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
- Consultation: Wednesday, 13:30-14:30

Hallo

- **Dr. Steffen Eger**
- Heisenberg Group Leader
- Research Interests:
  - Text Generation & Evaluation
  - Social Science Applications
  - Digital Humanities Applications
- Room: xxx
- eMail: eger.steffen@gmail.com
- Consultation: no fixed office hours, by appointment

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: xxx
- eMail: christoph.leiter@uni-bielefeld.de

Hallo

- **M. Sc. Daniil Larionov**
- Graduate Research Associate
- Research Interests:
  - Evaluation Metrics for Text Generation
  - Efficiency
- Room: xxx
- eMail: daniil.larionov@uni-bielefeld.de

Hallo

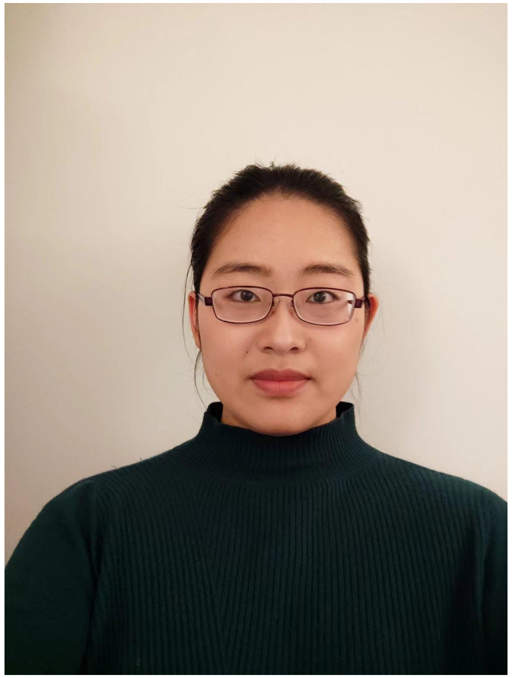
- **M. Sc. Wi-Inf. Keti Korini**
- Graduate Research Associate
- Research Interests:
  - Table Annotation using Deep Learning
  - Schema Matching
- Room: B6, 26, C 1.03
- eMail: kkorini@uni-mannheim.de

Hallo

- **M. Sc. 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: xxx
- eMail: ran.zhang@uni-bielefeld.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. Schlüter, Maria
2. Eroglu, Zeynep
3. Tomori, Flavjo
4. Jano, Stiliana
5. Bajri, Deidamea
6. Delev, Daniel
7. Wade, Saloni
8. Arenz, Joel
9. Hüllen, Kilian
10. Höppner, Jannis
11. Nghiem, Thuy
12. Petra Revesz

Agenda of Today‘s Kickoff Meeting

1. Seminar organization
2. Introduction to LLMs and Prompt Engineering
3. Topic Assignment
4. How to structure your seminar 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
- Designing experiments and present experimental results
- Searching and citing scientific papers / journal articles
- How to structure your thesis and presentation
- How to write a scientific paper using LaTeX

Schedule

Date	Session
Tuesday, 19.09.2023 (10:15-11:45)	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 9.10.2023	Meet with your mentor to discuss outline and/or experimental setup
	Prepare draft of your presentation
Until 27.10.2023	Send draft presentation to your mentor
	Finalize your presentation
Monday, 20.11.2022 (10:00-12:00) (14:00-16:00)	Presentation and discussion of your topic (30 % of your final grade)
	Write seminar thesis
Wednesday, 31.01.2024	Submission of your seminar thesis (70 % of your final grade)

Formal Requirements

- Presentation
  - 12 minutes + 8 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.

