Day 7 - LLM

UMN CSS Workshop
Instructor: Alvin Zhou
Presentor: Jiacheng Huang

Learning Goals

- Learn how LLM works
- Practice using LLM to label text and image

Recap of *Day 3* - word embeddings

- Each word is represented by a vector

Recap of *Day 3* - word embeddings

- Each word is represented by a vector

Cat: [0.0074, 0.0030, -0.0105, ..., 0.0002]

Recap of *Day 3* - word embeddings

- Each word is represented by a vector

Cat: [0.0074, 0.0030, -0.0105, ..., 0.0002]

Gorgeous: [0.21, -0.04, 0.11, ..., 0.08]

Xiangling = Gorgeous + Cat

Recap of *Day 3* - word embeddings

- Each word is represented by a vector

CAT: [0.0074, 0.0030, -0.0105, ..., 0.0002]

- Words with similar meanings are close in vector space
- Captures semantic and syntactic similarity
- “king” – “man” + “woman” ≈ “queen”
- “love” and “like” have similar embeddings

Words often have multiple meanings

John went to the **bank** to withdraw cash.

He sat on the river **bank.**

Words often have multiple meanings

John went to the **bank** to withdraw cash.

He sat on the river **bank.**

John picks up a **magazine**.

Susan works for a **magazine**.

Words often have multiple meanings

John went to the **bank** to withdraw cash

He sat on the river **bank**

Homonyms
(two unrelated meanings)

John picks up a **magazine**.

Susan works for a **magazine**.

Polysemy
(two closely related meanings)

Limitation of traditional word embeddings

Traditional word embeddings:

- Each word has a **single vector** (same vector in every context).

Limitation:
Cannot capture **polysemy** (subtle variations of meaning) or **homonyms** (entirely different meanings).

Embeddings in LLM

Large language models (LLMs) like BERT, GPT:

- Assign **different vectors** to the **same word** in different contexts.
- Vectors **update dynamically** as the model processes the sentence.

Example:

- “bank” in “John went to the bank to withdraw cash” (financial)
- “bank” in “He sat on the river bank” (geographic)

Embeddings in LLM

- “bank” in “John went to the bank to withdraw cash” (financial)
- “bank” in “He sat on the river bank” (geographic)

➜ LLM embeddings are very different because the context changes the meaning significantly.

- “magazine” in “John picks up a magazine.” (physical object – printed publication)
- “magazine” in “Susan works for a magazine.” (organization – periodical publisher)

LLMs <u>use more similar vectors</u> for polysemous meanings than for homonymous meanings.

Difference from LLM to Doc2Vec

LLM: The embedding of vector is context dependent, so we can not generate a embedding to work without context.

E.g., Embedding of "**cat**" 🙅

Embedding of "the big **cat** that is beautiful" 🙆

*theoretically if the LLM is smart enough they should know the cat in this context refer to Xiangling

How can I get embedding from LLM?

ELMo

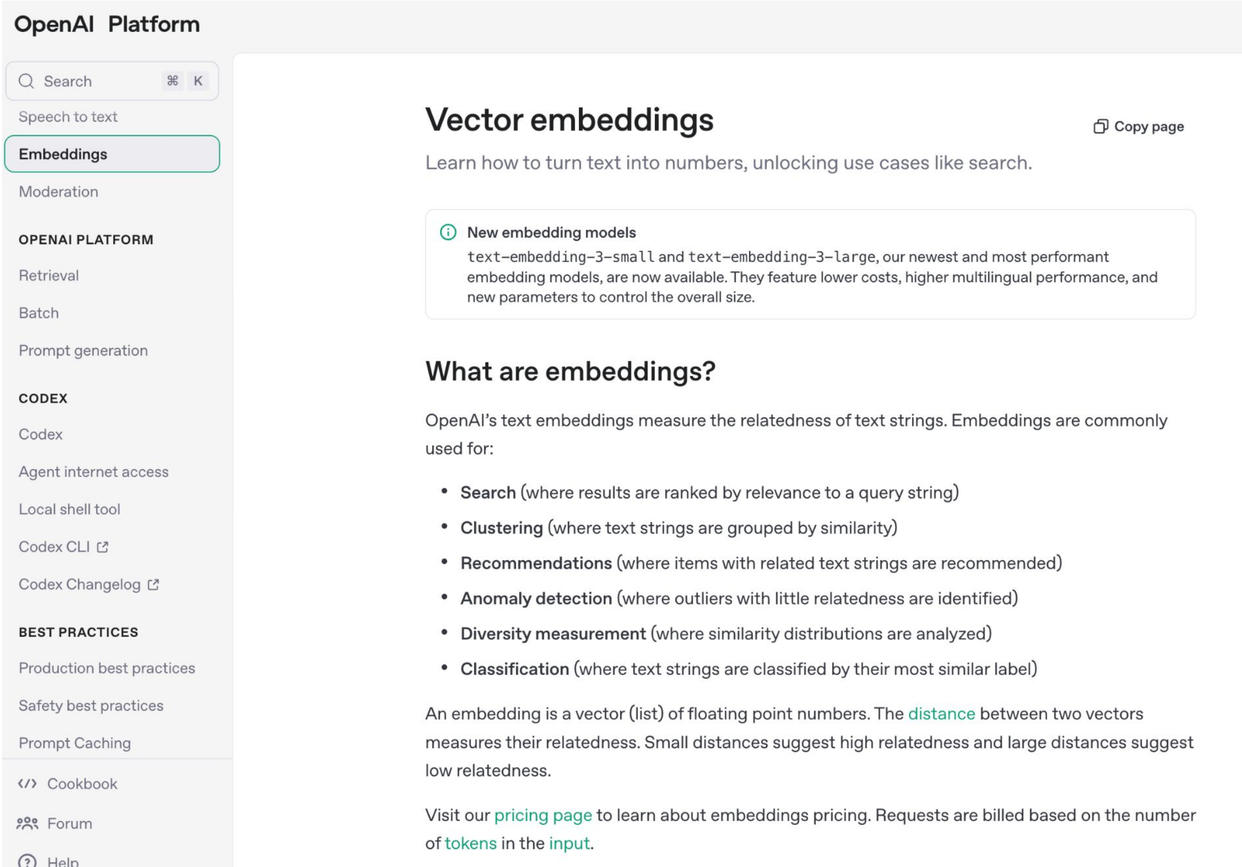
Introduced by Peters et al. in Deep contextualized word representations

