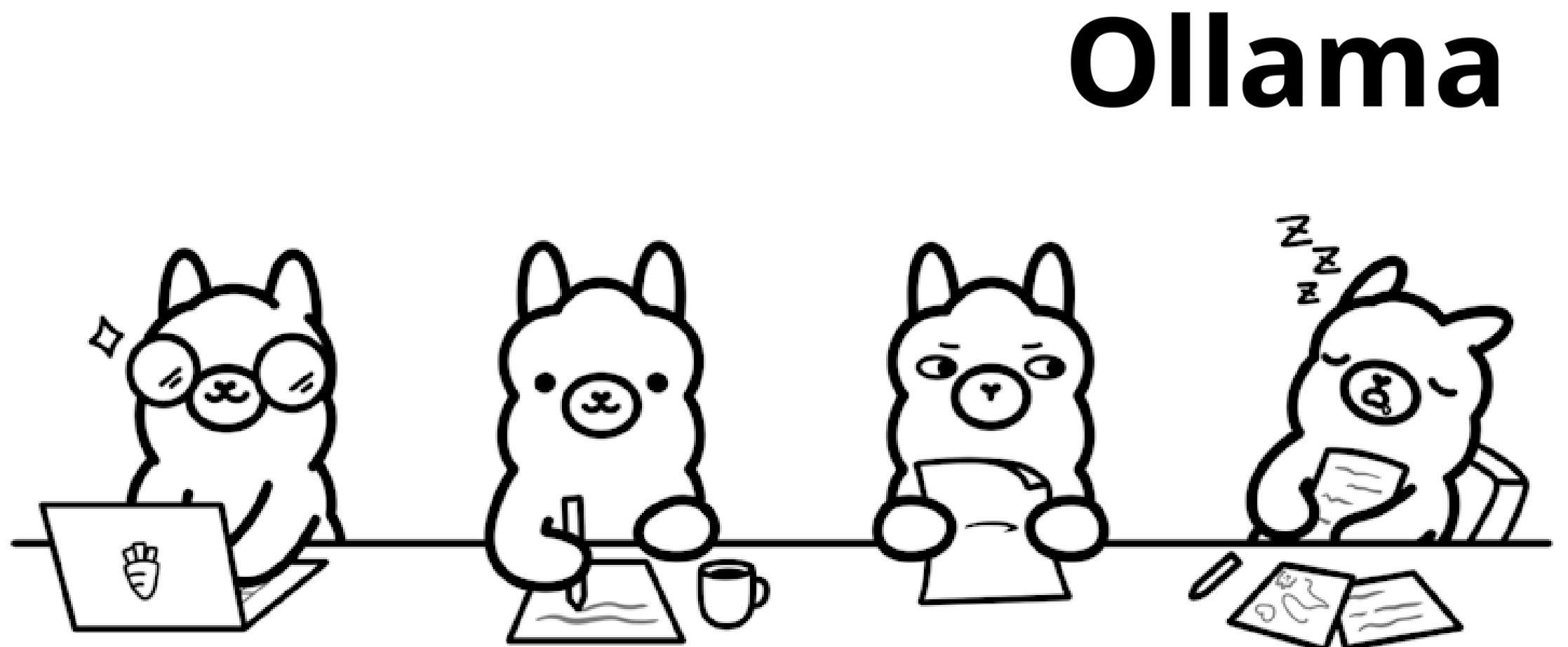
Hugging face Transformer LLM-based annotations



Jupyter notebook

[https://github.com/jimenaRL/ILmBenchmarks/blob/main/venice/
venice_some4dem_workshop_transformers.ipynb](https://github.com/jimenaRL/ILmBenchmarks/blob/main/venice/venice_some4dem_workshop_transformers.ipynb)

Ollama Server LLM-based annotations

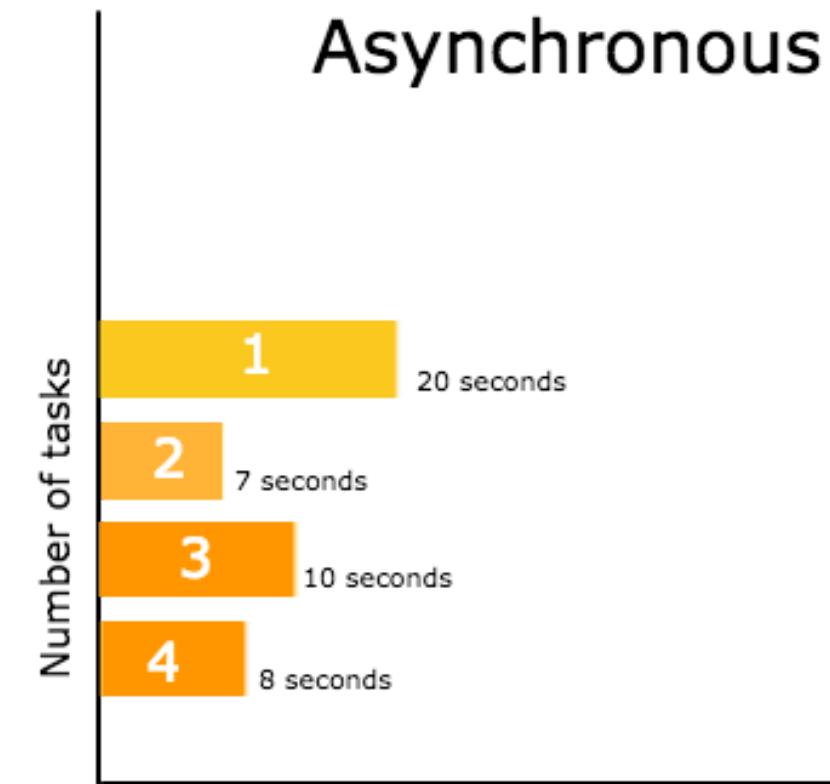
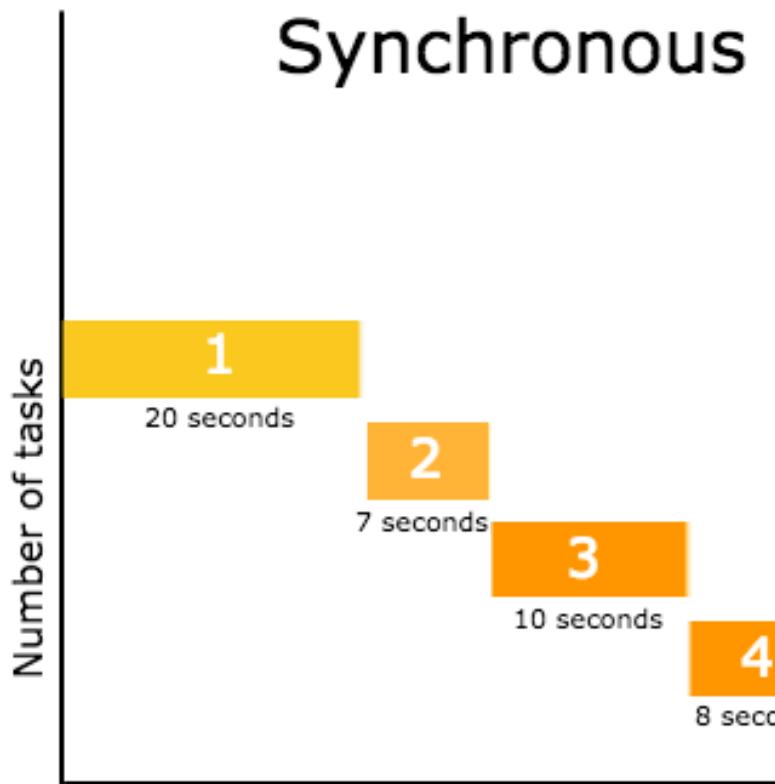


Ollama

Jupyter notebook

https://github.com/jimenaRL/llmBenchmarks/blob/main/venice/venice_some4dem_workshop_transformers.ipynb

online serving - OpenAI API compatibility - async requests



```
import asyncio
from openai import OpenAI

# 1/ Wait for vllm server to be available and retrieve model
openai_api_key = "EMPTY"
openai_api_base = "http://localhost:8000/v1"
client = OpenAI(
    base_url=openai_api_base,
    api_key=openai_api_key)

model = client.models.list().data[0].id

# 2/ Create async functions to request vllm server through openAI API
async def doCompletion(model, messages, extra_body, tweet):
    completion = client.chat.completions.create(
        model=model,
        messages=messages,
        extra_body=extra_body)
    return tweet, completion.choices[0].message.content

# 3/ Create async message iterator
async def messageIterator():
    for tweet in tweets:
        yield tweet, [
            {
                "role": "system",
                "content": system_prompt
            },
            {
                "role": "user",
                "content": Template(user_prompt).substitute(tweet=tweet)
            }
        ]

# 4/ Create coroutines
async def run_all():
    # Asynchronously call the function for each prompt
    tasks = [
        doCompletion(model, it[1], extra_body, it[0])
        async for it in messageIterator()
    ]
    # Gather and run the tasks concurrently
    results = await asyncio.gather(*tasks)
    return results
```

