UBL Writing Day 2024 Prompt Engineering

> Input session on 2024-02-29, Leipzig University Library, Martin Czygan, software developer, author and data engineer

Random voice on the internet: "The prompt engineers will be the first self-proclaimed engineers to be replaced by AI."

> LARGE LANGUAGE MODELS ARE HUMAN-LEVEL PROMPT ENGINEERS 2211.01910.pdf

About Me

- Software Developer at Leipzig University Library, Open Data Engineer at Internet Archive, working on Internet Archive Scholar
- Misc: consultant, author, open source contributor, community organizer, former Lecturer at Lancaster University Leipzig
- curious about computers since about 1985, about machine learning since about 2011

A perspective on "AI"

- "AI" is mostly "ML"
- testing open models, mostly - models for which the research and development process is (at least somewhat) documented

Figure 1: Running Sosaka/Alpaca-native-4bit-ggml [9c1bb480] from 2023-03-21 on a 2018 laptop w/ i7-8550u CPU and w/o GPU, recorded 2023-04-19

> But there is frustration in the science community over OpenAI's secrecy around how the model was trained and what data were used, and how GPT-4 actually works. "**All of these closed-source models, they are essentially dead ends in science**," says Sasha Luccioni, a research scientist specializing in climate at HuggingFace, an open-source AI cooperative. – GPT-4 IS HERE: WHAT SCIENTISTS THINK (03/2023)

- main "serious" topic, beside haikus, is the conversion of unstructured data (e.g. "strings", "bytes") to structured data (e.g. "metadata"), information retrieval
- previous talks: NN tour (2016), PyTorch tour (2018), ML w/ Go (2018), cgosamples (2023), local models (2023)

What is a prompt? Engineering?

> A prompt is an input, a text command or a question provided to an AI model, to generate **desired output** like content or answer. The process of crafting effective and efficient prompts is called prompt design or prompt engineering. – Azure ML docs

Why does it exist at all?

- the wikipedia article about Prompt Engineering is not that old, it first appeared in 2021-10-20, a Wednesday
- I used tweet 1599971348717051904 as a joke on 2022-12-12 during an intro to programming CS class

These language models learn are META-LEARNERS:

> language models can also be understood as meta-learners where slow outer-loop gradient descent based learning is combined with **fast "in-context" learning implemented within the context activations of the model** – Language Models are Few-Shot Learners

This is new and only observed in larger models. We not only want a distribution over words, but words and tasks.

> Learning to perform a single task can be expressed in a probabilistic framework as estimating a conditional distribution `p(output|input)`. Since a general system should be able to perform many different tasks, even for the same input, **it should condition not only on the input but also on the task to be performed**. That is, it should model `p(output|input, task)`. – Language Models are Unsupervised Multitask Learners