Embeddings from Language Models, or **ELMo**, is a type of deep contextualized word representation that models both (1) complex characteristics of word use (e.g., syntax and semantics), and (2) how these uses vary across linguistic contexts (i.e., to model polysemy). Word vectors are learned functions of the internal states of a deep bidirectional language model (biLM), which is pre-trained on a large text corpus.

A biLM combines both a forward and backward LM. ELMo jointly maximizes the log likelihood of the forward and backward directions. To add ELMo to a supervised model, we freeze the weights of the biLM and then concatenate the ELMo vector $\mathbf{ELMO}_k^{task}$ with $\mathbf{x}_k$ and pass the ELMO enhanced representation $[\mathbf{x}_k; \mathbf{ELMO}_k^{task}]$ into the task RNN. Here $\mathbf{x}_k$ is a context-independent token representation for each token position.

Image Source: here

How can I get embedding from LLM?



How does LLM actually work?

How does LLM actually work?

“It predicts the probability of the next word.”

How does LLM actually work?

“It leverages the contextual information.”

Word Vectors In, Predictions Out

- Sentence → Each word becomes an embedding.
- These word embedding pass through dozens of *transformer* layers (in GPT-3, 96 layers!).
- Each layer adjusts the word's meaning based on surrounding words
  - Update the word's embedding vector
- By the final layer, each word vector has been refined to predict the most likely next word.
  - GPT: Left-to-right (causal)
  - Bidirectional (masked)

Transformer: Attention is all you need

1. **Attention step:** words “look around” for other words that have relevant context and share information with one another.