python async requests

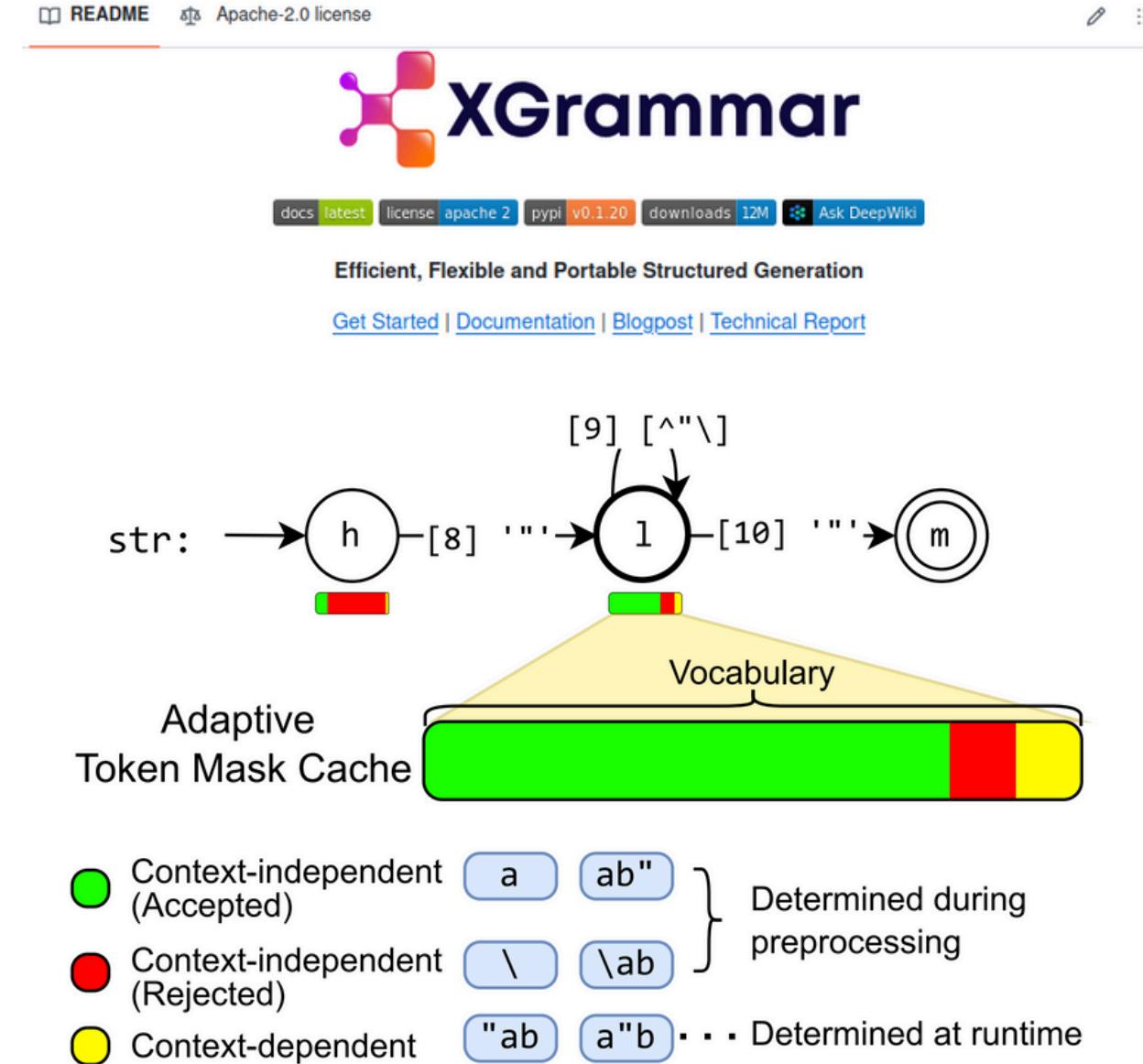
Hugging face Transformer LLM-based annotations



Jupyter notebook

[https://github.com/jimenaRL/IlmBenchmarks/blob/main/venice/
venice_some4dem_workshop_transformers.ipynb](https://github.com/jimenaRL/IlmBenchmarks/blob/main/venice/venice_some4dem_workshop_transformers.ipynb)

Structures outputs



XGrammar is an open-source library for efficient, flexible, and portable **structured generation**.

It leverages constrained decoding to ensure 100% structural correctness of the output.

It supports general context-free grammar to enable a broad range of structures, including JSON, regex, custom context-free grammar, etc.

At generation time, a language model produces probabilities for possible next tokens. Structured output constrains this by masking invalid tokens, ensuring only tokens that comply with the defined constraints remain candidates for sampling.

Structures outputs

```
completion = client.chat.completions.create(  
    model=model,  
    messages=[  
        {  
            "role": "user",  
            "content": "Generate an example email address for Alan Turing, who works in Enigma. End  
        }  
    ],  
    extra_body={"guided_regex": r"\w+@\w+\.\com\n", "stop": ["\n"]},  
)  
print(completion.choices[0].message.content)
```

regex pattern

```
from openai import OpenAI  
client = OpenAI(  
    base_url="http://localhost:8000/v1",  
    api_key="-",  
)  
model = client.models.list().data[0].id  
  
completion = client.chat.completions.create(  
    model=model,  
    messages=[  
        {"role": "user", "content": "Classify this sentiment: vLLM is wonderful!"}  
    ],  
    extra_body={"guided_choice": ["positive", "negative"]},  
)  
print(completion.choices[0].message.content)
```

discrete choices

```
simplified_sql_grammar = """  
root ::= select_statement  
  
select_statement ::= "SELECT " column " from " table " where " condition  
  
column ::= "col_1" | "col_2"  
  
table ::= "table_1" | "table_2"  
  
condition ::= column "= " number  
  
number ::= "1" | "2"  
"  
  
completion = client.chat.completions.create(  
    model=model,  
    messages=[  
        {  
            "role": "user",  
            "content": "Generate an SQL query to show the 'username' and 'email' from the 'users' ta  
        }  
    ],  
    extra_body={"guided_grammar": simplified_sql_grammar},  
)  
print(completion.choices[0].message.content)
```

simplified SQL queries

```
from pydantic import BaseModel  
from enum import Enum  
  
class CarType(str, Enum):  
    sedan = "sedan"  
    suv = "SUV"  
    truck = "Truck"  
    coupe = "Coupe"  
  
class CarDescription(BaseModel):  
    brand: str  
    model: str  
    car_type: CarType  
  
    json_schema = CarDescription.model_json_schema()  
  
completion = client.chat.completions.create(  
    model=model,  
    messages=[  
        {  
            "role": "user",  
            "content": "Generate a JSON with the brand, model and car_type of the most iconic car f  
        }  
    ],  
    "response_format": {  
        "type": "json_schema",  
        "json_schema": {  
            "name": "car-description",  
            "schema": CarDescription.model_json_schema()  
        },  
    },  
)  
print(completion.choices[0].message.content)
```

json schema

https://docs.vllm.ai/en/latest/features/structured_outputs.html

Can we do better with the same model ?