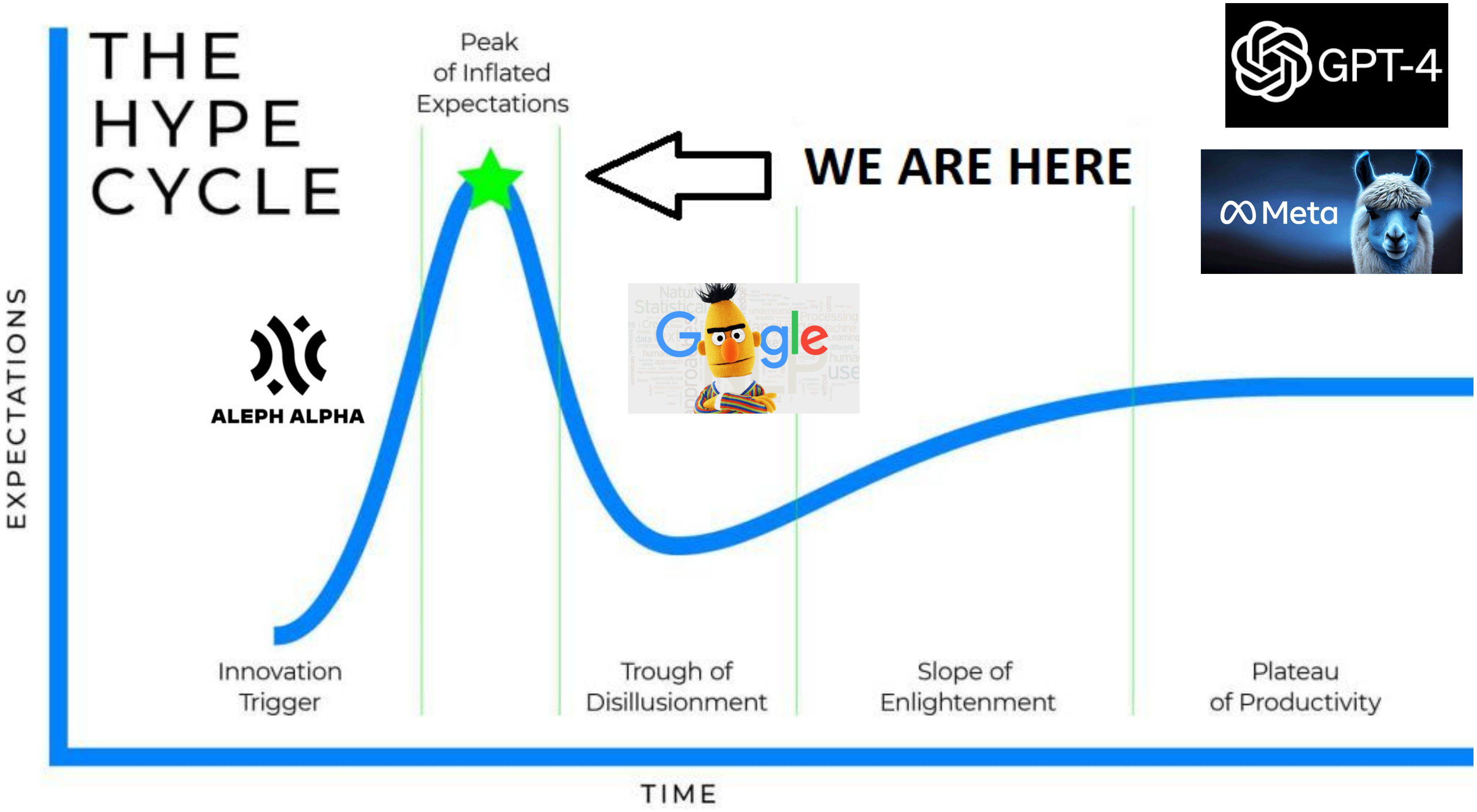
1 Introduction

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 LLMs and Prompt Engineering