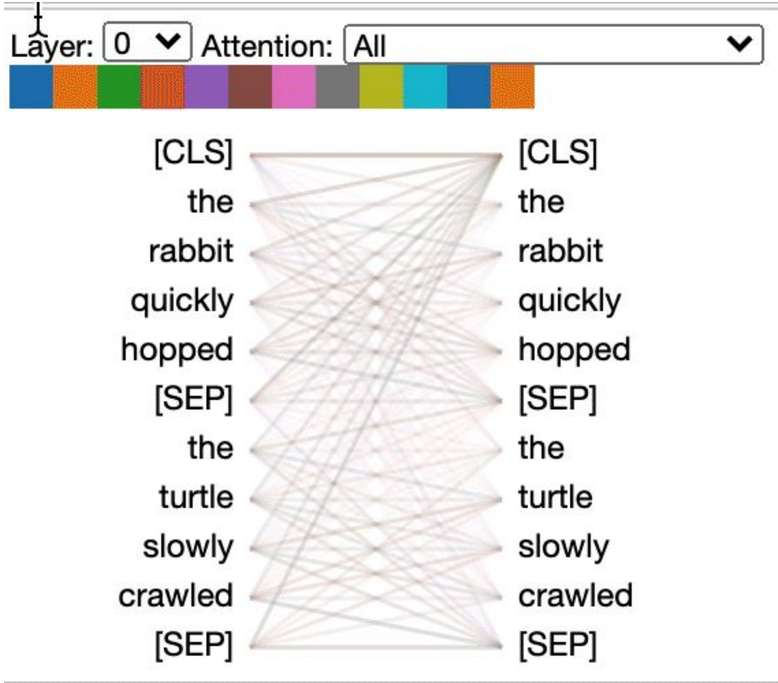
BertViz

Visualize Attention in NLP Models

Quick Tour • Getting Started • Colab Tutorial • Paper

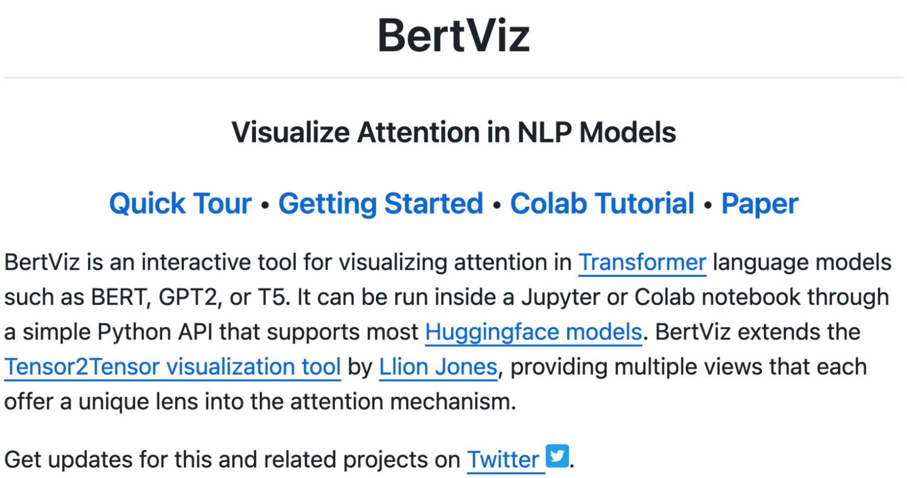
BertViz is an interactive tool for visualizing attention in Transformer language models such as BERT, GPT2, or T5. It can be run inside a Jupyter or Colab notebook through a simple Python API that supports most Huggingface models. BertViz extends the Tensor2Tensor visualization tool by Llion Jones, providing multiple views that each offer a unique lens into the attention mechanism.

Get updates for this and related projects on Twitter.



Transformer: Attention is all you need

1. **Attention step:** words “look around” for other words that have relevant context and share information with one another.

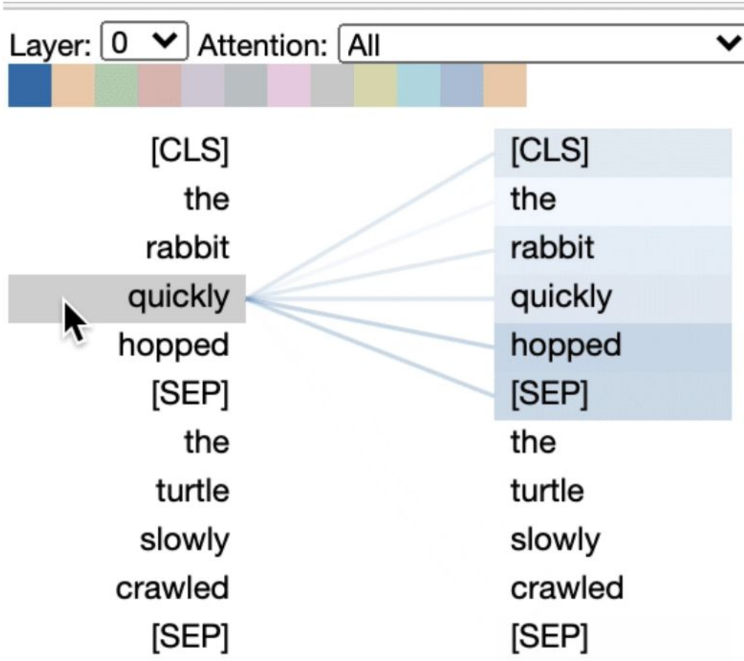

BertViz

Visualize Attention in NLP Models

Quick Tour • Getting Started • Colab Tutorial • Paper

BertViz is an interactive tool for visualizing attention in Transformer language models such as BERT, GPT2, or T5. It can be run inside a Jupyter or Colab notebook through a simple Python API that supports most Huggingface models. BertViz extends the Tensor2Tensor visualization tool by Llion Jones, providing multiple views that each offer a unique lens into the attention mechanism.

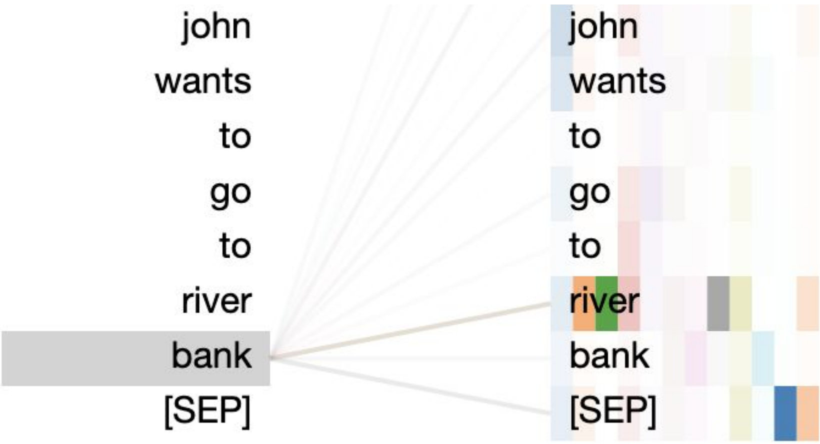
Get updates for this and related projects on Twitter.