> Inference is different from training

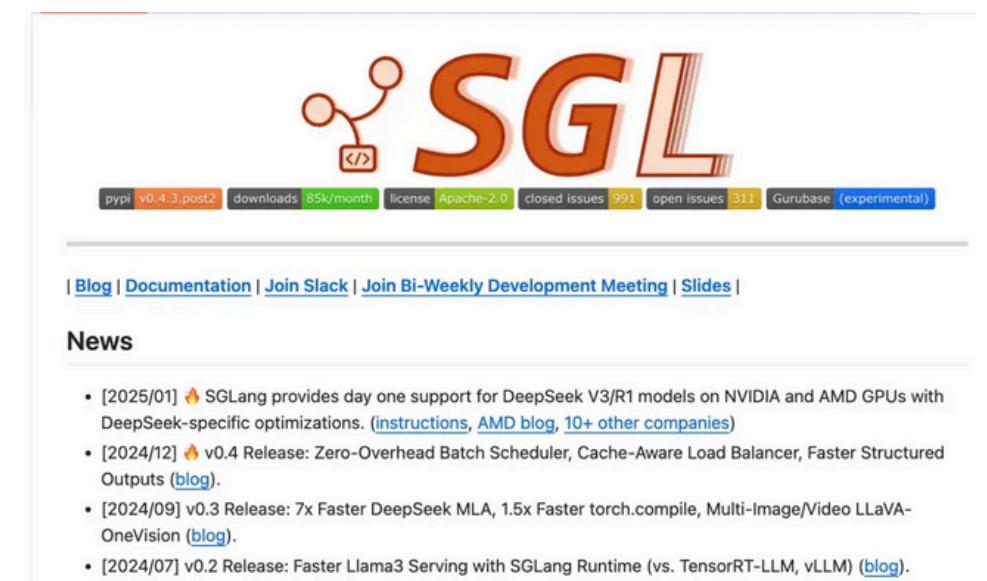
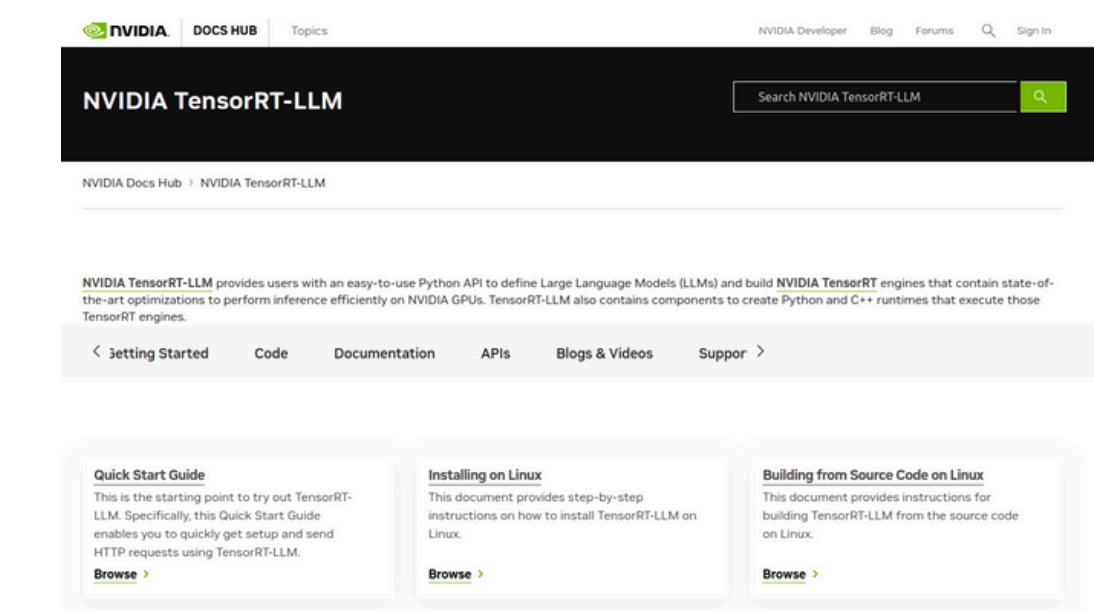
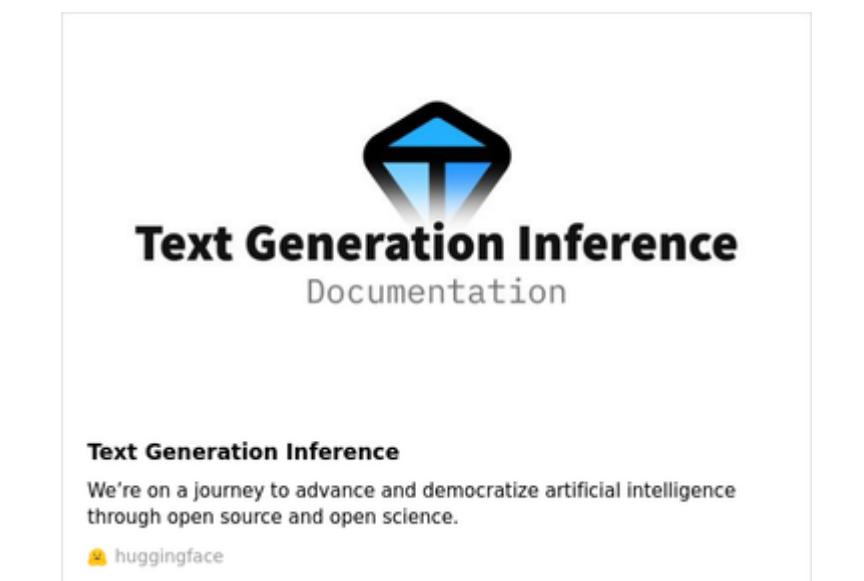
INFERENCE SERVING FRAMEWORKS (~2023)

<https://huggingface.co/docs/text-generation-inference>

<https://github.com/NVIDIA/TensorRT-LLM>

<https://github.com/sjl-project/sclang>

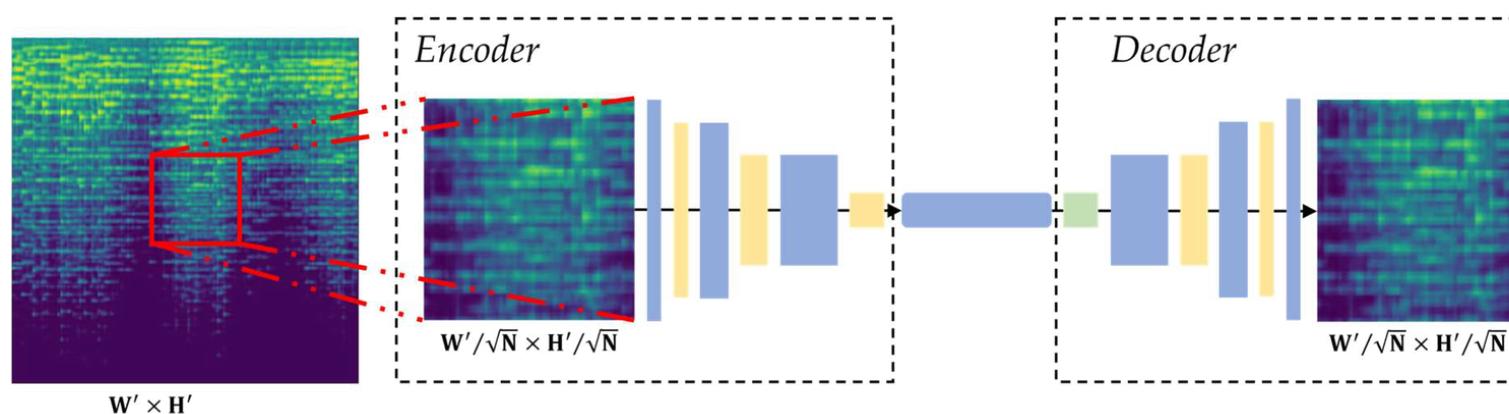
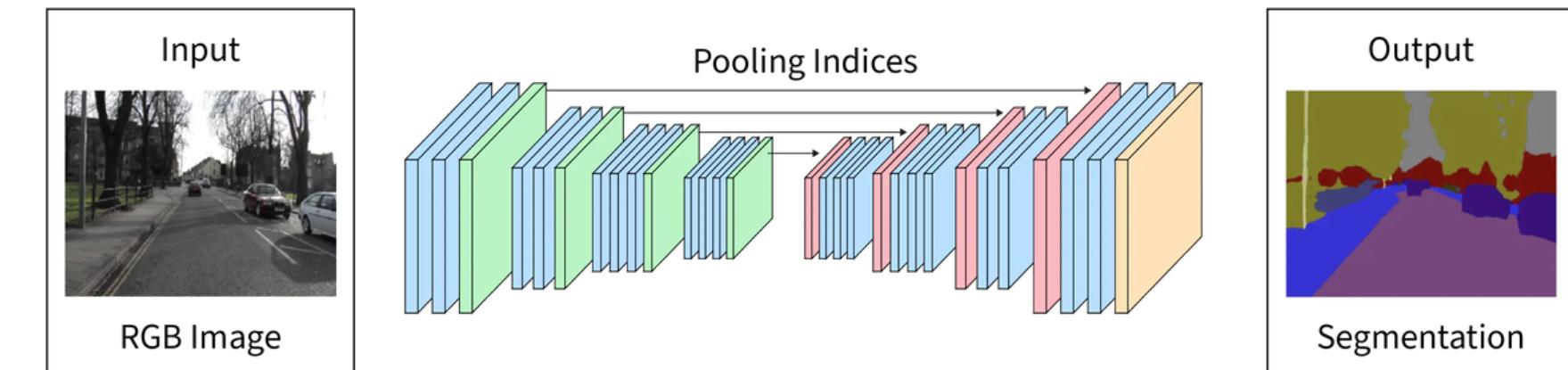
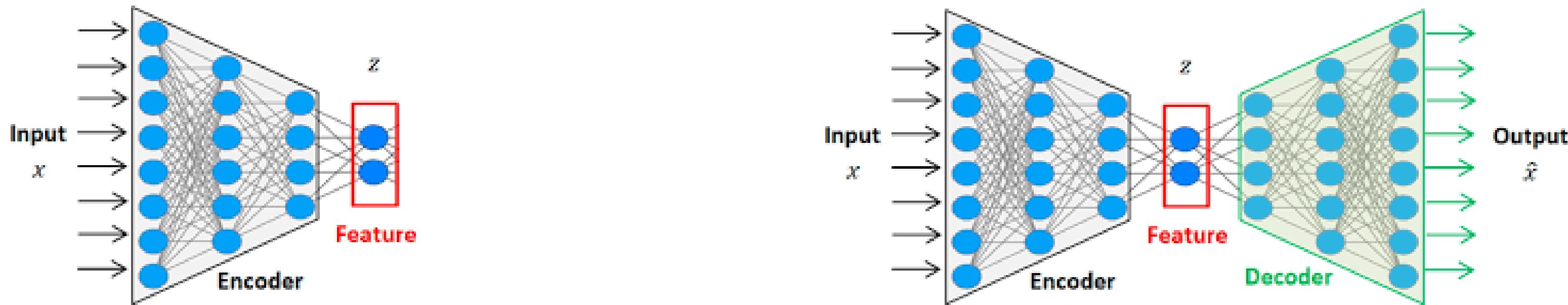
<https://docs.vllm.ai/en/latest/>



1. **Traditional machine/deep learning frameworks** (like Hugging Face's python library Transformers) are the standard for training LLM architectures, but they **are not optimized for inference after training**.
2. After plenty of developments in accelerating hardware (GPUs) and enhancing compute capacities, **performance of LLM serving is bottle-necked by memory**, since roughly speaking, GPU card operations are faster than loading data into memory, leading to inefficient GPU utilization.
4. Modern frameworks overcome these limitations through inference optimizations that **both increase throughput** (number of requests treated) **and reduce latency** (speed of treatment per request).
5. Modern frameworks are also **high-level packaged** and they deployed as servers that implement the **OpenAI API for completions and chat**.

