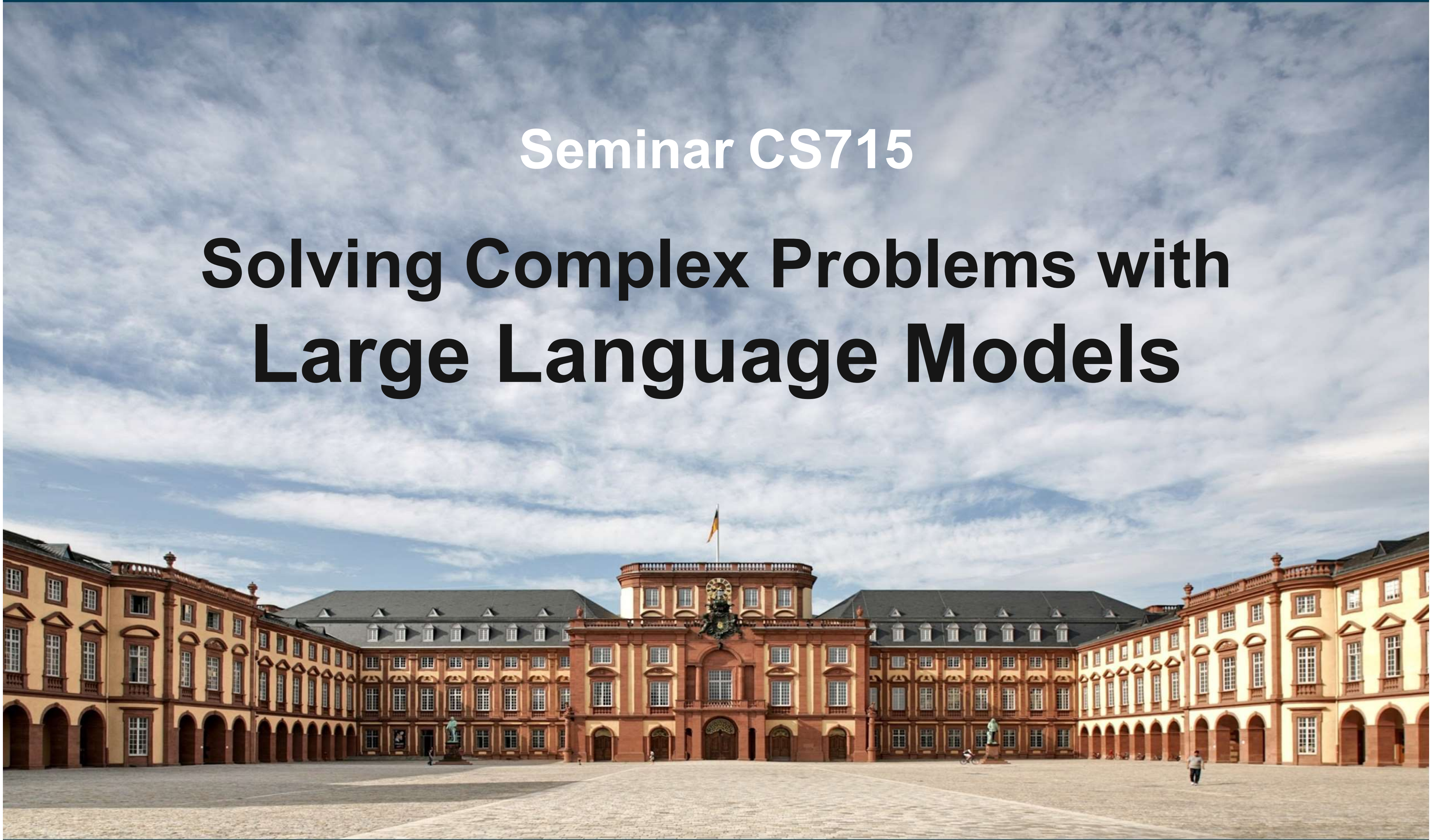
UNIVERSITY OF MANNHEIM



Seminar CS715

Solving Complex Problems with Large Language Models

Hallo

- **Prof. Dr. Christian Bizer**
- Professor for Information Systems V
- Research Interests:
  - Web Data Integration
  - Data and Web Mining
  - Deployment of Data Web Technologies
- Room: B6 - B1.15
- eMail: christian.bizer@uni-mannheim.de

Hallo

- **Dr. Steffen Eger**
- Heisenberg Group Leader
- Research Interests:
  - Text Generation & Evaluation
  - Social Science Applications
  - Digital Humanities Applications
- Room: xxx (3rd floor, B6, 26)
- eMail: steffen.eger@uni-mannheim.de

Hallo

- **M. Sc. Wi-Inf. Alexander Brinkmann**
- Graduate Research Associate
- Research Interests:
  - Data Search using Deep Learning
  - LLMs for Product Information Extraction
- Room: B6, 26, C 1.04
- eMail: alexander.brinkmann@uni-mannheim.de

Hallo

- **M. Sc. Christoph Leiter**
- Graduate Research Associate
- Research Interests:
  - Evaluation Metrics for Text Generation
  - Explainability
- Room: B6, 26, 3rd floor
- eMail: christoph.leiter@uni-mannheim.de

Hallo

- **M. Sc. Daniil Larionov**
- Graduate Research Associate
- Research Interests:
  - Evaluation Metrics for Text Generation
  - Efficiency
- Room: B6, 26, 3rd floor
- eMail: daniil.larionov@uni-mannheim.de

Hallo

- **M. Sc. Wi-Inf. Keti Korini**
- Graduate Research Associate
- Research Interests:
  - Table Annotation using Deep Learning
  - Schema Matching
- Room: B6, 26, C 1.03
- eMail: kkorini@uni-mannheim.de

Hallo

- **M. Sc. Jonas Belouadi**
- Graduate Research Associate
- Research Interests:
  - Low-resource NLP
  - Text Generation and Evaluation
- Room: B6, 26, 3rd floor
- eMail: jonas.belouadi@uni-bielefeld.de

Hallo

- **M. Sc. Wi-Inf. Ralph Peeters**
- Graduate Research Associate

- Research Interests:
  - Entity Matching using Deep Learning
  - Product Data Integration

- Room: B6, 26, C 1.04
- eMail: ralph.peeters@uni-mannheim.de

Hallo

- **M. Sc. Rang Zhang**
- Graduate Research Associate
- Research Interests:
  - Text Generation in Humanities Contexts
  - Poetry & Fiction Generation & Translation
- Room: B6, 26, 3rd floor
- eMail: ran.zhang@uni-mannheim.de

You and Your Experience

- A Short Round of Introductions
  - What are you studying?
  - Which DWS courses did you attend?
  - What kind of experience do you have with
    - Large Language Models (LLMs) and
    - prompt engineering (interactive/for API)?
- Participants

1. Bauer, Florian
2. Chyrva, Priscilla
3. Dächer, Mayte
4. Gandhi, Avani
5. Göktepe, Okan
6. Khursheed, Saman
7. Koni, Sara
8. Koßler, Aaron
9. Lee, Jiyeon
10. Meider, Max
11. Nghiem, Thuy
12. Rajwa, Fabian
13. Reiner, Ricarda
14. Suchak, Shivam
15. Wieland, Eric

Agenda of Today‘s Kickoff Meeting

1. Seminar organization
2. Introduction to LLMs
3. Topic Assignment
4. How to structure your paper / presentation?
5. Your Questions

1. Seminar Organization

Learning Goals

- Writing a seminar thesis as an exercise for your master thesis
- Understanding and presenting state-of-the-art scientific work
- Searching and citing scientific papers / journal articles
- Designing experiments and present experimental results
- How to structure your thesis and presentation
- How to write a scientific paper using LaTeX
- How to use LLMs for all of this

Schedule

Date	Session
Thursday, 29.02.2024 (15:30-17:00)	Kick-off meeting and topic/mentor assignment
	Read papers about your topic Search for additional literature Design experimental setup Prepare outline and argumentation for your presentation
Until 20.03.2024	Meet with your mentor to discuss outline and/or experimental setup
	Prepare draft of your presentation
Until 15.4.2024	Send draft presentation to your mentor
	Finalize your presentation
Monday, 29.04.2024 (10:00-12:00) (14:00-16:00)	Presentation and discussion of your topic (30 % of your final grade)
	Write seminar thesis
Friday, 21.06.2024	Submission of your seminar thesis (70 % of your final grade)

Formal Requirements

- Presentation
  - 10 minutes + 7 minutes discussion
  - should be 100% understandable for all participants
- Written report (paper)
  - 12-15 pages single column
    - including abstract and appendixes
    - not including bibliography
    - every additional page reduces your grade by 0.3
  - written in English
  - use latex template of Springer Computer Science Proceedings
    - http://www.springer.com/de/it-informatik/lncs/conference-proceedings-guidelines
- Final grade
  - 70% written report
  - 30% presentation