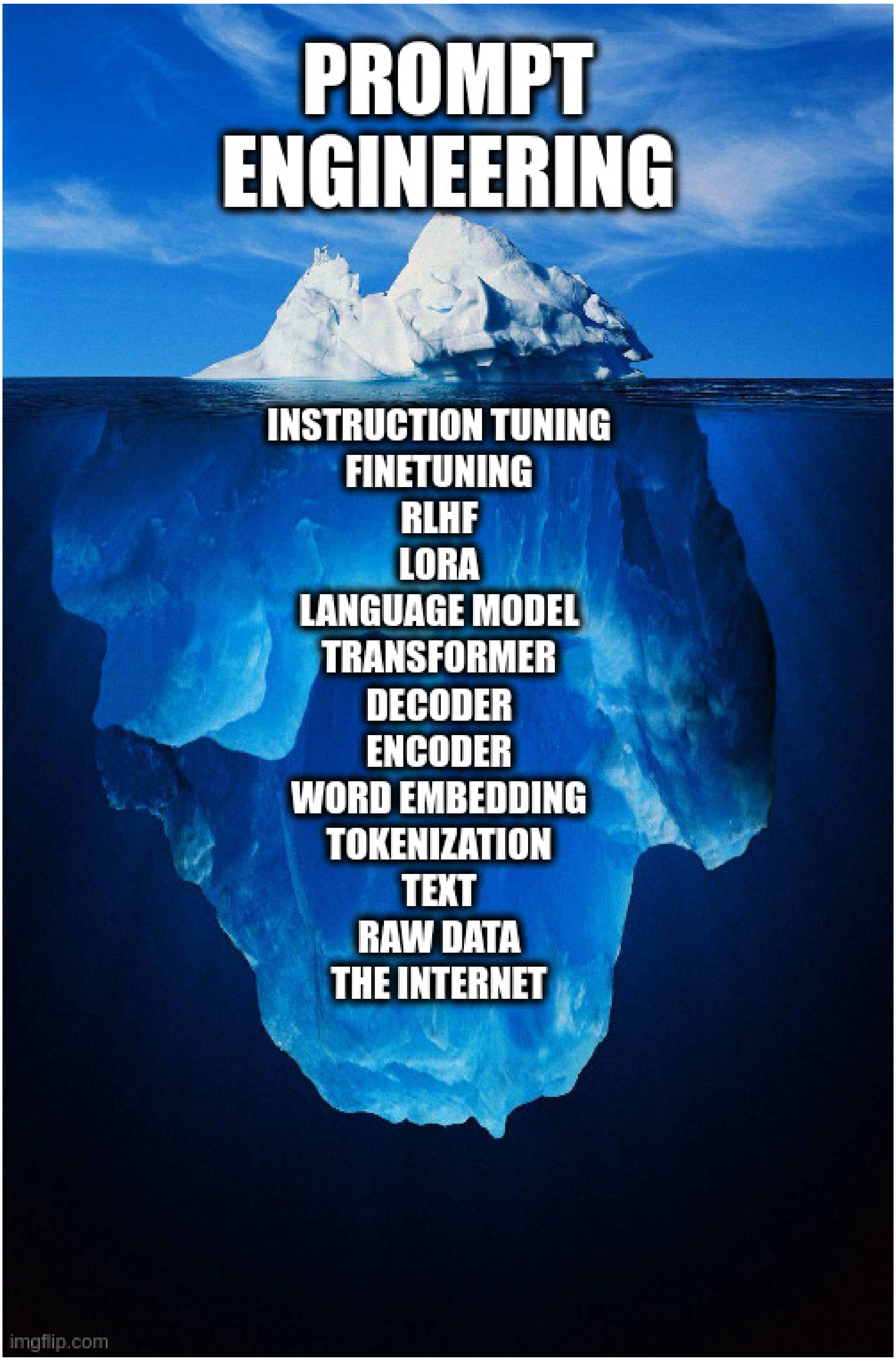


Figure 2: The LLM iceberg

Data as fuel; one early example: 45M outlinks from reddit, w/ at least 3 karma; an (early) web-based dataset ("WebText", 2019, not made publicly available).

> Manually filtering a full web scrape would be exceptionally expensive so as a starting point, we scraped all outbound links from Reddit, a social media platform, which received at least **3 karma**. This can be thought of as a heuristic indicator for whether other users found the link **interesting, educational, or just funny**. The resulting dataset, WebText, contains the text subset of these 45 million links.

There are attempts to recreate that dataset (e.g. "OpenWebText"), other open datasets include, e.g. The Pile.

Also note: beside all human input in form of training data, current "AI" requires a lot of mostly global south based labor to function, cf. The Glamorisation of Unpaid Labour: AI and its Influencers.

Classic language model

In the June 1989 issue of Scientific American, on page 122-125, we find a column, titled A potpourri of programmed prose and prosody:

COMPUTER RECREATIONS

A potpourri of programmed prose and prosody

by A. K. Dewdney

It takes about 10s on a CPU to create a language model from 400K words, using a slightly strange combination of texts, e.g. bible, python docs, etc.