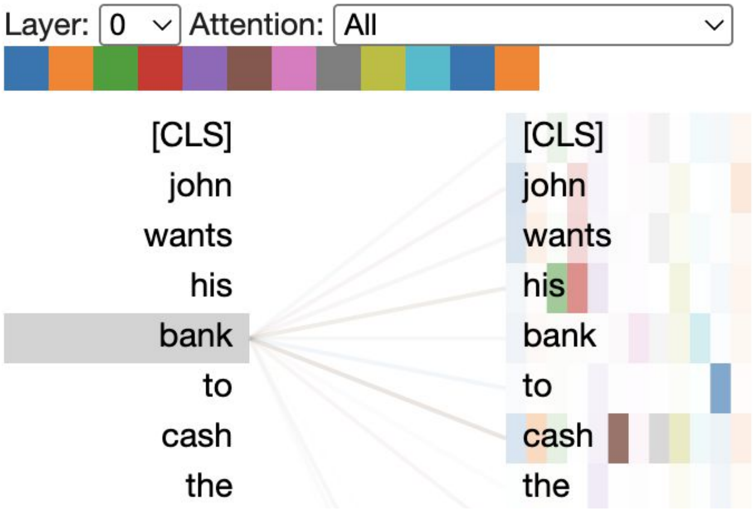


Word Vectors In, Predictions Out

John wants his bank to cash the

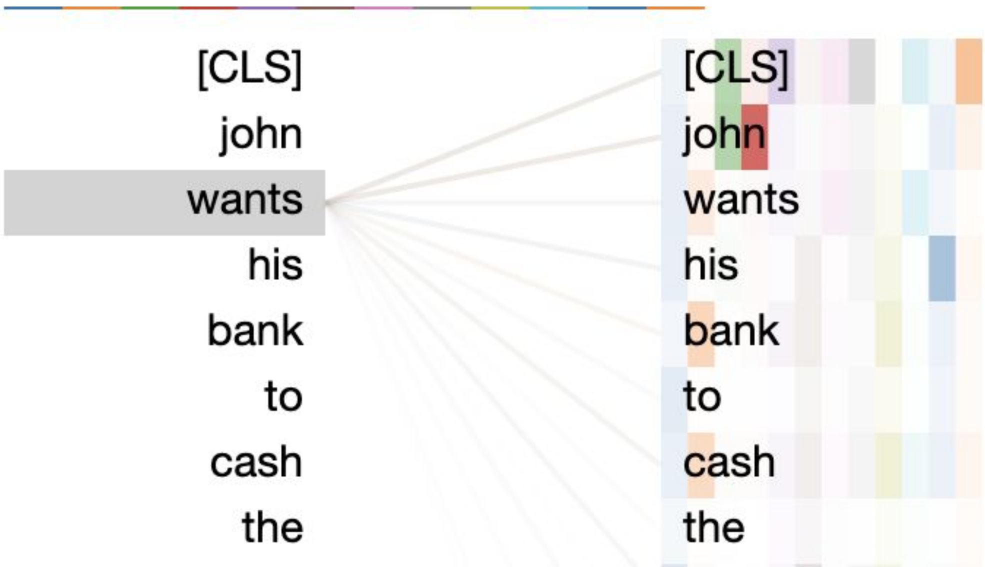
Word Vectors In, Predictions Out





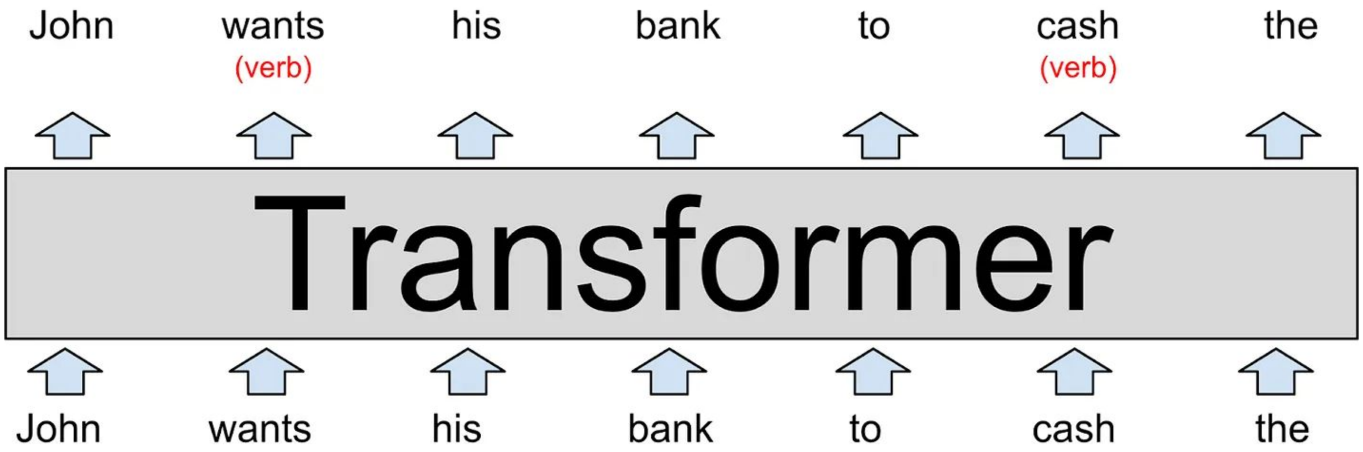
John wants his bank to cash the

Word Vectors In, Predictions Out



John wants his bank to cash the

Word Vectors In, Predictions Out

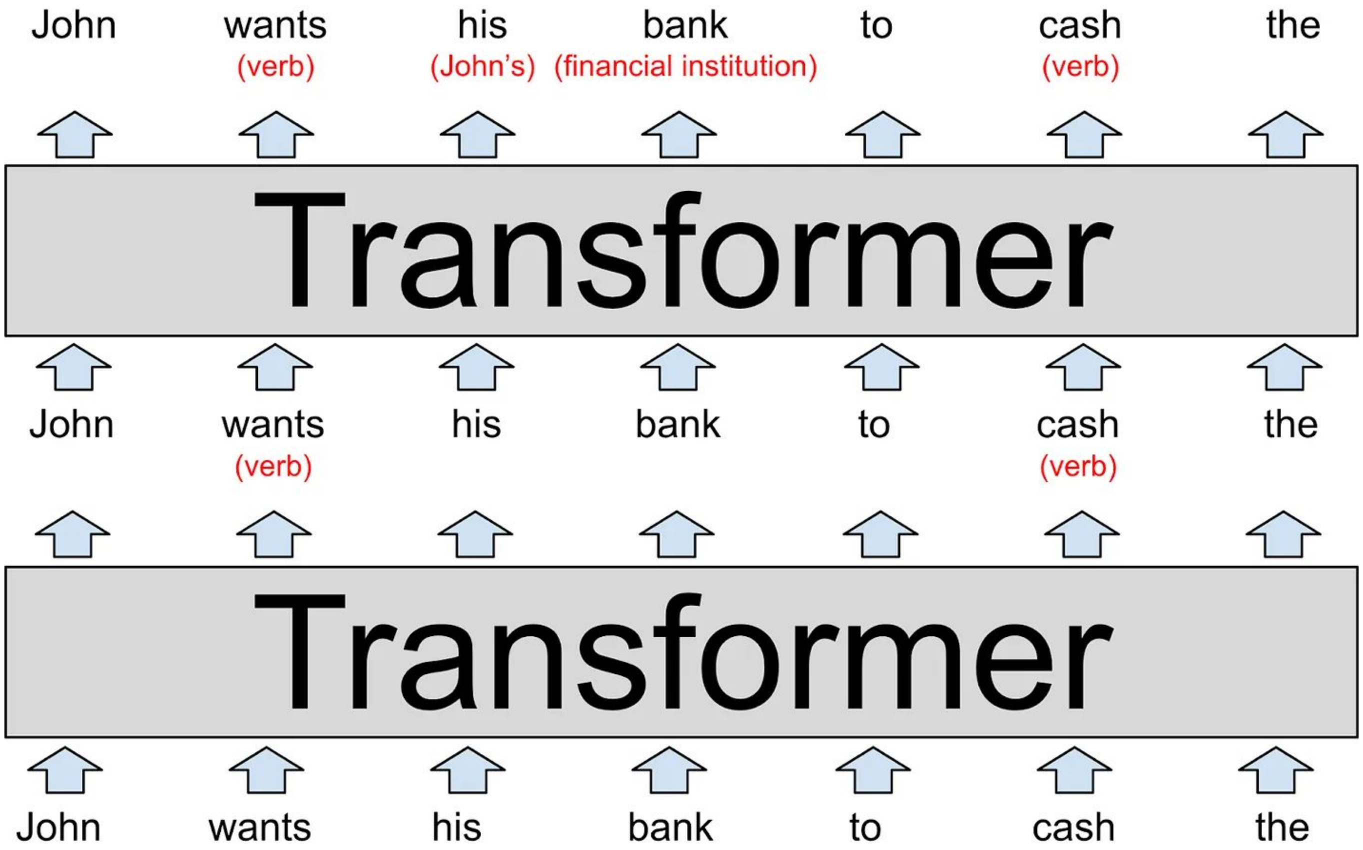


Transformer: Attention is all you need