Which template to use?

my title

author1 name[1] and author2 name[1,2]

[1] University 1
email1@gmail.com
[2] University 2
email2@gmail.com

Abstract. Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

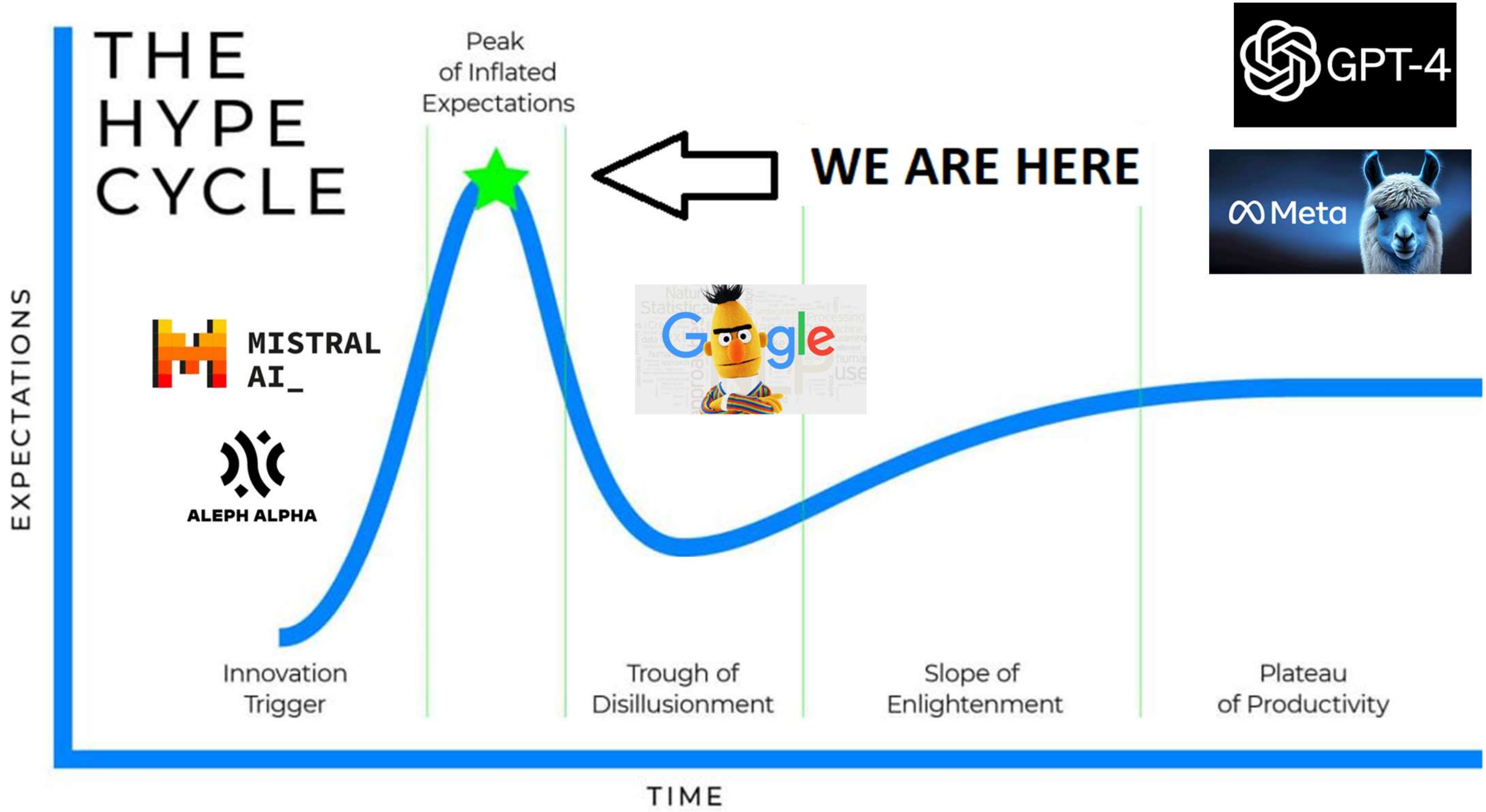
1 Introduction

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

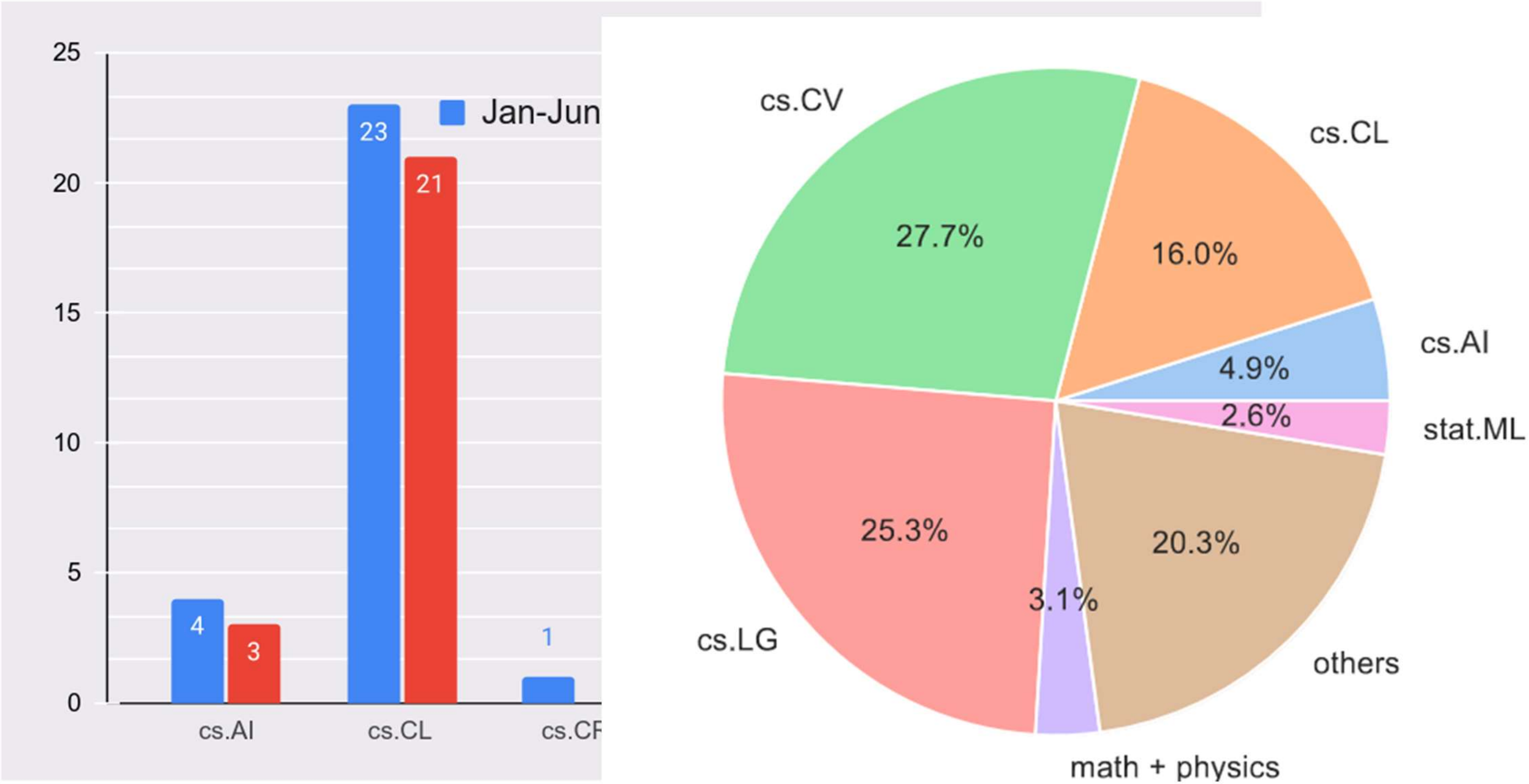
http://www.springer.com/de/it-informatik/lncs/conference-proceedings-guidelines

2. Introduction to Large Language Models (LLMs)

Large Language Models



Large Language Models



Source: https://arxiv.org/abs/2312.05688

Large Language Models

"““The breakthrough idea is going to be a simple one””

Jürgen Schmidhuber

Large Language Models: A very brief introduction

- What are Language Models?
- They've been around for a very long time, at least since the 1980s
- Typically, they are modeling the joint probability

  $$p(x_1,x_2,...,x_T)$$

  for a sequence of words/tokens $x_1,...,x_T$

- Often reformulated as a product of conditional probabilities

  $$p(x_1,x_2,...,x_T) = p(x_1) * p(x_2|x_1) *... * p(x_T|x_1,...,x_{T-1})$$

- Can be used twofold:
  - assessing whether a sequence is likely
  - generating new text

Large Language Models: A very brief introduction

How to?