Example output:

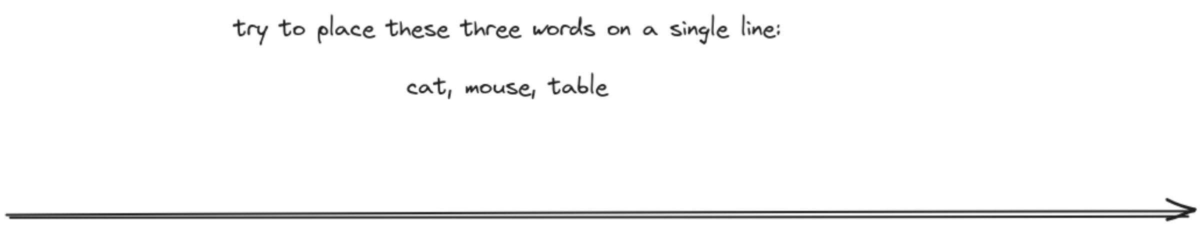
```
$ make generate
python gentext.py

with likewise the itertools module the result gives total ordering its
generally true that am the lord when thy days that were numbered of them as he
goeth in to wait for enqueued tasks have completed before ascompleted is called
without calling sysexcinfo bpo allow the creation of new features related to
fifo false if pattern is relative and then sort again
```

Well. Let's add more training data. How about 224M files of text, about 40M words. Screenie.

Embeddings and word arithmetic

Am word embedding is a representation of word, **encoding meaning**.

> Mathematically, an embedding space, or latent space, is defined as a manifold in which similar items are positioned closer to one another than less similar items.



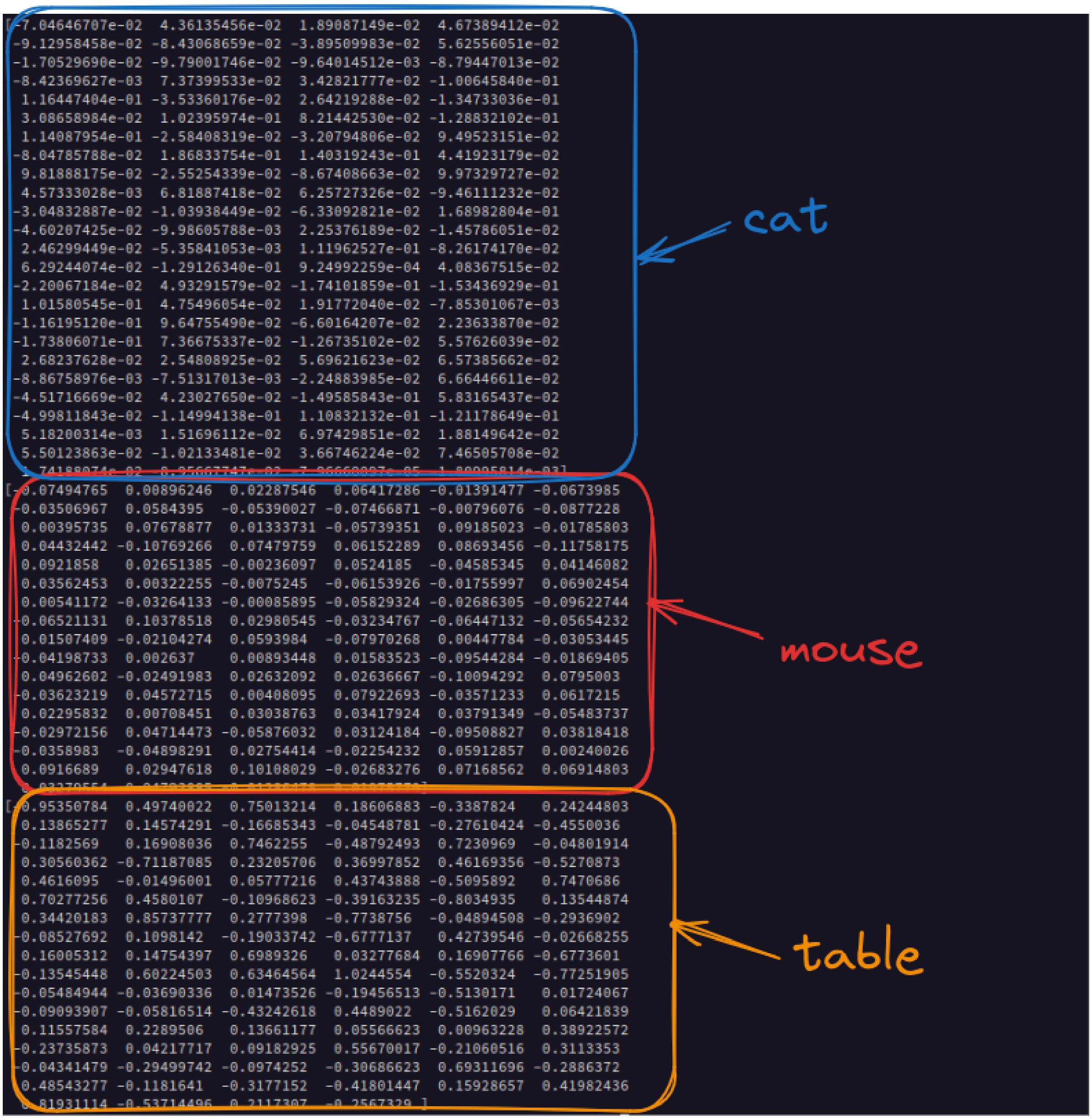
- is cat closer to mouse? is table closer to cat or mouse? …

```
$ python make_model.py

real    0m1.905s
user    0m2.284s
sys     0m1.430s
```