1. **Attention step:** words “look around” for other words that have relevant context and share information with one another.

2. **Feed-forward step:** each word “thinks about” information gathered in previous attention steps and tries to predict the next word.

Word Vectors In, Predictions Out

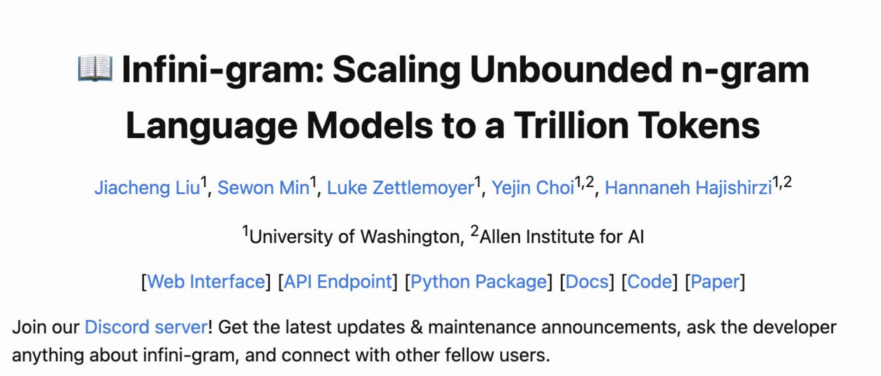


- "John wants his bank to cash the **check**."
- "John wants his bank to cash the **money order**."
- "John wants his bank to cash the **bond**."