HOW LLMs/TRANSFORMERS WORK

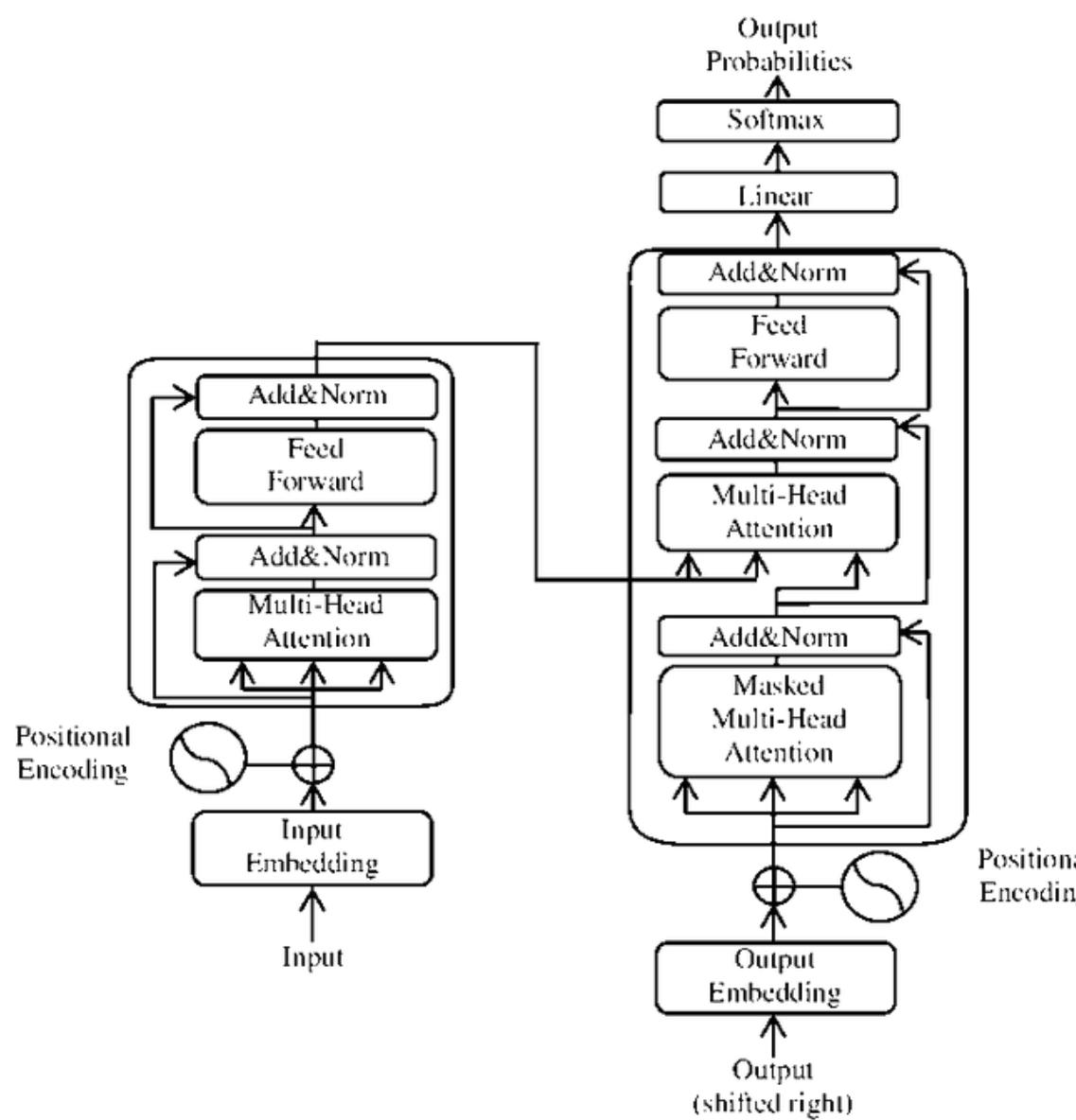
encoder decoder architecture



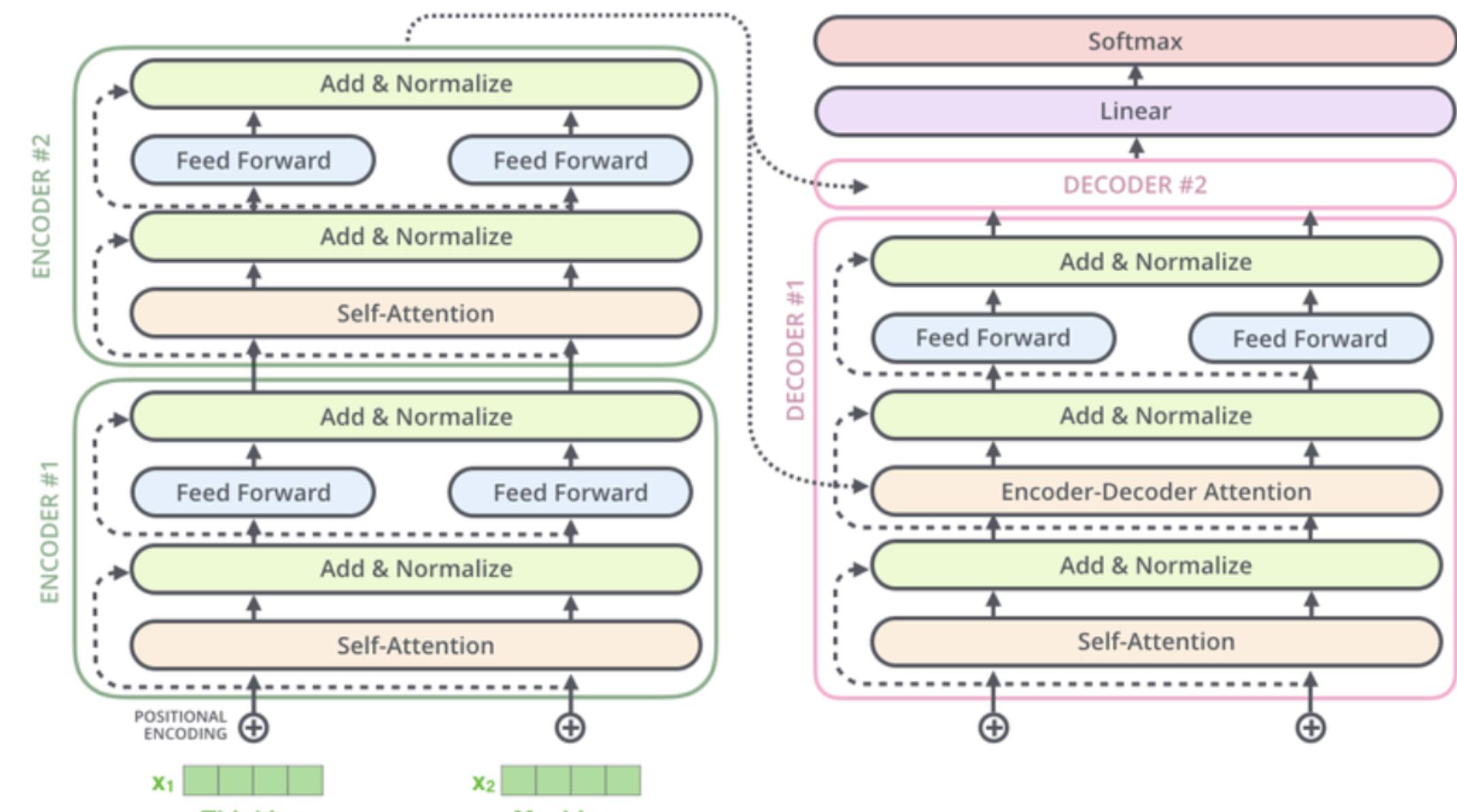
HOW LLMs/TRANSFORMERS WORK

encoder decoder architecture with self-attention mechanisms

context length
token sampling
self-attention



KV cache
batching