We are able to build some model from little data, 1M words, 6MB file.

```
$ wc -cw corpus.file
1096052 6050923 corpus.file
```

The model file is 27MB in size.



These vectors encode some “latent” structure of language:

```
$ python most_similar.py forest | column -t
air       0.914194643497467
fish      0.9120122194290161
heat      0.9068334698677063
flood     0.8980449438095093
heaps     0.8974539041519165
rivers    0.883648693561554
```

```
salt     0.8798093199729919
springs  0.8796758055686951
herbs    0.8791329264640808
furnace  0.8751659989356995

$ python most_similar.py face | column -t
head      0.8484194874763489
bed       0.8150191307067871
feet      0.7782107591629028
arm       0.7758032083511353
foot      0.7634758353233337
eyes      0.7624323964118958
mule      0.7610143423080444
shoulder  0.759937584400177
fury      0.7592475414276123
neck      0.7585527896881104

$ python most_similar.py book | column -t
chronicles    0.8260485529899597
gospel        0.7329334616661072
law           0.7292922139167786
acts          0.7145913243293762
kings         0.7094066739082336
lamentations  0.7042030096054077
service       0.6916581988334656
sect          0.6836168766021729
written       0.6775466799736023
presence      0.674580991268158
```

Let’s ask our 100-D model about cat-mouse-table relationship.

```
$ python calculate_distances.py | column -t
cat    mouse  0.2123
cat    table  0.3126
mouse  table  0.4248
```

Latent-Space Navigation

So, we have a “latent” space now.

> Latent space refers to an abstract multi-dimensional space containing feature values that we cannot interpret directly, but which encodes a meaningful internal representation of externally observed events. – What is a latent space?

> navigating the latent space (HN)

Let’s do some surrealist word arithmetic in our latent space.

```
$ python arithmetic.py | column -t

KING+WOMEN-MAN
queen           0.6779043674468994
captain         0.6737080216407776
prophet         0.6566919684410095

WAR-BLOOD
139:20          0.5924164652824402
kadeshbarnea    0.566044270992279
scuffling       0.5186205506324768

SHIP+MOVE
sail            0.9151894450187683
close           0.878623902797699
characters      0.8774296641349792

CITY+PEOPLE
congregation    0.7869210243225098
camp            0.778090238571167
```

NOTE: these embeddings are dependent on your input. What is similar to what is learned directly from the data.

> Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings,

> e.g. “We show that even word embeddings trained on Google News articles exhibit female/male gender stereotypes to a disturbing extent.” (2016)

Lot of research into this area alone. But how would a solution look like?

> Large language models propagate race-based medicine

And more.

> One key ethical consideration is representation. It is essential to ensure that the training data used to develop generative AI models are representative of the diverse range of perspectives, experiences, and backgrounds that exist within society ([99, 35, 94, 58, 100]). This helps to reduce the risk that biases are absorbed and propagated by models, leading to more equitable outcomes. **Transparency is another important aspect**. Developers should be transparent about the methodologies, data sources, and potential limitations of their generative AI models ([101, 102]). – Should ChatGPT be Biased? Challenges and Risks of Bias in Large Language Models (2023)

Following instructions

Fewer people cared about a pure “text” language model, but people immediately jumped onto the “instruct” type of models.

Co-creation

> Prompt engineering is iterative and interactive - a dialogue between humans and AI in an act of co-creation. – Prompting AI Art: An Investigation into the Creative Skill of Prompt Engineering

LLM prompting as retrieval problem

Imagine an infinite index, for which there is no “built-in retrieval model” - how do we resurrect the document we want?

Endless Text (XXX: rerun calculation)

Human text production back of the envelope calculation (since 1440, so 584 years).

- 1B writers writing 1 page per day every day for 584 years, 1 page contains 250 words, average word length of 5
- 1250 characters (bytes) * 1B * 365 * 584 = 266450000000000000 bytes of human text production = about 236 petabyte of text
- text can be compressed well, say to 20% of the size: 47,3 PB to store = need 2956 16T disks, one 16T disk costs about $300 = $868,600

How many tokens can LLMs produce?

- on off-the-shelf consumer hardware, it is possible to generate coherent text about 25 words/s, or 10s for a single page. That's 86400x times the output of a human
- on a single machine (with an RTX 4000 SFF, 70W GPU) I can generate the amount of text equal to the amount of text all humans have produced in 2,4 days

- NVIDIA sold 550000 H100 GPUs in 2023 (maybe 20x more powerful than the RTX 4000 SFF ADA)
- with all 550000 H100 GPUs produced in 2023 we can generate the equivalent of human text output from the past 584 years in 0.018s

Prompt Engineering

- the wikipedia article about Prompt Engineering is not that old, it first appeared in 2021-10-20, a Wednesday

- I first used tweet 1599971348717051904 as a joke on 2022-12-12 during a intro to programming CS course

> […] I am going to assert that Riley is the first Staff Prompt Engineer hired *anywhere*. […]

- as of 2024-02-05, IA Scholar finds 2066, Google Scholar about 10100 publications about "prompt engineering"
- let's compress some of these learnings into the next 20 minutes

> However, the open-ended nature of text as interaction is double-edged; while users can input anything and have access to an infnite range of generations, they also must engage in brute-force trial and error with the text prompt when the result quality is poor. – Design Guidelines for Prompt Engineering Text-to-Image Generative Models

Other definitions:

> A prompt is an input, a text command or a question provided to an AI model, to generate desired output like content or answer. The process of crafting effective and efficient prompts is called prompt design or prompt engineering. – Azure ML docs

Where does PROMPT come from?

- we use the term "PROMPT" ("Eingabeaufforderung") in computing for a long time, at least since 1973, but surely for quite a bit longer (in operatings systems, games; the original ELIZA paper from 1966 does not mention "prompt")
- what a language model does is to find a probable continuation of some input, e.g. prefix or prompt
- language models go back to 1980's, and the statistical nature of text has been researched since at least 1913 (Andrey Markov), and later, in 1951 by Claude Shannon
- in 2024 there are "marketplaces" for "prompts" – 'care about some "Stoic Wisdom For Modern Challenges" for $3.99 or "Social Media Strategy For Companies" for $2.99?