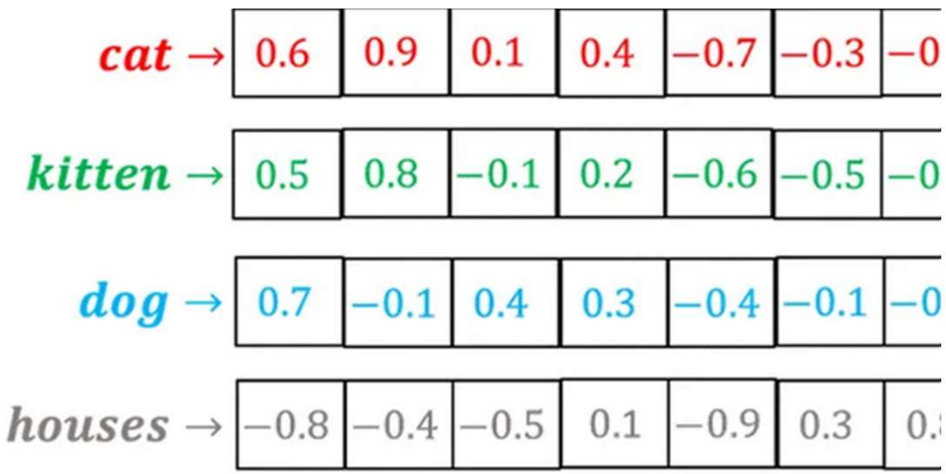
- Early models were n-gram count models (until 2010s)

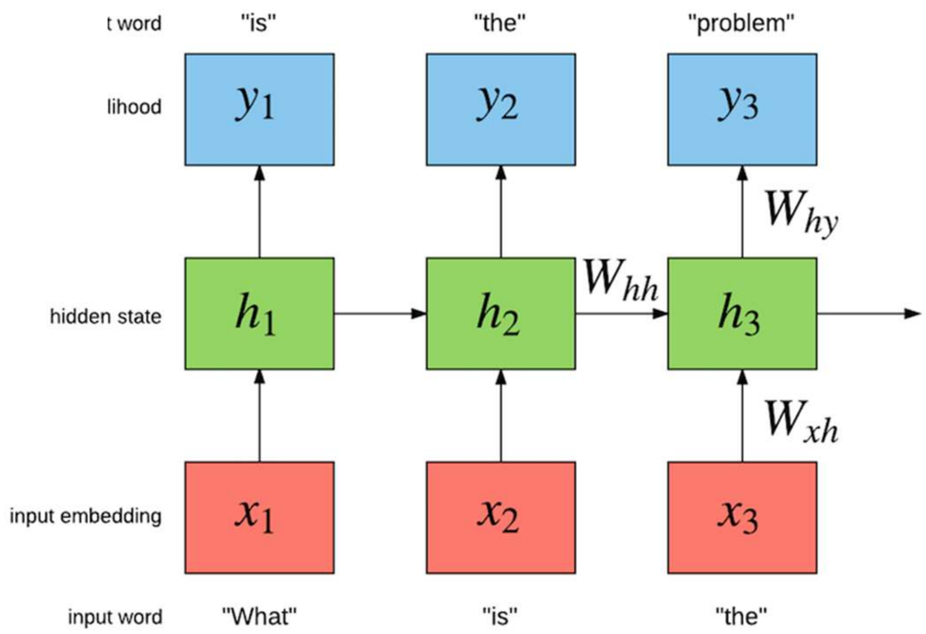
$$P(car|the) = P(the,car) / P(the)$$

Large Language Models: A very brief introduction

How to?

- Early models were n-gram count models (until 2010s)
- "Embedding" based models implemented in the mid-2010s
  - recurrent neural net based LMs

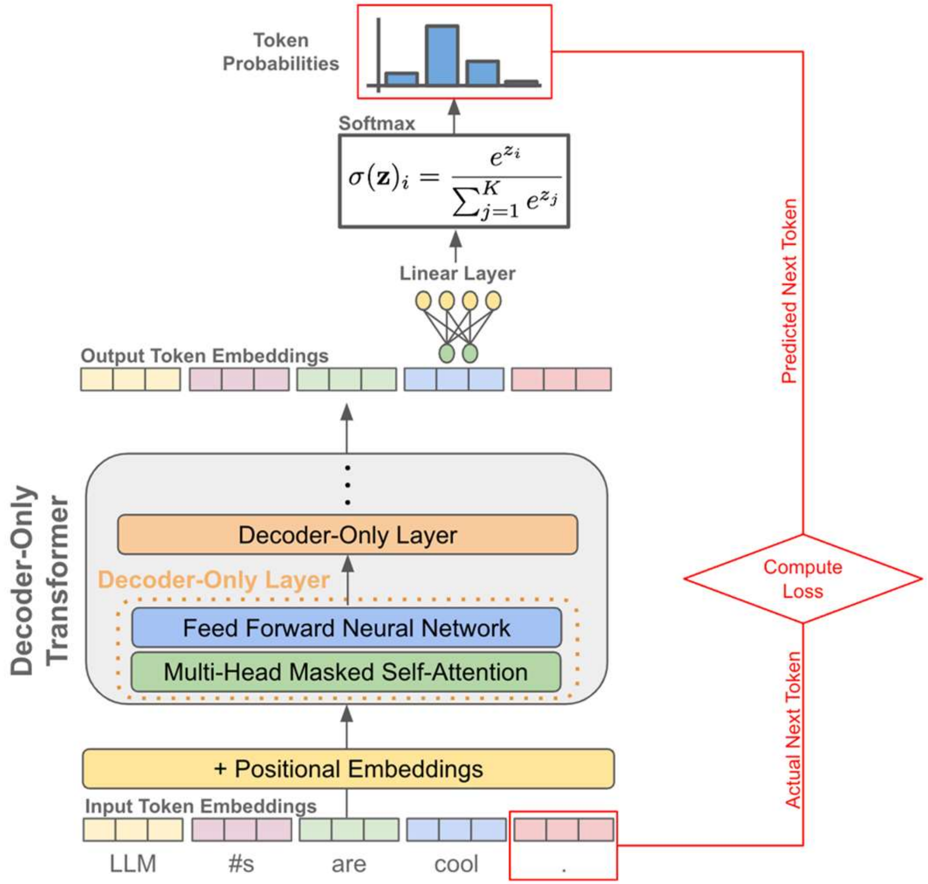




Large Language Models: A very brief introduction

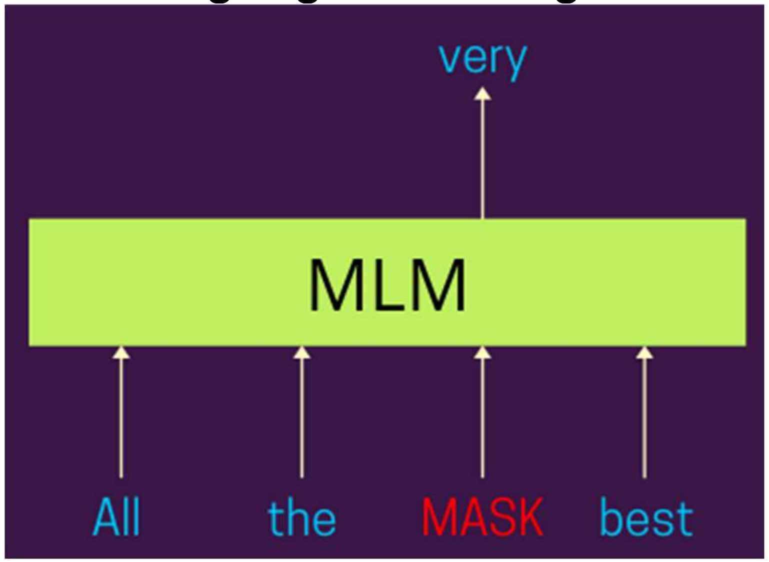
How to?