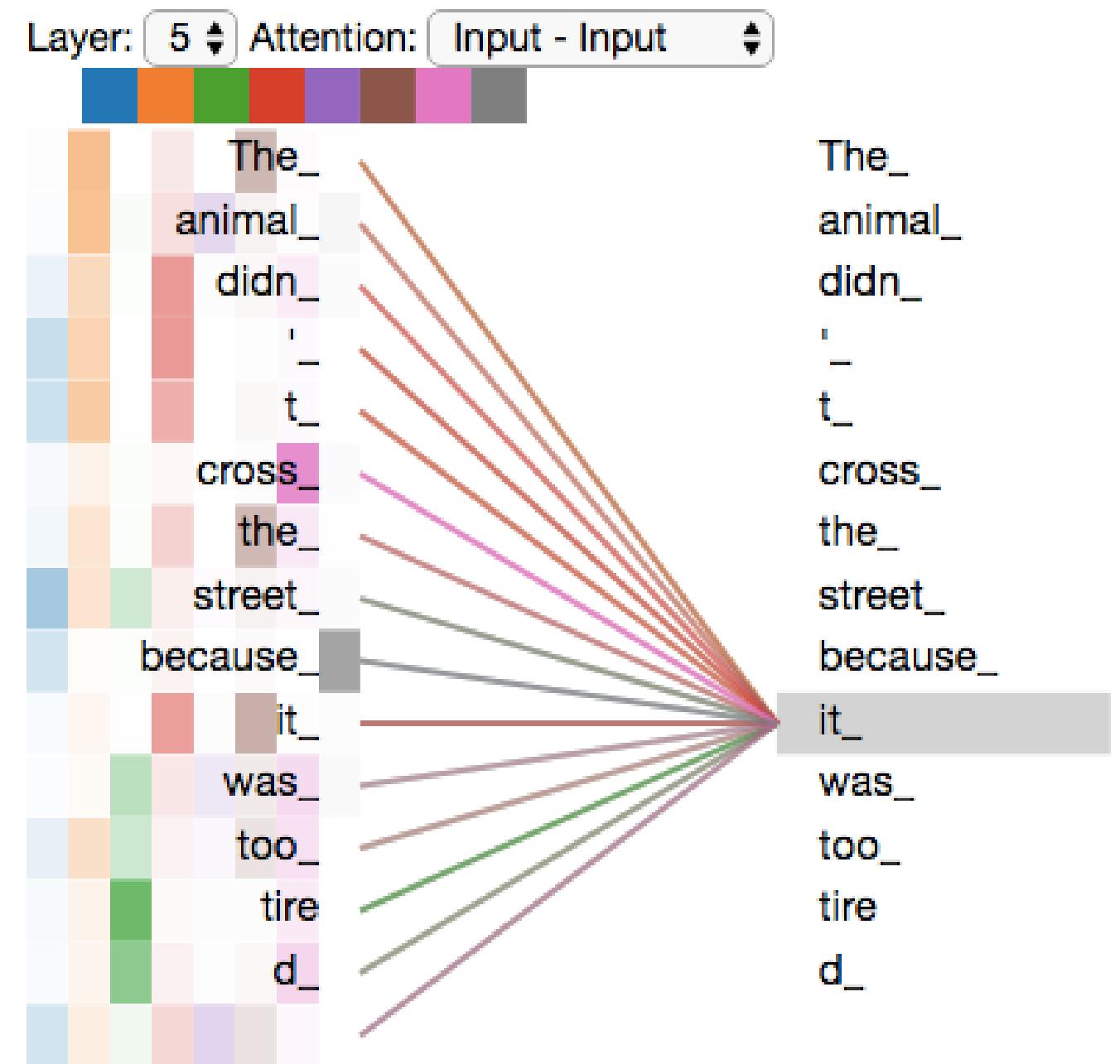
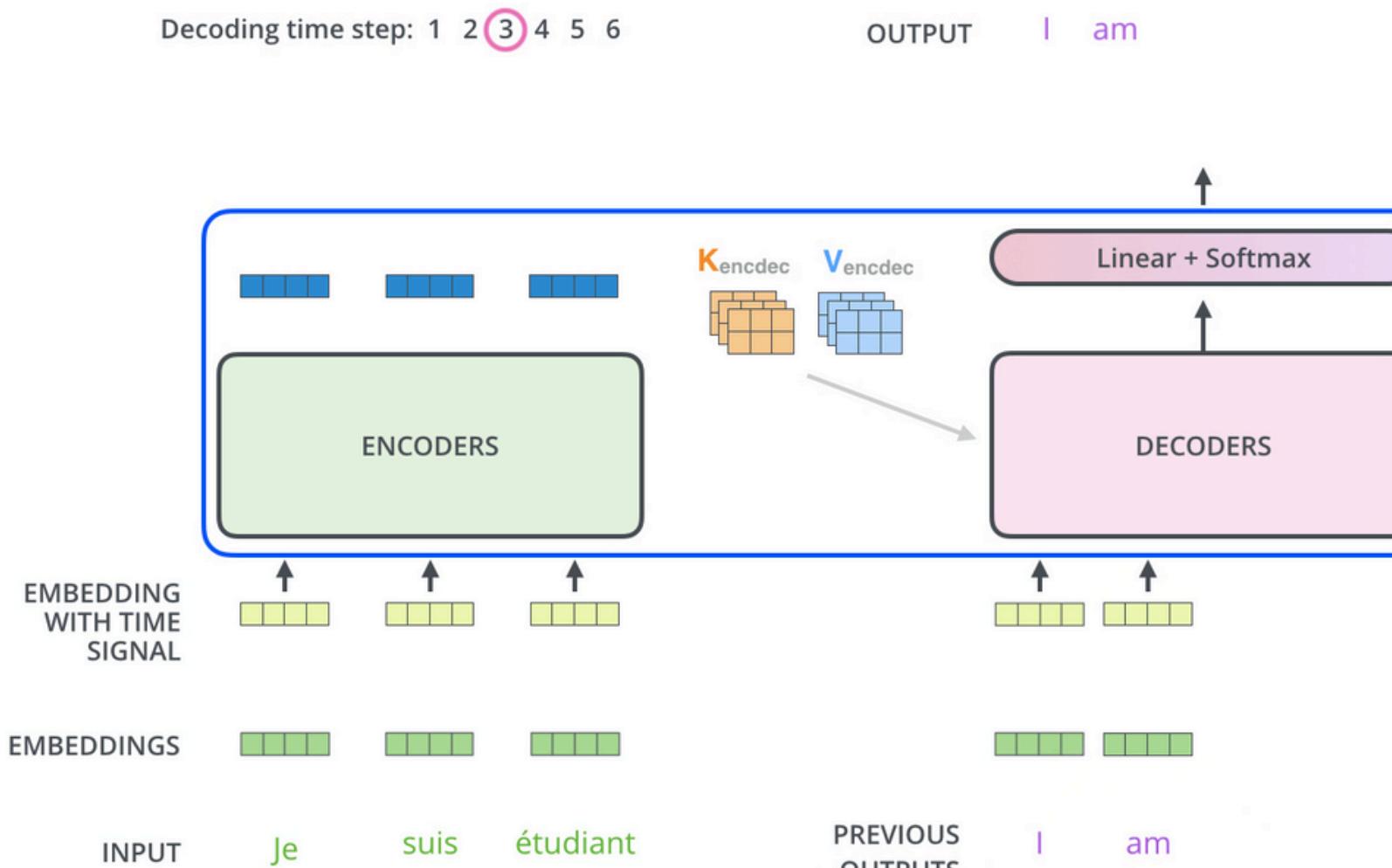


HOW TRANSFORMERS WORK

self-attention

KV cache

batching



HOW TRANSFORMERS WORK

self-attention

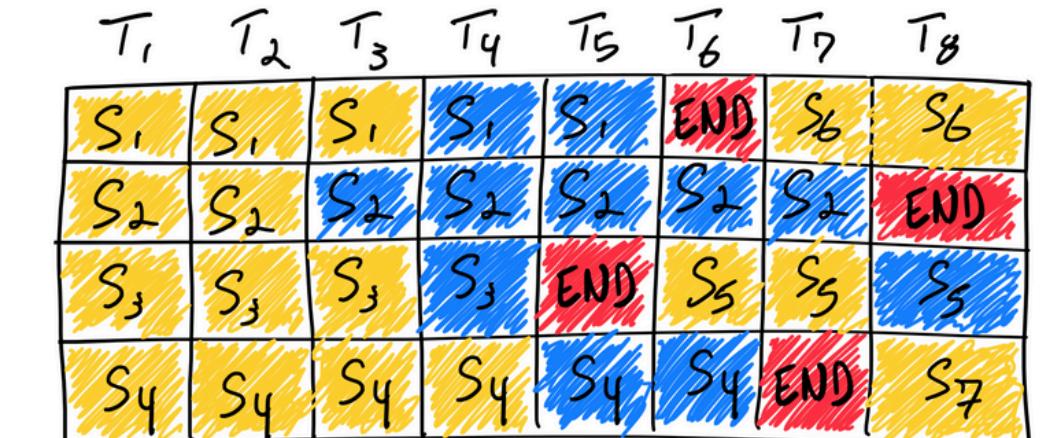
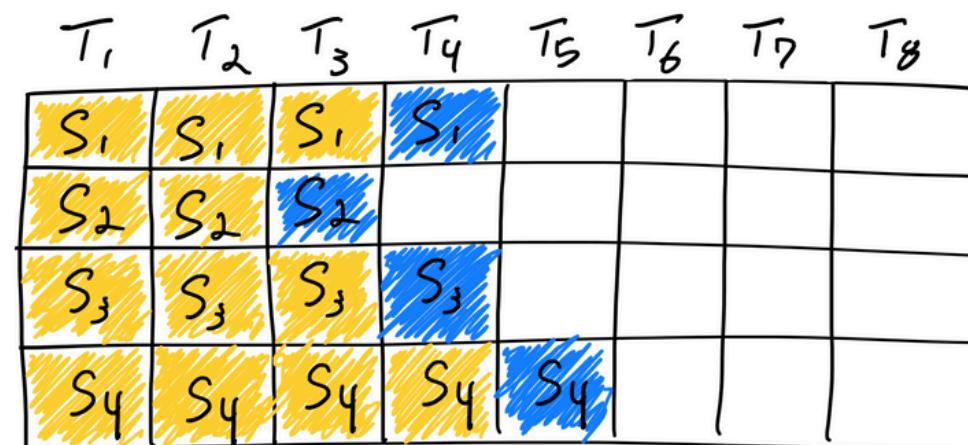
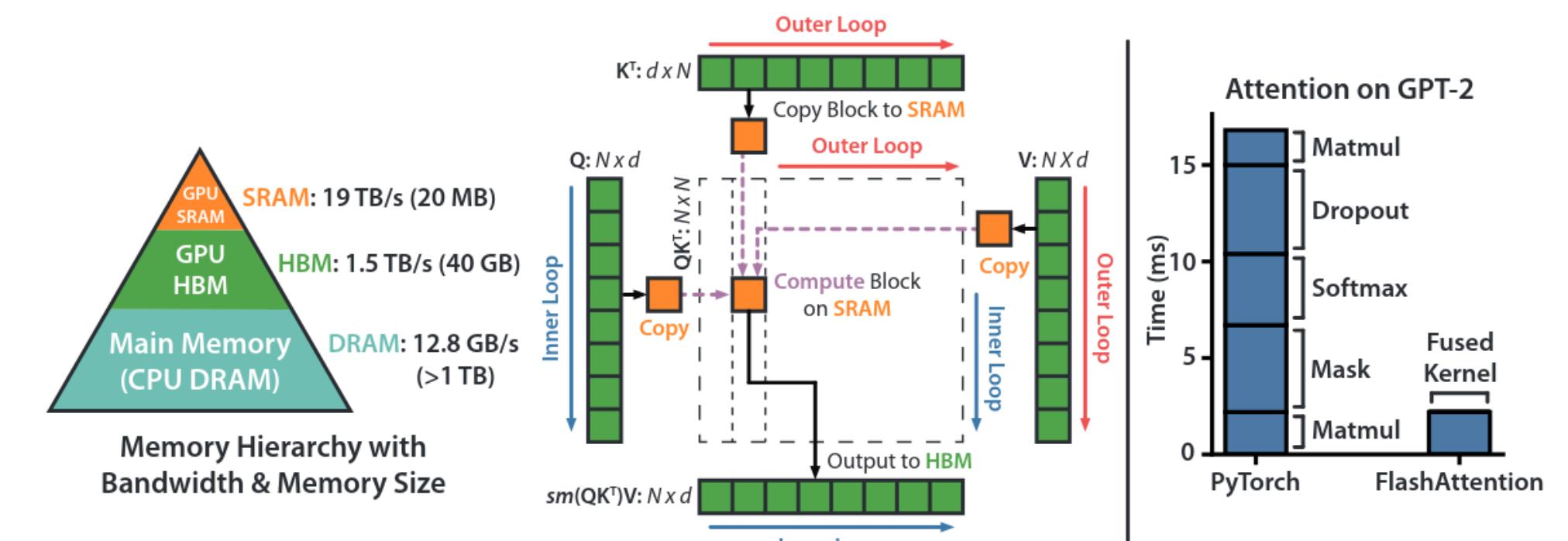
KV cache