To illustrate, let's train a model and use to generate some text in 60 seconds. XXX: king-james-bible + X

> Prompts are hyperparameters. Every time you test a different prompt on your data, you become less sure if the LLM actually generalizes to unseen data. – https://huggingface.co/posts/MoritzLaurer/918404500694779

Prompt Engineering

> Prompt engineering is the means by which *LLMs are programmed* via prompts.

LLM

An LLM is so large, it learns patters, so that we do not need to encode every NLP task seperately; there is an "instruct" style training (XXX: how), so it can use text ("prompt") to steer output towards *some* direction.

> "Language Models are Few-Shot Learners", 2020, Cited by 20138

Open Models

Here's a task for Mistral (September 27, 2023):

how can conversational AI be useful for a student, who is writing an essay or in general?

> Conversational AI can be beneficial for students when they are writing essays or working on academic projects in various ways:

> 1. Providing instant feedback and suggestions: Conversational AI can help students identify grammatical errors, suggest synonyms, and even provide feedback on the clarity and coherence of their writing in real-time. This can save students time and effort in manually proofreading their work.

> 2. Offering writing prompts and ideas: Conversational AI can provide students with a wide range of essay topics or writing prompts to choose from based on their interests and academic subjects. It can also suggest ideas for improving the structure, content, and style of their essays.

> 3. Improving language learning: For non-native English speakers, conversational AI can help improve their language skills by offering suggestions, corrections, and explanations for grammar rules, sentence structure, and pronunciation. This can make it easier for students to write effectively in English.

> 4. Encouraging creativity: Conversational AI can act as a writing partner or tutor, providing encouragement, ideas, and feedback to help students overcome writer's block and stay

> motivated while working on their essays. It can also suggest creative ways to approach different aspects of their writing projects.

> 5. Providing time-management assistance: Conversational AI can help students manage their time by setting deadlines, providing reminders for upcoming assignments or exams, and offering suggestions for organizing their workload and prioritizing tasks. This can help students stay focused and productive while working on their essays and other academic projects.

> Overall, conversational AI can provide students with valuable support, feedback, and resources to help them improve their writing skills and successfully complete their academic projects.

Conversations

> One key feature of LLMs is in-context learning [14], where the model is trained to generate text based on a given context or prompt. This enables LLMs to generate more coherent and contextually relevant responses, making them suitable for interactive and conversational applications. Reinforcement Learning from Human Feedback (RLHF) [25, 268] is another crucial aspect of LLMs. – A Survey on Evaluation of Large Language Models

A main way to interact with a language model.

> One common approach to interacting with LLMs is prompt engineering [26, 222 , 263], where users design and provide speciic prompt texts to guide LLMs in generating desired responses or completing specific tasks. This is widely adopted in existing evaluation eforts. People can also engage in question-and-answer interactions [83], where they pose questions to the model and receive answers, or engage in dialogue interactions, having natural language conversations with LLMs.

Patterns

> A software pattern provides a reusable solution to a recurring problem within a particular context [10]. – A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT

- persona pattern, "act as X"
- EmotionPrompt – "Large Language Models Understand and Can Be Enhanced by Emotional Stimuli"

> EmotionPrompt outperforms existing existing prompt engineering approaches such as CoT and APE in most cases. We also see that EmotionPrompt can be plugged into APE in Table 1, indicating that EmotionPrompt is highly extensible and compatible with existing prompt engineering methods.

More:

- analytic augmentation
- Chain-of-Thought Prompting Elicits Reasoning in Large Language Models
- RAG

> Retrieval-augmented generation (RAG) is a technique for enhancing the accuracy and reliability of generative AI models with facts fetched from external sources.

And:

> To capture knowledge in a more modular and interpretable way, we augment language model pre-training with a latent knowledge retriever, which allows the model to retrieve and attend over doc- uments from a large corpus such as Wikipedia, used during pre-training, fine-tuning and infer-ence. – REALM: Retrieval-Augmented Language Model Pre-Training

- Language Models Can Teach Themselves to Use Tools

> They also, paradoxically, struggle with basic functionality, such as arithmetic or factual lookup, where much simpler and smaller models excel.