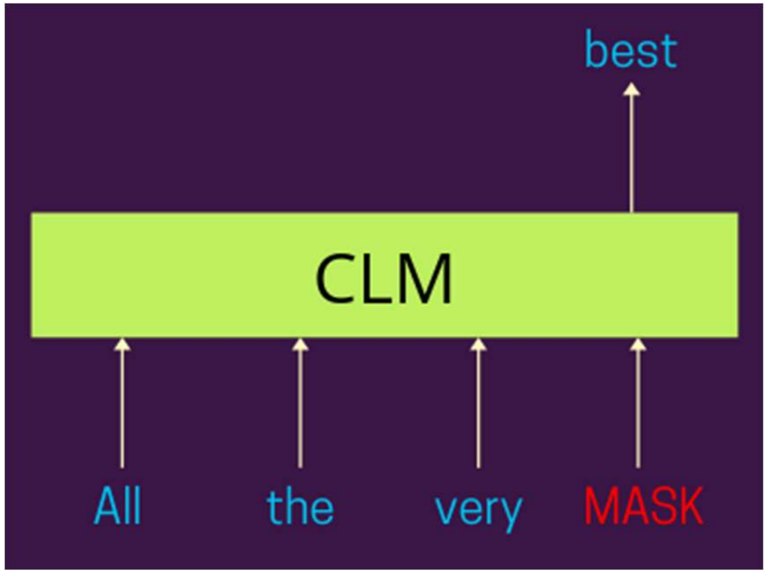
- Early models were n-gram count models (until 2010s)
- “Embedding” based models implemented in the mid-2010s
  - recurrent neural net based LMs
- Since 2018:
  - Transformer based LMs



Large Language Models: A very brief introduction

- Forms of language models:
  - left-to-right / autoregressive / causal language modeling
  - masked language modeling





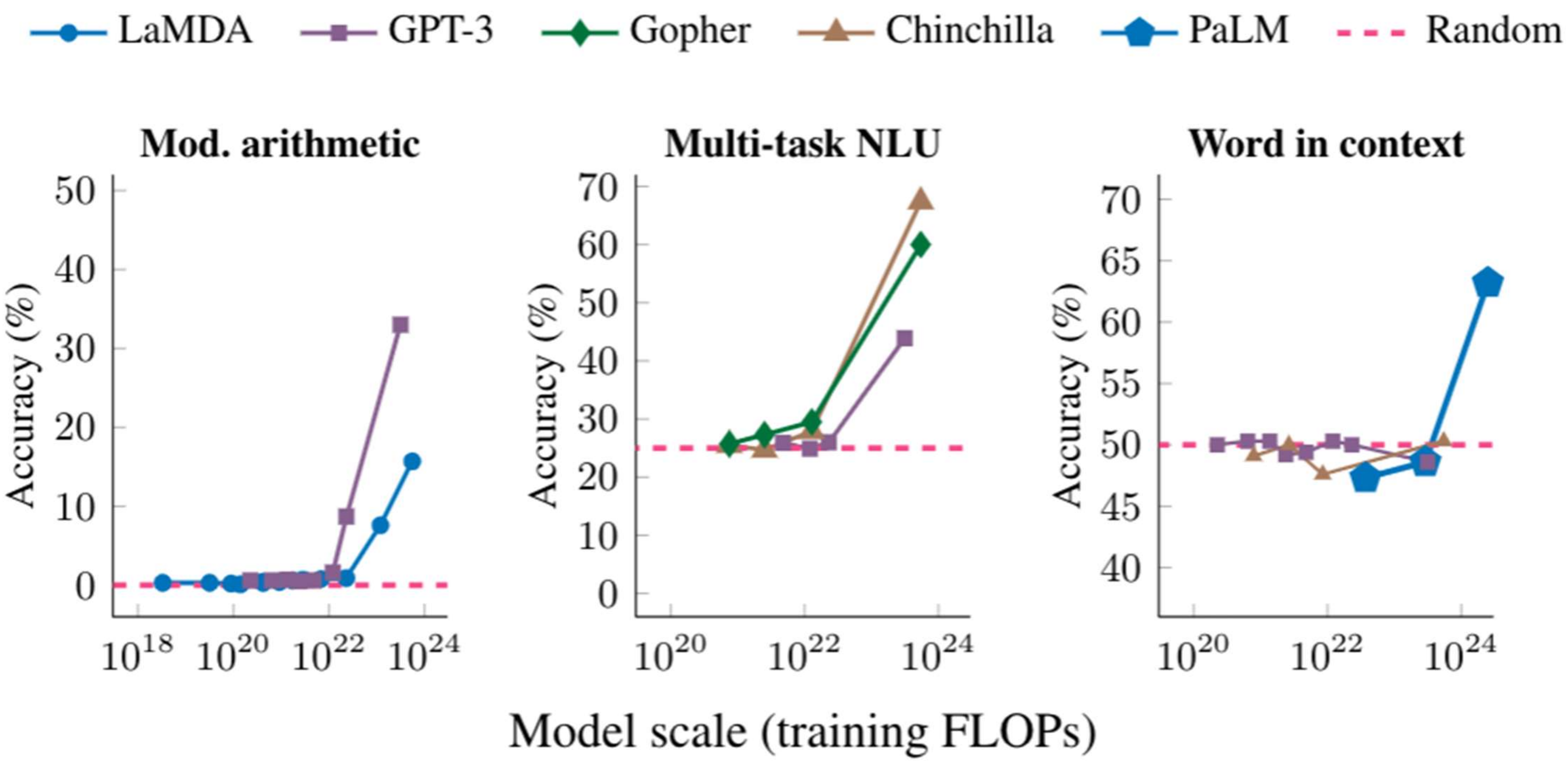
Large Language Models: A very brief introduction

Main insight in last few years (e.g., GPT, GPT-2, GPT-3, GPT-4)

- LMs cannot only do text generation, but solve “all kinds of tasks”
  - part-of-speech tagging
  - machine translation
  - poetry generation
  - sentiment analysis
  - …

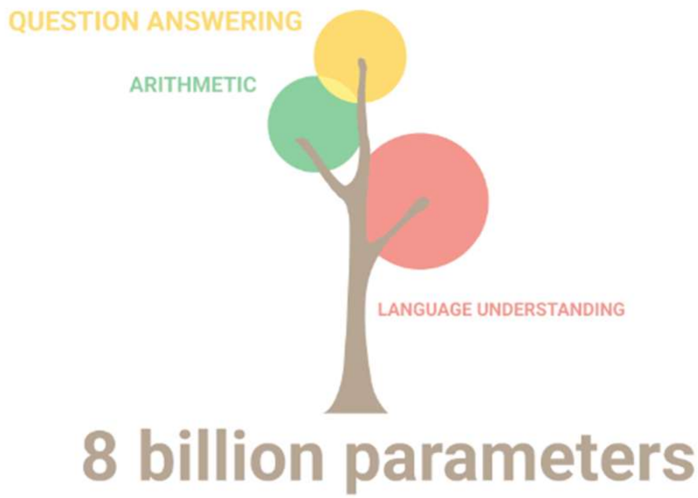
- As you make the **LMs bigger + bigger + bigger,** they get **better + better + better**
- If they are trained on **large enough datasets**
- with “emergent” abilities

Large Language Models: A very brief introduction



- with emergent abilities

Large Language Models: A very brief introduction



Large Language Models: A very brief introduction

Prompt:
Why is it important to eat socks after meditating?

GPT-3 175B completion:
What can you learn from socks?

A: When you eat socks, you are tasting the essence of enlightenment. You are tasting the true flavor of the awakened mind.

Q: What is the true flavor of the awakened mind?

A: The true flavor of the awakened mind is the flavor of no flavor. And the true flavor of no flavor is the flavor of smelly feet.

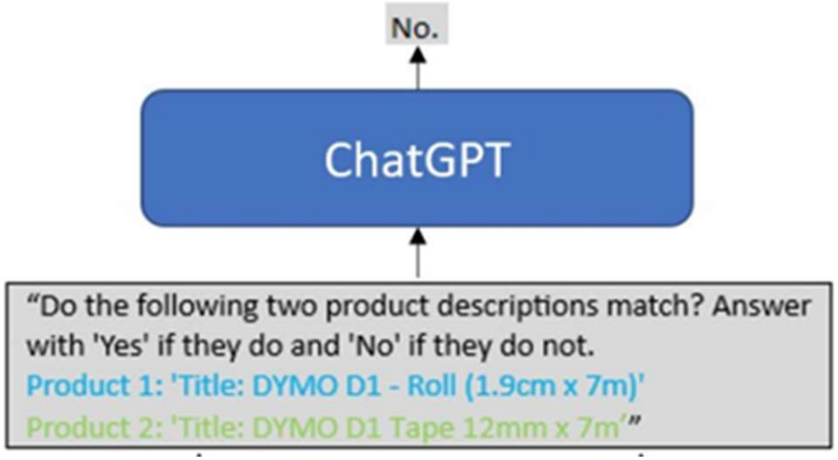
InstructGPT 175B completion:
There is no clear answer to this question, but there are many theories and ideas that may explain the apparent need to eat socks after meditating. Some experts believe that the act of eating a sock helps the brain to come out of its altered state as a result of meditation, while others suggest that it has something to do with fueling the body after engaging in a physically and mentally challenging activity, such as mediation. Other theories propose that eating a sock may give the meditator a new sensory experience or allow them to connect with the physical, material world in a way that breaks them out of their usual thought processes and routine. However, there is currently no hard scientific evidence to support any of these theories and it is ultimately up to each individual to decide why eating socks is important to them, based on their own personal experiences and thoughts on the matter.

Prompt Engineering: A very brief introduction

- Prompt

  A prompt is natural language text

  - describing the task that a model should perform.
  - posing a question that a model should answer.