batching

T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
S_1	S_1	S_1	S_1				
S_2	S_2	S_2					
S_3	S_3	S_3	S_3				
S_4	S_4	S_4	S_4	S_4			

SEVERAL KINDS OF OPTIMISATIONS

1. Internal transformers structure data flow
2. Data batching / serving
3. Tensor and Pipeline Parallelism
4. Quantitation
5. Automatic Prefix Caching



SELF-ATTENTION OPTIMATIONS

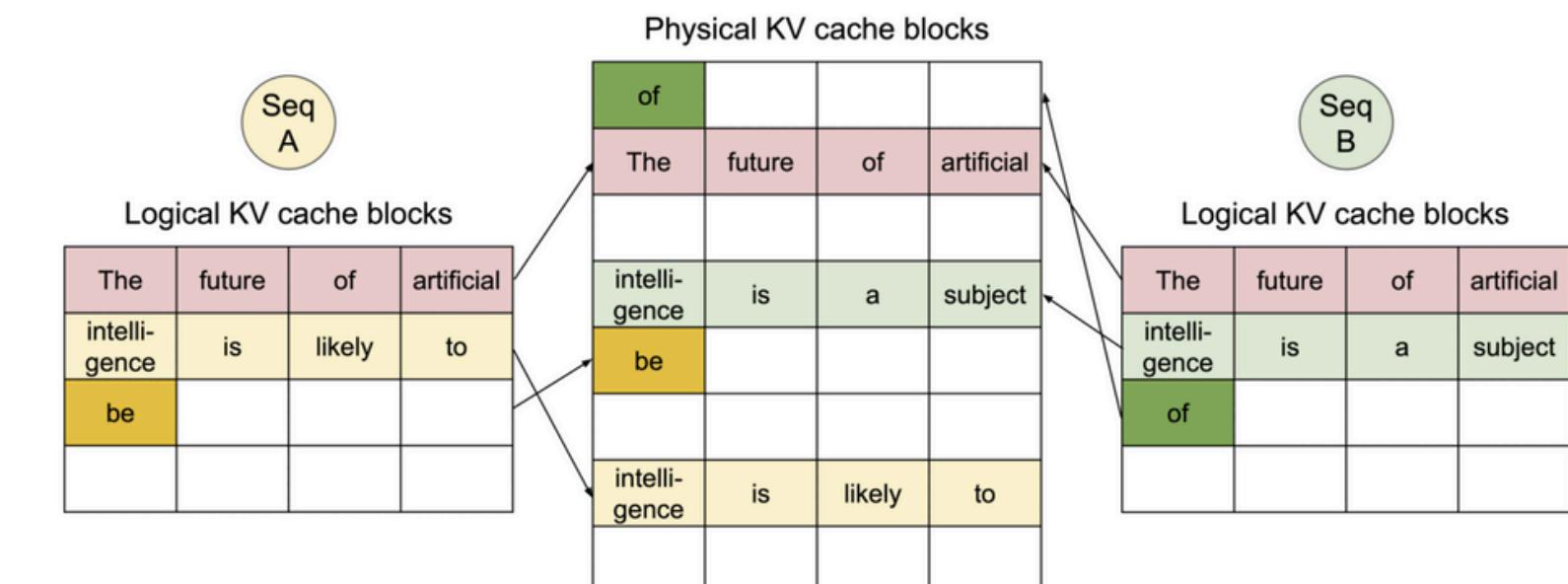
1. Paged Attention

KV cache may take up a large amount of memory for large models and long sequences.

PagedAttention optimizes memory use by **partitioning the KV cache into blocks that are accessed through a lookup table**.

KV cache is not stored in contiguous memory, and blocks are allocated as needed, increasing GPU utilization on memory-bound workloads, so more inference batches can be supported.

5. Seq A and B generated 3rd token.

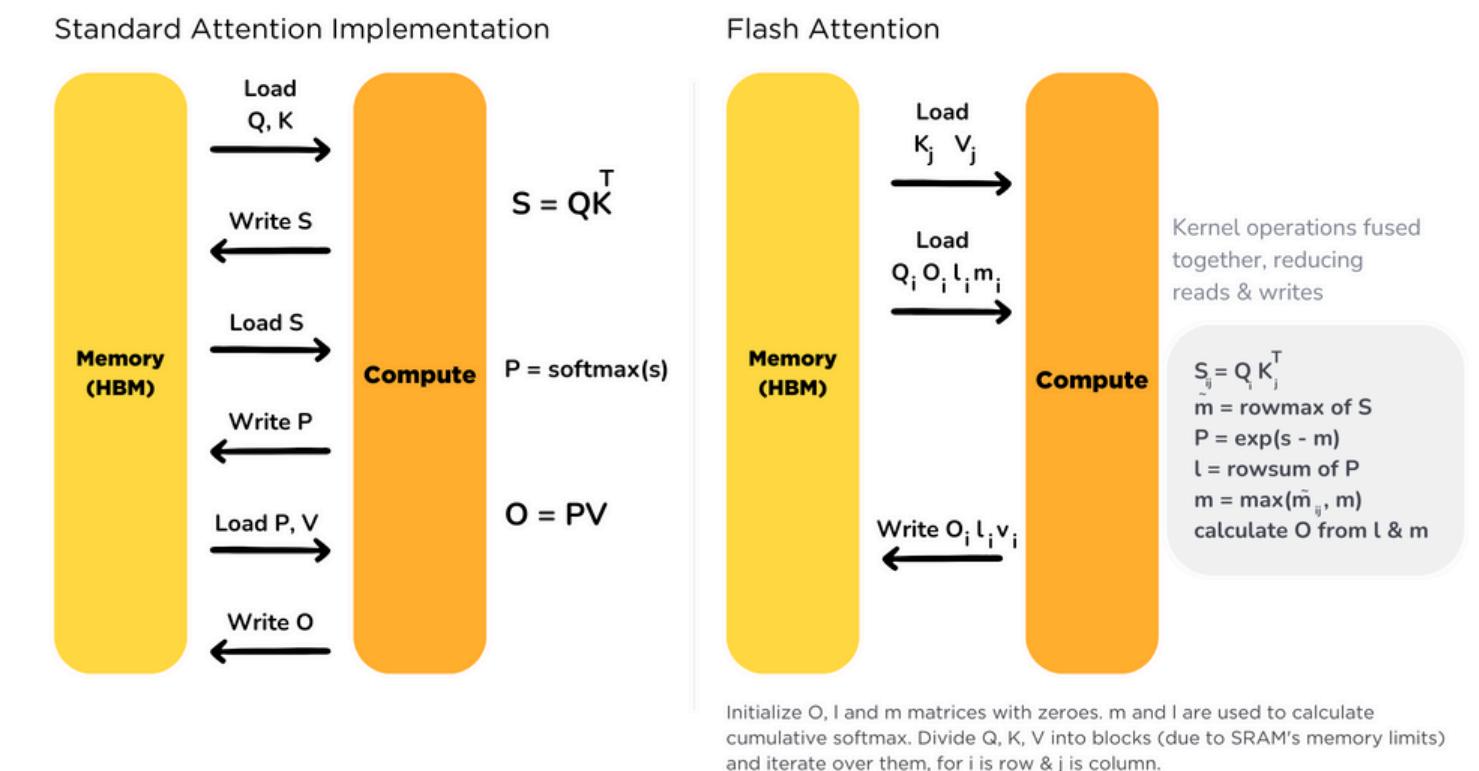
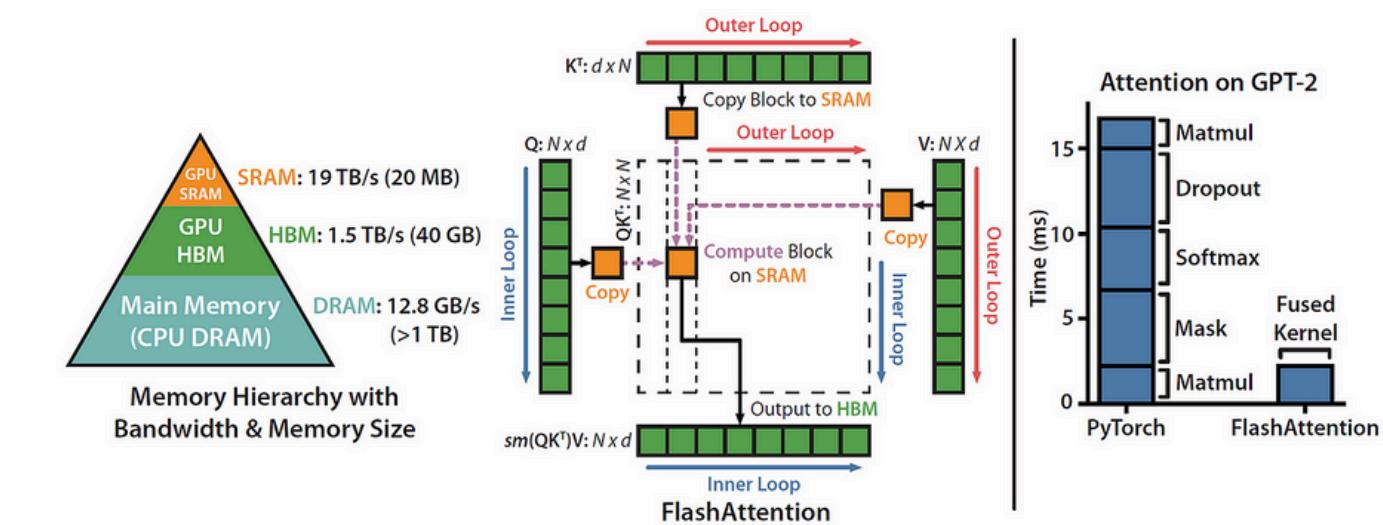