See what’s inside the black box: XAI

- https://huggingface.co/spaces/liujch1998/infini-gram

- https://playground.allenai.org



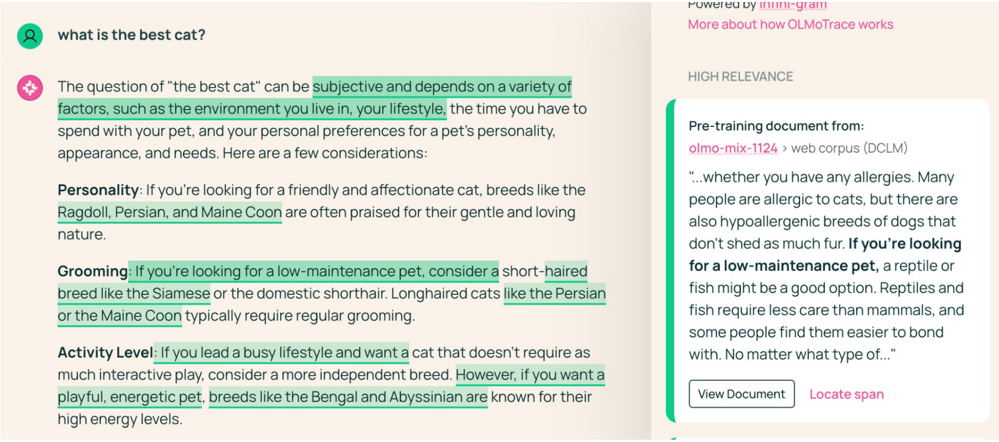
📖 **Infini-gram: Scaling Unbounded n-gram Language Models to a Trillion Tokens**

Jiacheng Liu[1], Sewon Min[1], Luke Zettlemoyer[1], Yejin Choi[1,2], Hannaneh Hajishirzi[1,2]

[1]University of Washington, [2]Allen Institute for AI

[Web Interface] [API Endpoint] [Python Package] [Docs] [Code] [Paper]

Join our Discord server! Get the latest updates & maintenance announcements, ask the developer anything about infini-gram, and connect with other fellow users.



what is the best cat?

The question of "the best cat" can be subjective and depends on a variety of factors, such as the environment you live in, your lifestyle, the time you have to spend with your pet, and your personal preferences for a pet's personality, appearance, and needs. Here are a few considerations:

Personality: If you're looking for a friendly and affectionate cat, breeds like the Ragdoll, Persian, and Maine Coon are often praised for their gentle and loving nature.

Grooming: If you're looking for a low-maintenance pet, consider a short-haired breed like the Siamese or the domestic shorthair. Longhaired cats like the Persian or the Maine Coon typically require regular grooming.

Activity Level: If you lead a busy lifestyle and want a cat that doesn't require as much interactive play, consider a more independent breed. However, if you want a playful, energetic pet, breeds like the Bengal and Abyssinian are known for their high energy levels.

Powered by infini-gram
More about how OLMoTrace works

HIGH RELEVANCE

Pre-training document from:
olmo-mix-1124 › web corpus (DCLM)

"...whether you have any allergies. Many people are allergic to cats, but there are also hypoallergenic breeds of dogs that don't shed as much fur. **If you're looking for a low-maintenance pet,** a reptile or fish might be a good option. Reptiles and fish require less care than mammals, and some people find them easier to bond with. No matter what type of..."