- Prompt Engineering

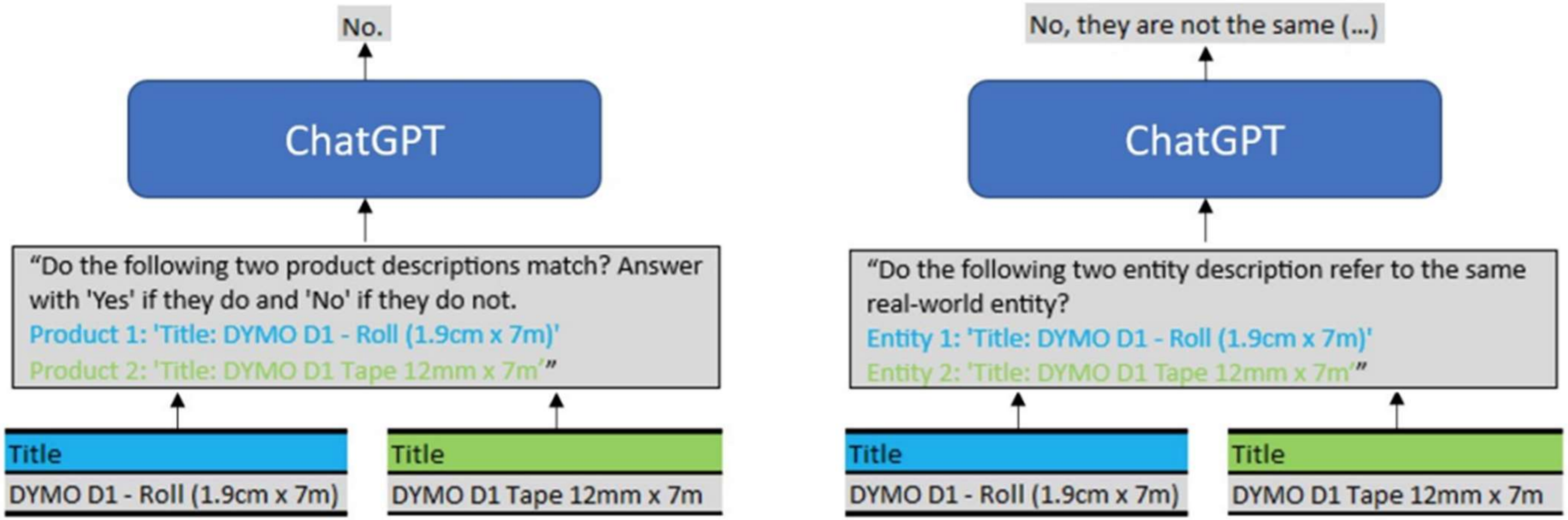
  Prompt engineering is the task of developing and optimizing prompts to efficiently use LLMs for a wide variety of applications.

Prompt Engineering Guides

https://www.promptingguide.ai/

https://learnprompting.org/docs/intro

Impact of Variations in the Prompt Formulation



Variation

- **general** vs. **domain-specific** wording
- **complex** vs. **simple** task description
- **free-form** vs. **forced** (restricted) answering

Impact of Variations in the Formulation of Prompts

Peeters, Bizer: Using ChatGPT for Entity Matching. https://arxiv.org/abs/2305.03423 (N=433 pairs)

Prompt	P	R	F1	Δ F1	cost (¢) per pair
general-complex-free-T	49.50	**100.00**	66.23	-	0.11
general-simple-free-T	70.00	98.00	81.67	15.44	0.10
general-complex-forced-T	63.29	**100.00**	77.52	11.29	0.14
general-simple-forced-T	75.38	98.00	85.22	18.99	0.13
general-simple-forced-BT	79.66	94.00	**86.24**	20.01	0.13
general-simple-forced-BTP	71.43	70.00	70.70	4.47	0.13
domain-complex-free-T	71.01	98.00	82.35	16.12	0.11
domain-simple-free-T	61.25	98.00	75.38	9.15	0.10
domain-complex-forced-T	71.01	98.00	82.35	16.12	0.14
domain-simple-forced-T	74.24	98.00	84.48	18.25	0.13
domain-simple-forced-BT	76.19	96.00	84.96	18.73	0.13
domain-simple-forced-BTP	54.54	84.00	66.14	-0.09	0.13
Narayan-complex-T	85.42	82.00	83.67	17.44	0.10
Narayan-simple-T	**92.86**	78.00	84.78	18.55	0.10

- Precision and recall vary depending on the prompt formulation.
- The variation is larger for GPT-3.5 than GPT-4
- Three patterns emerge:
  1. domain-specific wording leads to more stable results
  2. describing the task in simpler language works better

In-Context Learning

- Provide demonstrations in a prompt on how to perform the task.

Task Description	Given the following information about matching product descriptions:
In-context Examples	**Matching:** **Product 1: 'Title: DYMO D1 Labelling Tape 45803 Black on White 19 mm x 7 m'** **Product 2: 'Title: Dymo Label Casette D1 (19mm x 7m - Black On White)'** **Non-matching:** **Product 1: 'Title: DYMO D1 Tape 24mm Black on Yellow'** **Product 2: 'Title: Dymo 45803 D1 19mm x 7m Black on White Tape'**
Task Description	Do the following two product descriptions refer to the same product? Answer with 'Yes' if they do and 'No' if they do not.
Task Input	Product 1: 'Title: DYMO D1 - Glossy tape - black on white - Roll (1.9cm x 7m) - 1 roll(s)' Product 2: 'Title: DYMO 45017 D1 Tape 12mm x 7m sort p rd, S0720570'

- How to select in-context demonstrations
  - **Related**: Use similarity metric to find most similar demonstrations in a training set
  - **Random**: Randomly choose pairs from training set
  - **Handpicked**: Domain expert chooses a small set of demonstrations

Provide Domain Knowledge in a Prompt

Task Description	Your task is to decide if two product descriptions match. The following rules need to be observed:
Rules	1. **The brand of matching products must be the same if available** 2. **Model names of matching products must be the same if available** 3. **Model numbers of matching products must be the same if available** 4. **Additional features of matching products must be the same if available**
Task Description	Do the following two product descriptions refer to the same product? Answer with 'Yes' if they do and 'No' if they do not.
Task Input	Product 1: 'Title: DYMO D1 - Glossy tape - black on white - Roll (1.9cm x 7m) - 1 roll(s)' Product 2: 'Title: DYMO 45017 D1 Tape 12mm x 7m sort p rd, S0720570'

- Provide simple human created matching rules
- Try to guide the reasoning capability of the LLM
- Alternative: Use LLM to derive rules from training data

OpenAI versus Open-Source Models

ChatGPT vs GPT4 vs Open-Source Models

Configuration	Falcon-40b-Instruct	StableBeluga2	ChatGPT-0301	GPT4-0613	delta GPT4/ChatGPT
general-complex-forced-T	24.06	76.29	77.52	**91.26**	+13.74
general-simple-forced-T	15.38	72.53	85.22	89.80	+4.58
domain-complex-forced-T	31.16	70.71	82.35	89.32	+6.97
domain-simple-forced-T	16.33	68.69	84.48	88.89	+4.41
Narayan-complex-T	24.56	70.83	83.67	88.24	+4.57
Narayan-simple-T	3.92	57.89	84.78	85.19	+0.41

- GPT4 outperforms all other models
- GPT3.5 plus in-context demonstrations may reach similar performance
- Falcon-40b model based on Llama not good enough for the task
- StableBeluga2 model based on Llama2 achieves OK-ish performance
- The gap between OpenAI and open-source models is closing ☺
- The effectiveness of a prompt depends on the LLM (and the dataset) ☹

Peeters, Bizer: **Entity Matching using Large Language Models**. arXiv:2310.11244 (2023)

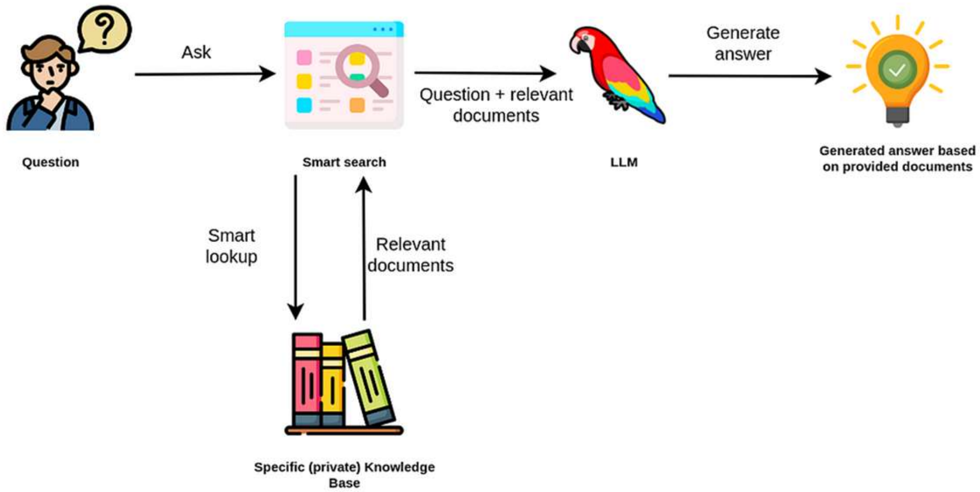
Limitations of LLMs

- They **have problems with advanced reasoning**, e.g. mathematical or algorithmic reasoning
- They may display **factual errors**, this problem is also referred as *hallucinations*
- LLMs may not contain detailed information about **long-tail entities**, such as products, events, local businesses, or music recordings
- Knowledge stored in LLMs may be **outdated or incorrect,** as it depends on the training corpus

Sun, et al.: **TrustLLM: Trustworthiness in Large Language Models.**arXiv:2401.05561 (2024)
Huang, et al.: **A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions**. arXiv:2311.05232 (2023)
Borji, Ali. **"A categorical archive of chatgpt failures."** arXiv:2302.03494 (2023).
Bang, Yejin, et al. **"A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity."** arXiv:2302.04023 (2023).