SELF-ATTENTION OPTIMATIONS

2. Flash Attention

Self-attention mechanism is quadratic in time and memory complexity.

Flash Attention make **factorized operations on keys, queries, and values**, instead of loading these from CPU High Bandwidth Memory (HBM) to GPU on-chip SRAM, performing a single step of the attention mechanism and writing it back to HBM, as standard attention implementations does.



SERVING / BATCHING OPTIMATIONS

1. Continuous Batching

Unlike traditional deep learning models, batching for LLMs may not take advantage of full GPU resources.

Continuous Batching performs a iteration-level scheduling, by regularly running queries in the same forward step of the LLM and also removing them when they are finished.

T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
S_1	S_1	S_1	S_1				
S_2	S_2	S_2					
S_3	S_3	S_3	S_3				
S_4	S_4	S_4	S_4	S_4			

T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
S_1	S_1	S_1	S_1				
S_2	S_2	S_2					
S_3	S_3	S_3	S_3				
S_4	S_4	S_4	S_4	S_4			

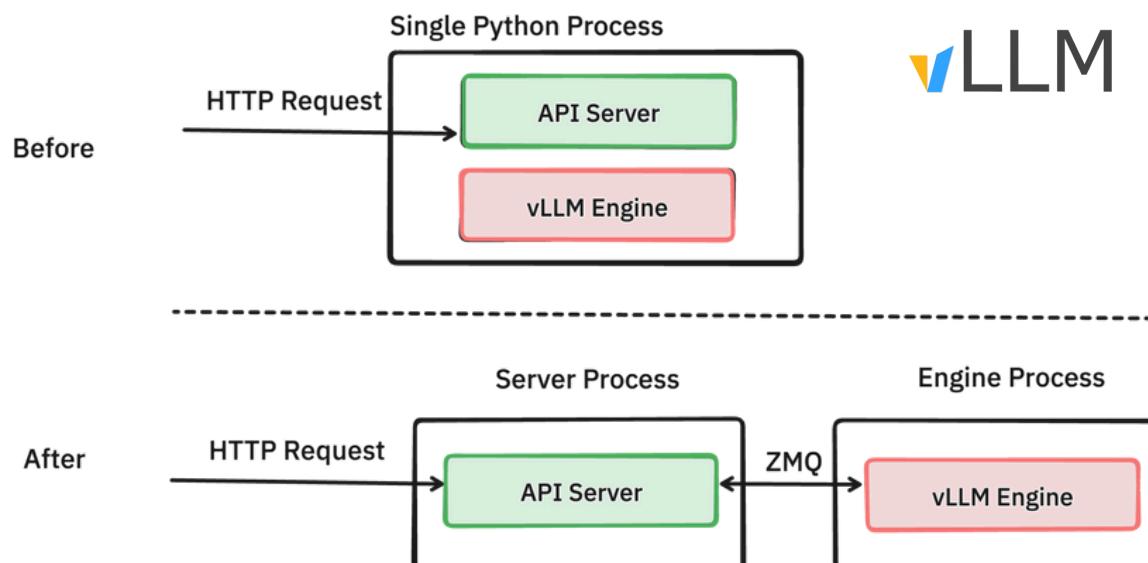
T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
S_1	S_1	S_1	S_1	S_1	END	S_6	S_6
S_2	END						
S_3	S_3	S_3	S_3	S_3	END	S_5	S_5
S_4	S_7						

T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
S_1	S_1	S_1	S_1	S_1	END		
S_2	END						
S_3	S_3	S_3	S_3	S_3	END		
S_4	END						

System-level batching optimization instead of static client-controlled batching.

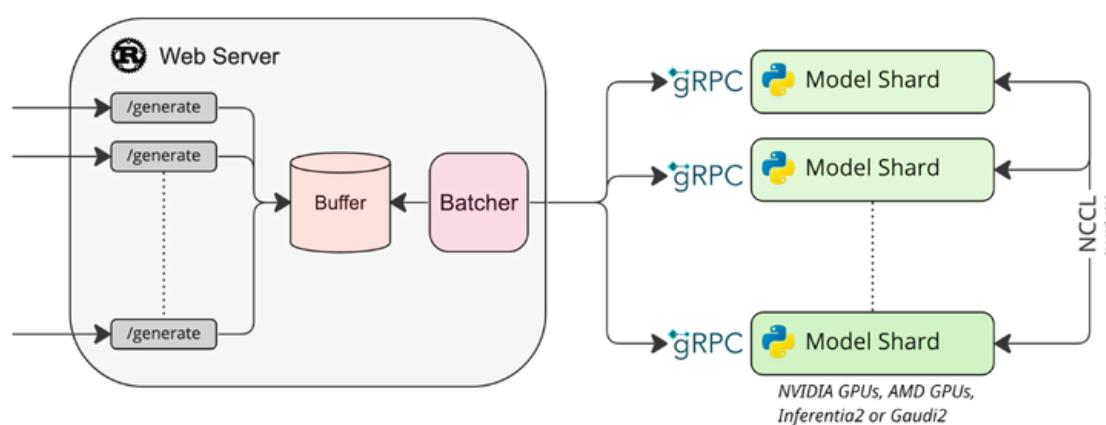
SERVING / BATCHING OPTIMATIONS

2. Asynchronous output processing



Text Generation Inference

Fast optimized inference for LLMs



```
import asyncio
from openai import OpenAI

# 1/ Wait for vllm server to be available and retrieve model
openai_api_key = "EMPTY"
openai_api_base = "http://localhost:8000/v1"
client = OpenAI(
    base_url=openai_api_base,
    api_key=openai_api_key)

model = client.models.list().data[0].id

# 2/ Create async functions to request vllm server through openAI API
async def doCompletion(model, messages, extra_body, tweet):
    completion = client.chat.completions.create(
        model=model,
        messages=messages,
        extra_body=extra_body)
    return tweet, completion.choices[0].message.content

# 3/ Create async message iterator
async def messageIterator():
    for tweet in tweets:
        yield tweet, [
            {
                "role": "system",
                "content": system_prompt
            },
            {
                "role": "user",
                "content": Template(user_prompt).substitute(tweet=tweet)
            }
        ]

# 4/ Create coroutines
async def run_all():
    # Asynchronously call the function for each prompt
    tasks = [
        doCompletion(model, it[1], extra_body, it[0])
        async for it in messageIterator()
    ]
    # Gather and run the tasks concurrently
    results = await asyncio.gather(*tasks)
    return results
```