View Document Locate span

LLM can be do a lot of things …

Textual/ visual analysis	**Stimuli**	**Simulation**	**Participants**

… …

LLM can be do a lot of things …

Textual/ visual analysis	**Stimuli**	Simulation	Participants

… …

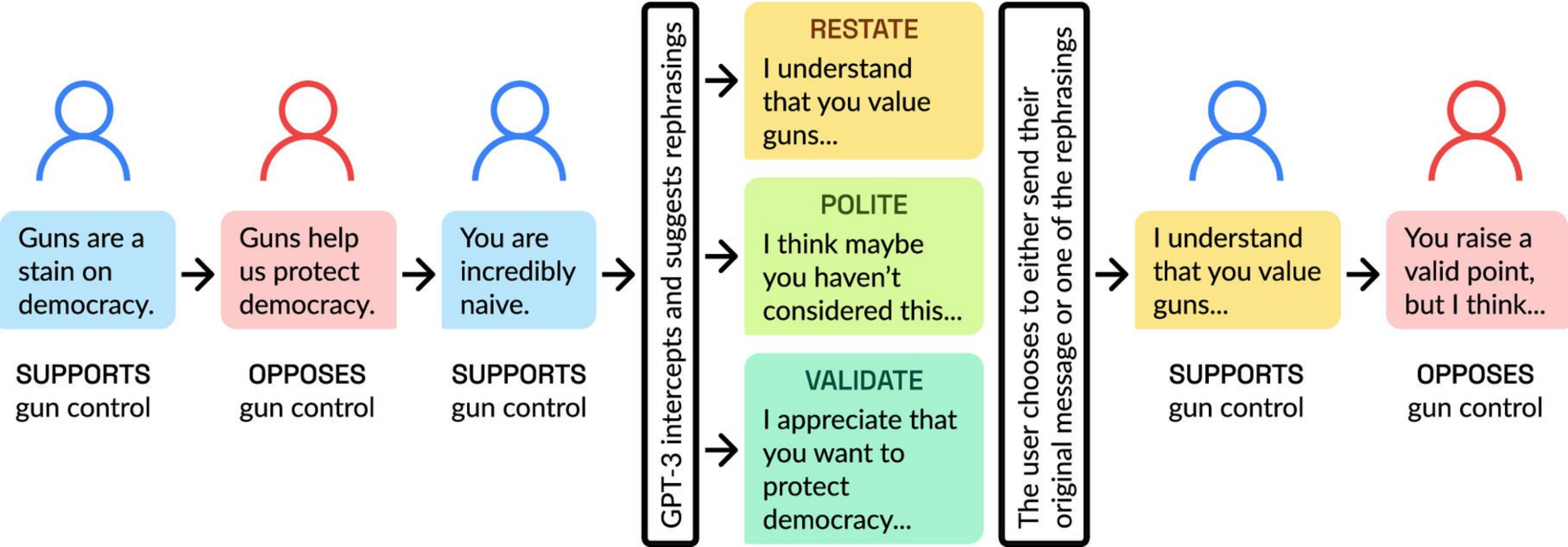
Stimuli: Argyle et al., 2023

Can real-time AI chat suggestions improve online political conversations?

Hypotheses:

- AI suggested conversations will improve perceived conversation quality.
- Will promote democratic reciprocity—respecting political opponents' rights to speak.
- No expected change in policy attitudes.

Stimuli: Argyle et al., 2023



Stimuli: Argyle et al., 2023

AI suggestions increased:

- Conversation quality (feeling understood & respected).
- Democratic reciprocity (willingness to hear political opponents).
  - Effects strongest for partners of AI-assisted users.
  - No effect on policy attitude changes.

Stimuli: Argyle et al., 2023

AI suggestions increased:

- Conversation quality (feeling understood & respected).
- Democratic reciprocity (willingness to hear political opponents).
  - Effects strongest for partners of AI-assisted users.
  - No effect on policy attitude changes.

Mechanism

- AI rephrasings → more polite, validating language (text analysis).

LLM can be do a lot of things …

Textual/ visual analysis	Stimuli	**Simulation**	Participants

… …

Simulation: Park et al., 2023

Simulation: Park et al., 2023

Agent-based modeling (ABM): A computational approach to simulate actions & interactions of autonomous agents within an environment.

Core Features:

- Agents – independent "actors" (e.g., people, animals, firms).
- Rules – simple, hand-coded decision rules (e.g., if-then logic).
- Environment – spatial or networked setting where agents interact.
- Emergence – system-level patterns arise from micro-level behaviors.

Simulation: Park et al., 2023

Micro-level ABM

Focus: Simulation of individual-level interactions based on rule/ mechanism

Outcome: Emergent social patterns from local rules

Macro-level ABM

Focus: Simulation of macro outcomes (like public health, market dynamics)

Outcome: Macro-level patterns (infection rates, herd immunity)

Let’s try to do a ABM together

Neighbor	Neighbor
Neighbor	You

Schelling’s Segregation Model

Agents – students (50% red, 50% blue)

Environment – Classroom seating grid

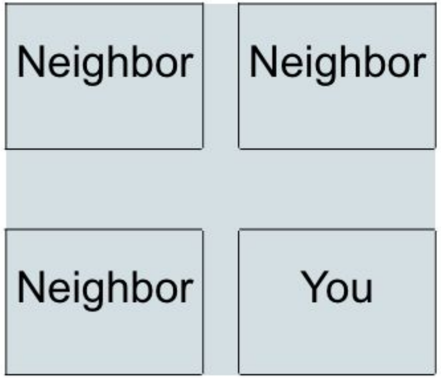
Rules – Happy if ≥ 2 neighbors are same color.

If unhappy, move to an empty seat with more same-color neighbors.

(Move one by one)

Emergence –

Let’s try to do a ABM together



Schelling’s Segregation Model

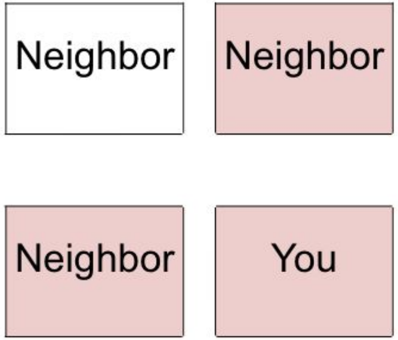
Agents – students (50% red, 50% blue)

Environment – Classroom seating grid

Rules – Happy if ≥ 2 neighbors are same color.

If unhappy, move to an empty seat with more same-color neighbors.