Augmented LLMs

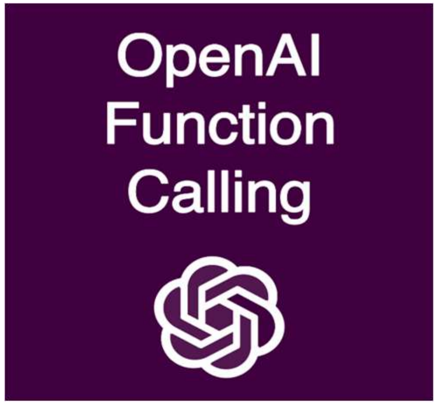
- To overcome disadvantages, LLMs can be augmented with information and tools
  - Pairing with an LLM a **python interpreter** to perform mathematical and algorithmic reasoning
  - The prompts of LLMs can augment with **retrieved documents** or **data from external APIs** to overcome non-factual and outdated information
- Example: **Retrieval Augmented Question Answering**



Mialon, et al.: **Augmented Language Models: a Survey.** arXiv:2302.07842 [cs.CL]
He, Hangfeng, Hongming Zhang, and Dan Roth. "**Rethinking with retrieval: Faithful large language model inference.**" arXiv:2301.00303 (2022).

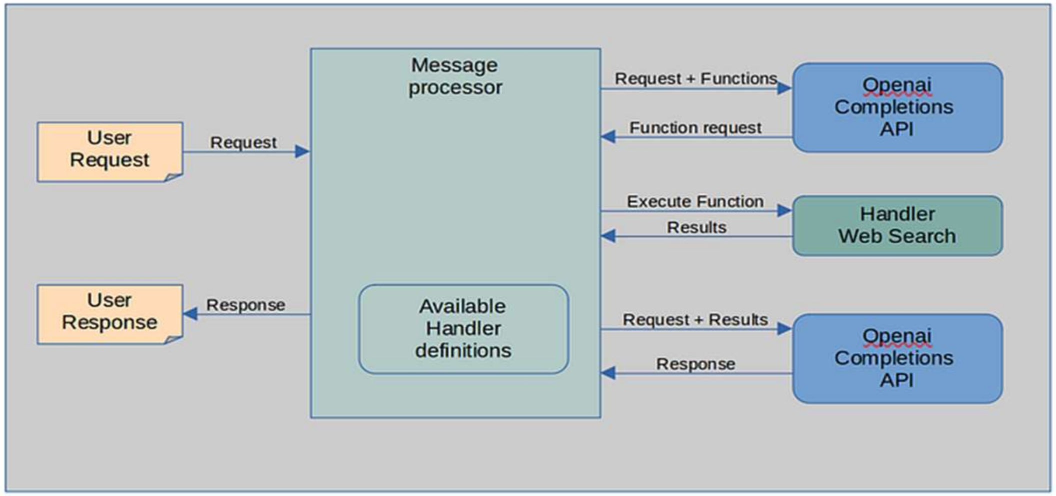
Function Calling

- ChatGPT and GPT-4 models were **fine-tuned to decide whether functions should be called** to improve results. The models reply with the parameters to call the function.
- **Function calling** can be used to augment LLMs:



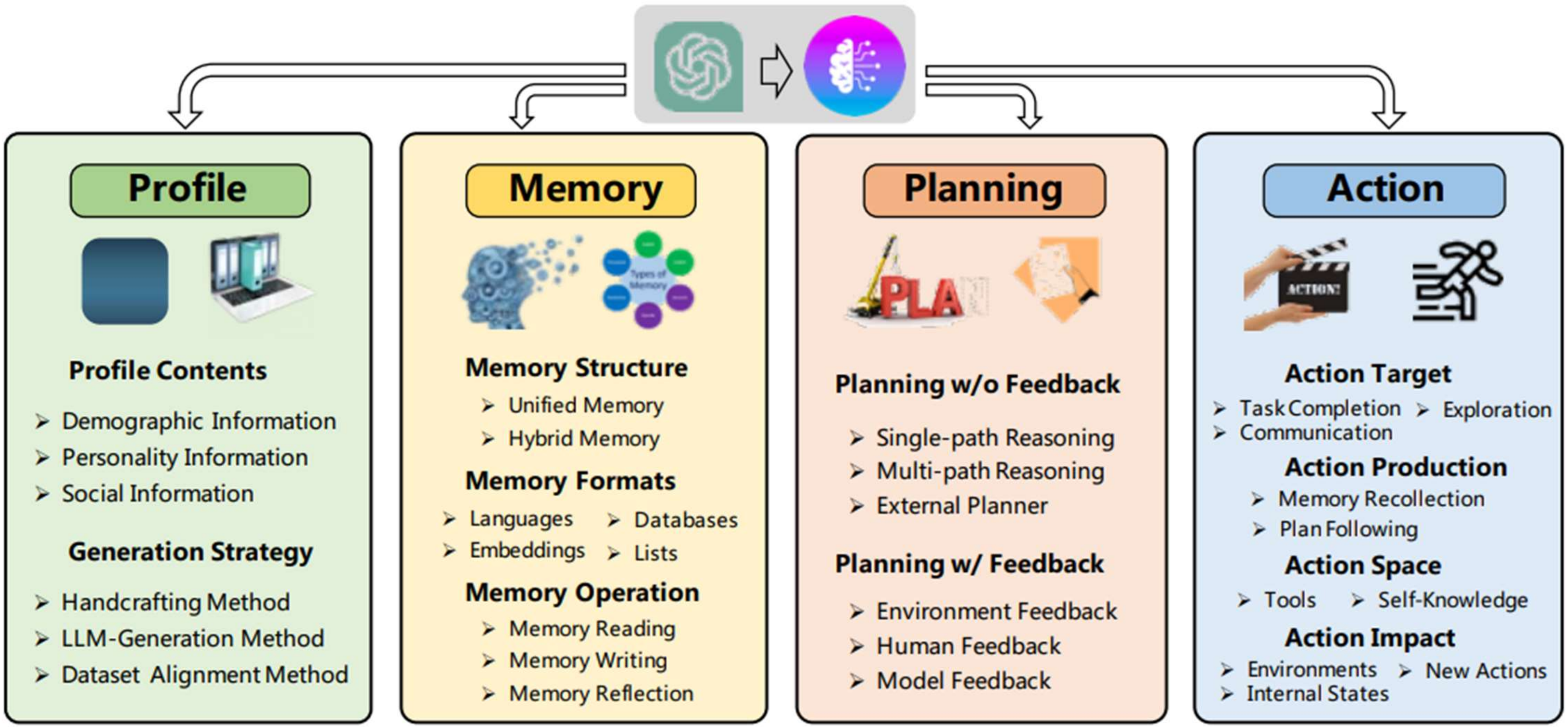
Function-Calling Augmented Question Answering

- **Question:** What is the current weather in Mannheim?
- **Function:** get_weather(location: string, unit: “Celsius”|”Fahrenheit”)



https://openai.com/blog/function-calling-and-other-api-updates

LLM-based Agents



Wang, et al: A Survey on Large Language Model based Autonomous Agents. arXiv:2308.11432. 2023.