python async requests

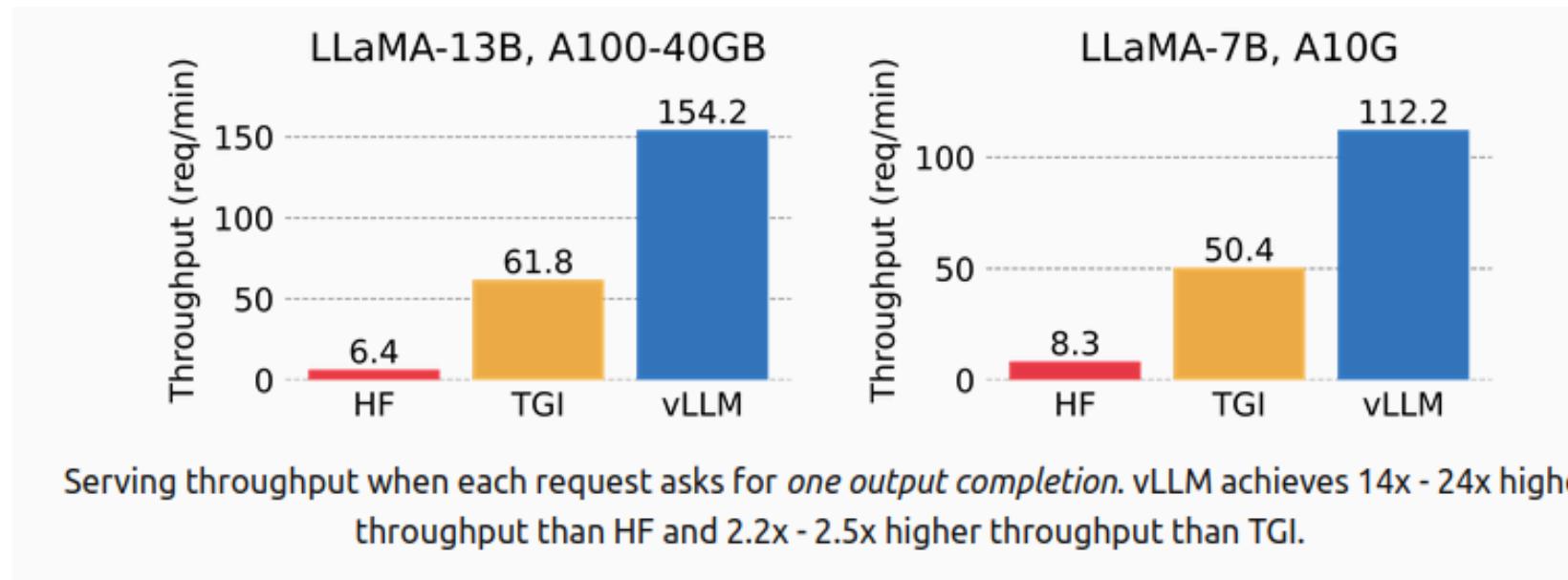
Hugging face Transformer LLM-based annotations



Jupyter notebook

[https://github.com/jimenaRL/llmBenchmarks/blob/main/venice/
venice_some4dem_workshop_transformers.ipynb](https://github.com/jimenaRL/llmBenchmarks/blob/main/venice/venice_some4dem_workshop_transformers.ipynb)

BENCHMARKS - vLLM 0.6



country	days per label in 1 gpu HF	days per label in 1 gpu vLLM
us	64,6	3,2
brazil	32,8	1,6
turkey	31,3	1,6

Key results:

Up to 24x higher throughput vs. HuggingFace Transformers

Up to 3.5x higher throughput vs. TGI

These gains are largely attributed to Paged Attention.

zephyr-7b-beta on a NVIDIA Tesla V100 32G

US dataset ~5.5 millions of X/Twitter bios

Brazil dataset ~2.8 millions of X/Twitter bios

Turkey dataset ~2.7 millions of X/Twitter bios

>>> From more than 2 months to 3 days

[vLLM's blog post](#)

LLaMA-7B on one NVIDIA A10G GPU

and

LLaMA-13B on one NVIDIA A100 GPU (40GB)

 vLLM

Quantized models on HF

The screenshot shows the Hugging Face Model Card for the **Meta-Llama-3-8B** model. The card includes the following details:

- Model card**: Shows a note about a newer version and a "Gated model" access notice.
- Downloads last month**: 2,081,998
- Safetensors**: Model size: 8B params
- Inference Providers**: Text Generation, with a prompt input field: "Your prompt here..."
- Libraries**: Transformer, Inference, Inference Providers, Notebooks, Google Colab, Kaggle, Local Apps, vLLM, TGI.
- Callout**: A green callout box at the bottom right provides instructions: "Browse Quantizations to use this model in llama.cpp, Ollama, LM Studio, or any compatible app."

At the top, the Hugging Face navigation bar is visible with links to Models, Datasets, Spaces, Community, Docs, Enterprise, Pricing, and a user profile icon.

Quantized models on HF

The screenshot shows a web browser window displaying the Hugging Face Model Hub at huggingface.co/models?other=base_model:quantized:meta-llama%2FLlama-3.1-8B-Instruct&sort=downloads. The page lists 455 quantized models, sorted by most downloads. A large arrow on the left side points from the search bar area towards the list of models.

Search Bar: Models - Hugging Face

Page Headers: Models, Datasets, Spaces, Community, Docs, Enterprise, Pricing

Left Sidebar (Other tab selected):

- Model Tree:
 - meta-llama/Llama-3.1-8B-Instruct
 - Adapters
 - Finetunes
 - Quantizations (selected)
 - Merges
- Apps:
 - llama.cpp
 - LM Studio
 - Jan
 - Backyard AI
 - Draw Things
 - DiffusionBee
 - Jellybox
 - RecurseChat
 - Msty
 - Sanctum
 - Invoke
 - JoyFusion
 - LocalAI
 - vLLM
 - node-llama-cpp
 - Ollama
 - TGI
 - MLX LM
 - Docker Model Runner
- Inference Providers:
 - Select all
 - Fireworks
 - Nebius AI
 - Novita
 - Featherless AI
 - Together AI
 - SambaNova
 - Cerebras
 - Nscale
 - Hyperbolic
 - Groq
 - fal
 - Cohere
 - Replicate
 - HF Inference API
- Misc:
 - Inference Endpoints
 - text-generation-inference
 - Eval Results
 - Merge
 - 4-bit precision
 - custom_code
 - 8-bit precision
 - text-embeddings-inference
 - Carbon Emissions
 - Mixture of Experts

Results List: Models 455

- unslot/Meta-Llama-3.1-8B-Instruct-bnb-4bit
- RedHatAI/Meta-Llama-3.1-8B-Instruct-FP8
- lmstudio-community/Meta-Llama-3.1-8B-Instruct-GGUF
- unslot/Meta-Llama-3.1-8B-Instruct-unslot-bnb-4bit
- RedHatAI/Meta-Llama-3.1-8B-Instruct-FP8-dynamic
- RedHatAI/Meta-Llama-3.1-8B-Instruct-quantized.w4a16
- RedHatAI/Meta-Llama-3.1-8B-Instruct-quantized.w8a8
- shenzi-wang/Llama3.1-8B-Chinese-Chat
- unslot/Llama-3.1-8B-Instruct-bnb-4bit
- unslot/Llama-3.1-8B-Instruct-GGUF
- MaziyarPanahi/Meta-Llama-3.1-8B-Instruct-GGUF
- modularai/Llama-3.1-8B-Instruct-GGUF
- unslot/meta-Llama-3.1-8B-unslot-bnb-4bit
- SanctumAI/Meta-Llama-3.1-8B-Instruct-GGUF
- bartowski/Meta-Llama-3.1-8B-Instruct-GGUF
- amd/Llama-3.1-8B-Instruct-FP8-KV
- unslot/Llama-3.1-8B-Instruct-unslot-bnb-4bit
- mlabonne/Meta-Llama-3.1-8B-Instruct-abliterated-GGUF
- Saxo/Linkbricks-Horizon-AI-Korean-llama-3.1-sft-dpo...
- GPT4All-Community/Meta-Llama-3.1-8B-Instruct-128k-G...

vLLM/TGI frameworks - docker serving

```
jimena.royoletelier at gepehu in ~/work/dev/llmBenchmarks (main•) (vllm)
$ docker build -f Dockerfile.cpu -t vllm-cpu-env --shm-size=4g .
```

i/ Build the image

```
jimena.royoletelier at gepehu in ~/work/dev/llmBenchmarks (main•) (vllm)
$ docker run -d --rm --name zephyr-vllm-cpu-env \
  -v /home/jimena.royoletelier/storage/hf_cache:/root/.cache/huggingface \
  -p 8000:8000 \
  vllm-cpu-env \
  --model HuggingFaceH4/zephyr-7b-beta
699f5472d012634ale705e5a3dc2d55722885ce80943baa5f56dcf5f8e370dcd
```

ii/ Run a container implementing a vllm server

vllm server through docker image

```
$ curl http://localhost:8000/v1/chat/completions \
-H "Content-Type: application/json" \
-H "Authorization: Bearer <api-key-if-applicable>" \
-d '{
    "model": "meta-llama/Llama-3.1-8B-Instruct",
    "messages": [
        {"role": "system", "content": "You are an expert in French politics."}

        {"role": "user", "content": "Please classify the following social media posts (that were posted in the weeks leading up to the 2022 presidential election in France) according to whether it express support or positive attitudes towards Le Pen in this election. You must use only the information contained in the message. Be concise and answer only \"YES\" or \"NO\". Does the following message express support or positive attitudes towards Le Pen: \"and that if LePen comes to power all these fascist militias will take to the streets StopLePen\""}
    ],
    "guided_choice": ["NO", "YES"],
    "stream": false
}'
```

iii/ Request the server



curl requests