(Move one by one)



<u>**Emergence – Segregated seating patterns.**</u>

Simulation: Park et al., 2023

Limitations of Traditional ABM:

- Rules are static
- No long-term memory or self-reflection
- Poor at modeling nuanced social behavior
- Struggles with complex, open-ended interactions

Simulation: Park et al., 2023

	Micro-Level ABM	**Macro-Level ABM**
Focus	Individual-level interactions, specific rules	System-wide outcomes (public health, market dynamics)
Traditional Outcome	Emergent patterns (e.g., neighborhood segregation)	Macro-level trends (e.g., infection curves, herd immunity)
Generative Agents Add	- Rich cognitive states (memory, reflection) - Natural-language social behavior - Adaptive planning - Reasoning	- Heterogeneous reasoning and compliance - Realistic social interactions - Scenario testing with natural language prompts
Result	More **realistic micro-level dynamics** that still produce emergent social patterns	More **credible macro projections** grounded in nuanced individual behavior

LLM can be do a lot of things …

Textual/ visual analysis	Stimuli	Simulation	**Participants**

… …

Participants: Ashokkumar et al., 2024

Can Large Language Models (LLMs) accurately predict treatment effects in social science experiments?

Participants: Ashokkumar et al., 2024

70 U.S. survey experiments (476 treatment effects; 105,165 participants)

GPT-4 simulated average participant responses based on demographic prompts

Compared LLM predictions vs. human forecasts vs. actual experimental results

Participants: Ashokkumar et al., 2024

GPT-4 predictions surpassed human forecasters

- Even higher accuracy for unpublished studies (no exposure in training data)

GPT-4 have little biases across different subgroups:

- Didn't perform worse for Black participants, for women, or for one political party over another.

Participants: Ashokkumar et al., 2024

GPT-4 have little biases across different subgroups:

- Consistent performance across different subgroups
- Didn't perform worse for Black participants, for women, or for one political party over another.

Participants: Ashokkumar et al., 2024

GPT-4 have little biases across different subgroups:

- Consistent performance across different subgroups
- Didn't perform worse for Black participants, for women, or for one political party over another.

The paper shows consistent accuracy across gender, race, and political affiliation—do you think this suggests low bias?

LLM can be do a lot of things …

Textual/ visual analysis	**Stimuli**	**Simulation**	**Participants**

… …

Labeling Methods

Manual Labeling
Time-consuming &
Expensive

Machine Learning
Technical skills &
Computing power

Labeling Methods

Manual Labeling
Time-consuming & Expensive

LLM
Affordable & Easy

Machine Learning
Technical skills & Computing power

LabelGenius

LabelGenius

Data sharing

https://platform.openai.com/settings/organization/data-controls/sharing
https://help.openai.com/en/articles/10306912-sharing-feedback-evals-and-api-data-with-openai

Enable sharing of model feedback from the Platform

Enable all members of your organization to share feedback and chats with OpenAI to help improve and train our models. If enabled, users can click the "thumbs down" button on model responses in the Playground and choose to share their feedback and content from the current chat. Learn more.

- Disabled
- Enabled for all projects
- Enabled for selected projects

Share Logs with OpenAI

Turn on sharing with OpenAI for logs from your organization to help us develop and improve our services, including for improving and training our models. Only inference inputs and outputs created after turning this setting on will be shared. You can change your settings at any time to disable sharing inference inputs and outputs.

You're enrolled for up to 11 million complimentary tokens per day

- Disabled
- Enabled for all projects
- Enabled for selected projects

Save

LabelGenius

Module	Function	Purpose	Single-theme	Multiple-theme	Modality Support
CLIP	classification_CLIP_0_shot	Perform zero-shot classification with CLIP	✓		Text, Image, Text-Image Pair
(Runs locally; no data sharing required)	classification_CLIP_finetuned	Use a fine-tuned CLIP model for classification	✓		Text, Image, Text-Image Pair
	finetune_CLIP	Fine-tune CLIP on labeled data	✓		Text, Image, Text-Image Pair
GPT	classification_GPT	Perform text classification with GPT (zero-shot or few-shot)	✓	✓	Text, Image, Text-Image Pair
(Requires data sharing with third-party provider)	generate_GPT_finetune_jsonl	Prepare JSONL files for GPT fine-tuning	✓	✓	Text
	finetune_GPT	Fine-tune a GPT model on labeled data	✓	✓	Text
	price_estimation	Estimate the cost of OpenAI API calls	-	-	-

LabelGenius

Price estimation

https://openai.com/api/pricing/

OpenAI o4-mini

Our faster, cost-efficient reasoning model delivering strong performance on math, coding and vision

Price

Input:
$1.100 / 1M tokens

Cached input:
$0.275 / 1M tokens

Output:

GPT-4.1

Smartest model for complex tasks

Price

Input:
$2.00 / 1M tokens

Cached input:
$0.50 / 1M tokens

Output:
$8.00 / 1M tokens

GPT-4.1 mini

Affordable model balancing speed and intelligence

Price

Input:
$0.40 / 1M tokens

Cached input:
$0.10 / 1M tokens

Output:
$1.60 / 1M tokens

GPT-4.1 nano

Fastest, most cost-effective model for low-latency tasks

Pricing

Input:
$0.100 / 1M tokens

Cached input:
$0.025 / 1M tokens

Output:
$0.400 / 1M tokens

Prompt

Zero-shot prompt

Does the following tweet relate to politics, yes or no? Tweet: {the focal tweet}

Prompt

Zero-shot prompt

Few-shot prompt

Does the following tweet relate to politics, yes or no? Tweet: {the focal tweet}

Example1: Tweet: {the focal tweet}. Label :{xxxx}

Example2: Tweet: {the focal tweet}. Label :{xxxx}

Coding Time !