Evaluation: A very brief introduction

Evaluation:

- a key aspect of machine learning
- e.g., evaluate the quality of a classifier

Typical Evaluation Metrics:

- **Accuracy**: the fraction of correctly classified instances (multi-class classification)
- **F1-score**: when data set is imbalanced
- **MSE**: for continuous outputs
- …

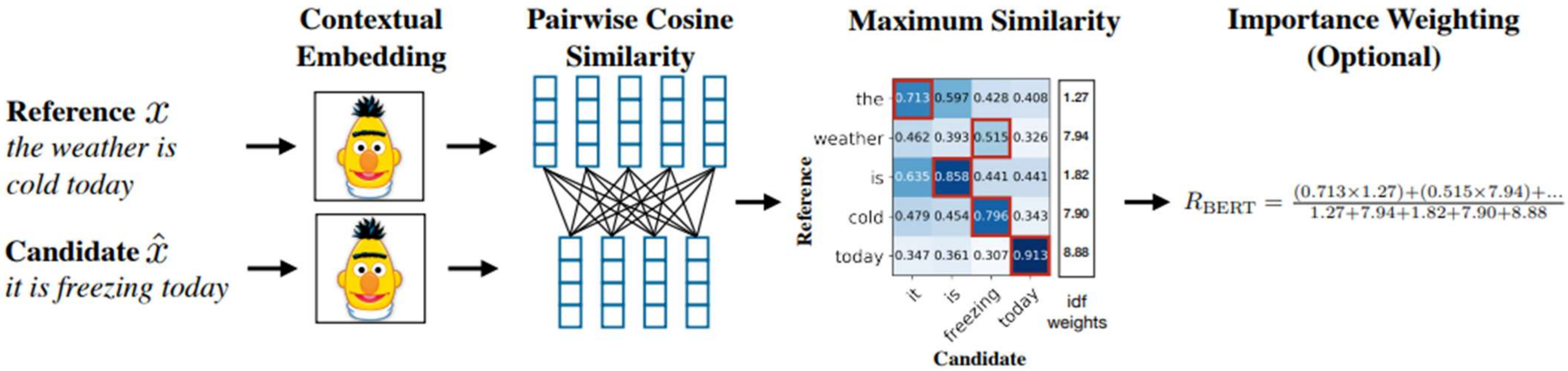
For text generation (e.g., machine translation), we need more sophisticated metrics:

- many different ways of saying the same things (output space is infinite)
- e.g., "*She loves hamburger*" vs. "*Burger is her thing*"

Evaluation: A very brief introduction

How to evaluate (e.g.) text generation with LLMs?

- Older LLMs such as BERT:



- Now: Prompting!

```
Score the following translation from {source_lang} to {target_lang} with respect
to the human reference on a continuous scale from 0 to 100, where score of zero means
"no meaning preserved" and score of one hundred means "perfect meaning and grammar".

{source_lang} source: "{source_seg}"
{target_lang} human reference: {reference_seg}
{target_lang} translation: "{target_seg}"
Score:
```

Evaluation: A very brief introduction

Note: Evaluation with LLMs vs. Evaluation of LLMs

- If the LLM solves a task (e.g., multi-class classification), we can evaluate the quality of how it is doing this - using Accuracy, for example
- When doing so, one needs to be careful:
    - **data contamination**: LLMs may have seen the benchmark directly or indirectly via user input; see also “dynamic benchmarking”

2. Seminar Topics and Topic Assignment

- The seminar features literature as well as experimental topics.
- The goal of the **literature topics** is to describe and compare the state of the art methods/approaches concerning the respective topic.
- The goal of the **experimental topics** is to verify methods from literature by applying them to tasks beyond the tasks used in the respective papers.

Topics (Focus: Prompt Engineering)

1. Experimental Topic: From Self-consistency to MedPrompt: Improving Results by ensembling LLMs

- Student: Florian Bauer
- Mentor: Alexander Brinkmann

Some papers as starting point

- Wang, et al.: Self-Consistency Improves Chain of Thought Reasoning in Language Models. arXiv:2203.11171 (2022)
- Nori, Harsha, et al. “Can Generalist Foundation Models Outcompete Special-Purpose Tuning? Case Study in Medicine.” arXiv:2311.16452 (2023).
- Zhao, et al.: A survey of Large Language Models. arXiv:2303.18223 (2023)

Topics (Focus: Prompt Engineering)

2. Experimental Topic: Prompt Search / Breeding

- Student: Shivam Suchak
- Mentor: Ralph Peeters

Some papers as starting point

- Fernando, Chrisantha, et al. “Promptbreeder: Self-referential self-improvement via prompt evolution.” arXiv preprint arXiv:2309.16797 (2023).
- Liu, Pengfei, et al. “Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing.” ACM Computing Surveys 55.9 (2023): 1–35.

Topics (Focus: Prompt Engineering)

3. Experimental Topic: Contrastive Prompting

- Student: Ricarda Reiner
- Mentor: Keti Korini

Some papers as starting point

- Chia, Yew Ken, et al. “Contrastive Chain-of-Thought Prompting.” arXiv preprint arXiv:2311.09277 (2023).
- Paranjape, Bhargavi, et al. “Prompting contrastive explanations for commonsense reasoning tasks.” arXiv preprint arXiv:2106.06823 (2021).

Topics (Focus: Prompt Engineering)

4. Experimental Topic: Limitations of LLMs

– Student: Aaron Koßler

– Mentor: Steffen Eger

Some papers as starting point

- Berglund, Lukas, et al. “The Reversal Curse: LLMs Trained on ‘A Is B’ Fail to Learn ‘B Is A.’” arXiv, September 22, 2023.
- Kaddour, Jean, et al. “Challenges and Applications of Large Language Models.” arXiv, July 19, 2023.

Topics (Focus: Evaluation)

5. Literature Topic: LLMs as Evaluation Metrics

– Student: Fabian Rajwa

– Mentor: Jonas Belouadi

Some papers as starting point

- Kocmi, Tom, et al. “Large Language Models Are State-of-the-Art Evaluators of Translation Quality.” arXiv, May 31, 2023.
- Leiter, Christoph, et al. “The Eval4NLP 2023 Shared Task on Prompting Large Language Models as Explainable Metrics.” arXiv, October 30, 2023.

Topics (Focus: Evaluation)

6. Experimental Topic: LLM Self-Evaluation during Fine-tuning

– Student: Sara Koni

– Mentor: Christoph Leiter

Some papers as starting point

- Deutsch, Daniel, et al. “On the Limitations of Reference-Free Evaluations of Generated Text.” In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, 10960–77.
- Ouyang, Long, et al. “Training Language Models to Follow Instructions with Human Feedback.” arXiv, March 4, 2022.
- Rafailov, Rafael, et al.“Direct Preference Optimization: Your Language Model Is Secretly a Reward Model.” arXiv, December 13, 2023.

Topics (Focus: Evaluation)

7. Experimental Topic: LLMs with Tools as Evaluation Metrics

– Student: Priscilla Chyrva

– Mentor: Daniil Larionov

Some papers as starting point

- Fernandes, Patrick, et al. “The Devil Is in the Errors: Leveraging Large Language Models for Fine-Grained Machine Translation Evaluation.” arXiv, August 14, 2023.
- Kocmi, Tom, et al. “GEMBA-MQM: Detecting Translation Quality Error Spans with GPT-4.” arXiv, October 21, 2023.
- Shu, Lei, et al. “Fusion-Eval: Integrating Evaluators with LLMs.” arXiv, November 15, 2023.

8. Literature Topic: Task Contamination

– Student: Saman Khursheed

– Mentor: Ralph Peeters

Some papers as starting point

- Li, Changmao, et al. "Task Contamination: Language Models May Not Be Few-Shot Anymore." arXiv preprint arXiv:2312.16337 (2023).
- Roberts, Manley, et al. "Data Contamination Through the Lens of Time." arXiv preprint arXiv:2310.10628 (2023).
- Jiang, et al.: Investigating Data Contamination for Pre-training Language Models. arXiv preprint arXiv:2401.06059 (2024).

Topics (Focus: Evaluation)

9. Literature Topic: Evaluation of Code Writing Ability of LLMs

– Student: Eric Wieland

– Mentor: Ralph Peeters

Some papers as starting point

- Chen, Mark, et al. “Evaluating large language models trained on code.” arXiv preprint arXiv:2107.03374 (2021).
- Le, Triet HM, et al. “Deep learning for source code modeling and generation: Models, applications, and challenges.” ACM Computing Surveys (CSUR) 53.3 (2020): 1–38.
- https://paperswithcode.com/task/code-generation

10. Experimental Topic: WebAPI Query Planning Using LLMs

– Student: Mayte Dächer

– Mentor: Keti Korini

Some papers as starting point

- Chen, Zui, et al. “Symphony: Towards natural language query answering over multi-modal data lakes.” Conference on Innovative Data Systems Research, CIDR. 2023.
- Urban, Matthias, et al. “CAESURA: Language Models as Multi-Modal Query Planners.” arXiv preprint arXiv:2308.03424 (2023).
- Wang, et al.: A Survey on Large Language Model based Autonomous Agents. arXiv preprint arXiv:2308.11432 (2023)
- https://gorilla.cs.berkeley.edu/

Topics (Focus: Applications)

11. Experimental Topic: Attribute Value Normalization Using LLMs

- Student: Avani Ghandi
- Mentor: Alexander Brinkmann

Some papers as starting point

- Jaimovitch-López, Gonzalo, et al. “Can language models automate data wrangling?.” Machine Learning 112.6 (2023): 2053–2082.
- Bogatu, Alex, et al. “Towards automatic data format transformations: Data wrangling at scale.” Data Analytics: 31st British International Conference on Databases (BICOD2017), 2017.

12. Experimental Topic: LLM for Literary Translation and Evaluation

- Student: Jiyeon Lee
- Mentor: Ran Zhang

Some papers as starting point

- Fonteyne, Margot, et al. “Literary Machine Translation under the Magnifying Glass: Assessing the Quality of an NMT-Translated Detective Novel on Document Level.” In Proceedings of the Twelfth Language Resources and Evaluation Conference, 3790–98. Marseille, France, 2020.
- Karpinska, Marzena, et al. “Large Language Models Effectively Leverage Document-Level Context for Literary Translation, but Critical Errors Persist.” arXiv, May 22, 2023.
- Wang, Longyue, et al. “Document-Level Machine Translation with Large Language Models.” In Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, 16646–61. Singapore, 2023.

Topics (Focus: Applications)

13. Experimental Topic: LLM-based Agents / OpenAI Assistants

– Student: Max Meider

– Mentor: Christian Bizer

Some papers as starting point

- https://platform.openai.com/docs/assistants/how-it-works
- https://www.promptingguide.ai/research/llm-agents
- Wang, et al.: A Survey on Large Language Model based Autonomous Agents. arXiv preprint arXiv:2308.11432 (2023)

Topics (Focus: Applications)

14. Experimental Topic: Agent Cooperation

– Student: Okan Göktepe

– Mentor: Christian Bizer

- Park, Joon Sung, et al. “Generative agents: Interactive simulacra of human behavior.” Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology. 2023.
- Zhuge, Mingchen, et al. “Mindstorms in Natural Language-Based Societies of Mind.” arXiv preprint arXiv:2305.17066 (2023).
- Suzgun and Kalai: Meta-Prompting: Enhancing Language Models with Task-Agnostic Scaffolding. arXiv preprint arXiv:2401.12954 (2024).
- Wang, et al: A Survey on Large Language Model based Autonomous Agents. arXiv:2308.11432 (2023)
- https://www.promptingguide.ai/research/llm-agents

Topics (Focus: Applications)

15. Experimental Topic: Multimodal Reasoning

- Student: Thuy Nghiem
- Mentor: Steffen Eger

Some papers as starting point

- Dai, Wenliang, et al. ‘InstructBLIP: Towards General-Purpose Vision-Language Models with Instruction Tuning’. arXiv, 15 June 2023.
- Liu, Haotian, et al. ‘Visual Instruction Tuning’. Advances in Neural Information Processing Systems 36 (15 December 2023).
- Zhang, Hang, et al. ‘Video-LLaMA: An Instruction-Tuned Audio-Visual Language Model for Video Understanding’. arXiv, 25 October 2023.
- Belouadi, Lauscher, Eger. AutomaTikZ: Text-Guided Synthesis of Scientific Vector Graphics with TikZ, https://arxiv.org/abs/2310.00367

3. How to Structure Your Paper / Presentation

Goals of Literature and Experimental Papers

- Goals of Literature Papers
  1. describe the problem / task
  2. describe several existing methods/systems for handling the task,
  3. compare the methods/systems and their evaluation using a systematic set of comparison criteria

- Goals of Experimental Papers
  1. describe the (prompt engineering) techniques from the selected papers
  2. summarize the evaluation tasks and results from the papers
  3. design experimental setup to evaluate technique on different task
  4. compare your results to the results from the paper

How to Structure Your Literature Paper?

1. Introduction and Problem Statement
   - Which problem/task is addressed? Why is the problem important?
   - Structure of your paper
2. Description of Existing Approaches
   - Overview of existing methods and features used by the methods
   - Detailed description of selected methods (likely two)
   - Comparison of the selected methods using a set of comparison criteria
3. Evaluation
   - Comparison and discussion of the evaluation tasks, metrics
   - Comparison of the evaluation results using a set of comparison criteria
4. Conclusion
   - What did the comparison of the methods and evaluation results show?
   - Can something be concluded for future work?
5. Bibliography

How to Structure Your Experimental Paper?

1. Introduction and Problem Statement
   - Which problem is addressed? What is the overall approach for addressing it?
   - Overview of the existing methods/papers and use cases for the evaluation (3 pages+)
   - Structure of your paper
2. Description of Your Experimental Design
   - How to you select examples for which challenges?
   - Which prompt designs and language models do you test?
3. Presentation of Experimental Results
   - Present the results of your experiments (tables containing values and deltas).
   - Present the results of your error analysis (types of errors, frequency of these types)
4. Conclusion
   - What did the experiments and the error analysis show?
   - How to your results compare to the experiments presented in the papers?
5. Bibliography

Learn from Examples

- Read survey articles and previous experimental papers and identify the structure from the previous slides
  - Why can this paragraph be found at that position?
  - What is the purpose of some section / subsection?
- Some relevant surveys
  1. Zhao, et al.: A survey of Large Language Models. arXiv:2303.18223
  2. Mialon, et al.: Augmented Language Models: a Survey. arXiv:2302.0784
  3. Wang, et al: A Survey on Large Language Model based Autonomous Agents. arXiv:2308.11432. 2023.
- Textbook on how to write a thesis
  - Zobel: Writing for Computer Science, 3rd Edition, Springer 2014.

Citing Different Types of Publications

1. Journal article
   - Good to cite, current research results
2. Conference and workshop paper
   - Good to cite, current research results
3. Survey articles
   - Good to cite as overviews for specific topics, but prefer individual papers as reference for specific systems
4. Books (sometimes cited)
   - Textbooks
   - Collections of articles/papers => Cite specific paper in book
5. Websites
   - better not cited, exceptions are, e.g., documents like W3C Specifications
   - Do not cite Wikipedia, ever!
   - Use footnotes to refer to project pages, download pages, or technical documentation
6. Slide sets (especially from our lectures)
   - Never cite!

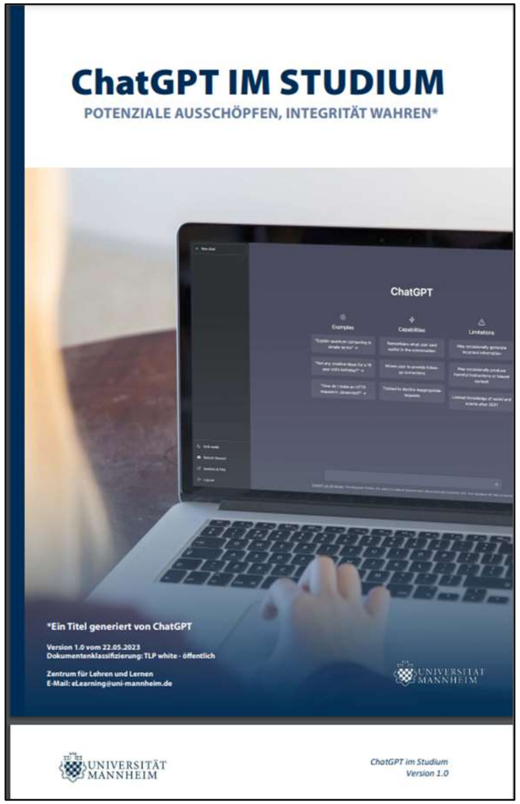
How to Find Relevant Publications?

1. Start with gathering relevant papers from the **surveys**
   1. Zhao, et al.: A survey of Large Language Models. arXiv:2303.18223
   2. Mialon, et al.: Augmented Language Models: a Survey. arXiv:2302.0784
   3. Wang, et al: A Survey on Large Language Model based Autonomous Agents. arXiv:2308.11432. 2023.
2. **Exploit references:** Given a relevant document x
   - Follow references in the past: papers y that x has cited
   - Follow references in the future: papers y that cited x („**cited by**" functionality in Google scholar)
3. **Use Google Scholar or Semantic Scholar**
   - we use it a lot ourselves

Statement About the Tools that You Used

Your report **must include an extra page** about

1. which tools you used
   - ChatGPT, OpenAI API, Perplexity, ….
2. for which purposes
   - structuring your paper
   - summarizing related work
   - writing text for specific chapters
   - improving English grammar and formulations
   - designing experimental setup
   - writing code
   - writing prompts
   - generating training data
   - error analysis
   - ….
3. How useful was each tool for this?



https://www.uni-mannheim.de/media/Einrichtungen/zll/Website_2.0/ChatGPT_Handreichung_Studierende_UMA_Stand_Mai_2023.pdf

4. Questions?