Large Language Models



Large Language Models

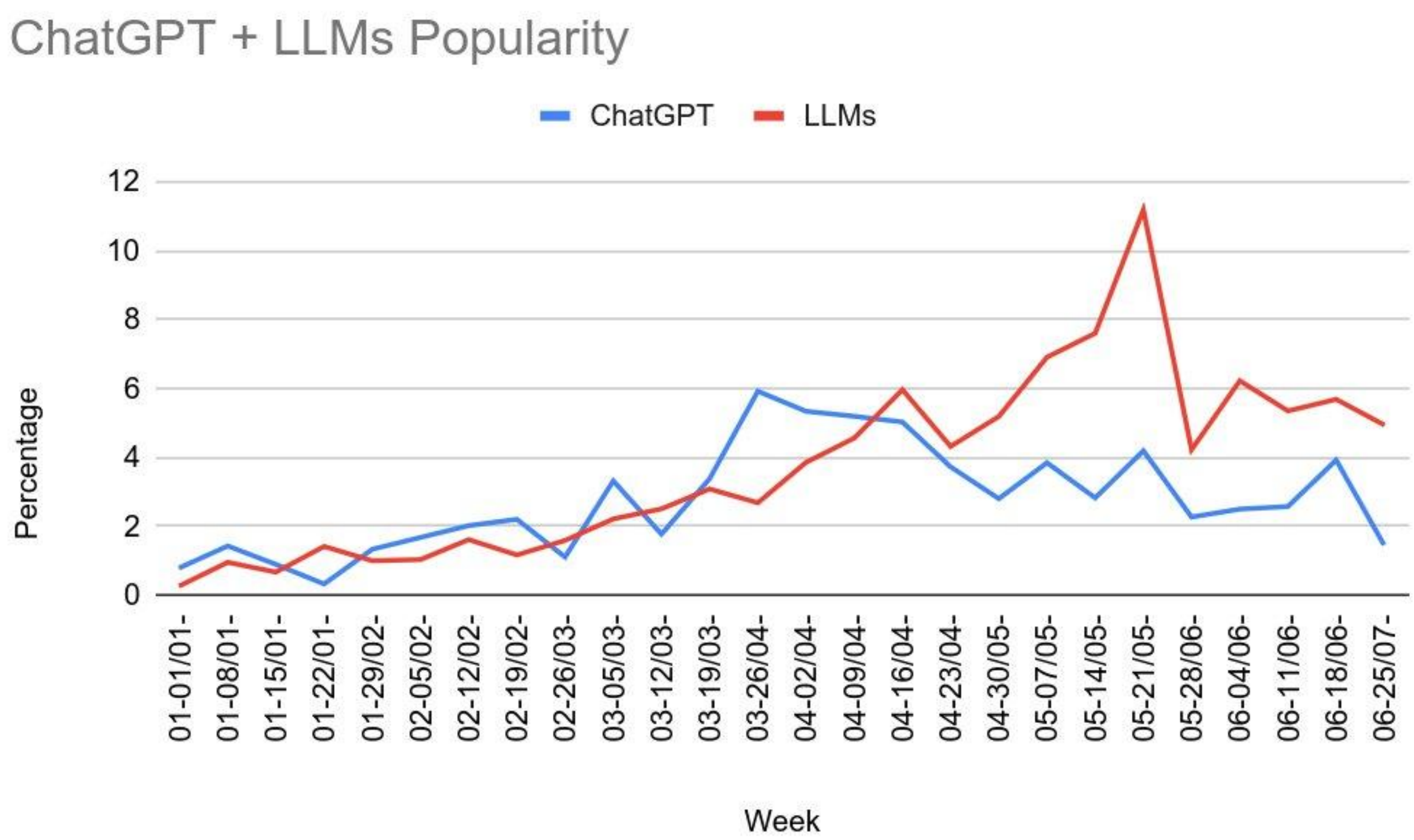


Figure 4: Popularity of ChatGPT and LLMs (in percentage of papers having the words in their abstracts or titles) over time in our dataset.

Source: https://arxiv.org/pdf/2308.04889.pdf

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

  $$p(x_1,x_2,\dots,x_T)$$

  for a sequence of words/tokens $x_1,\dots,x_T$
- Often reformulated as a product of conditional probabilities

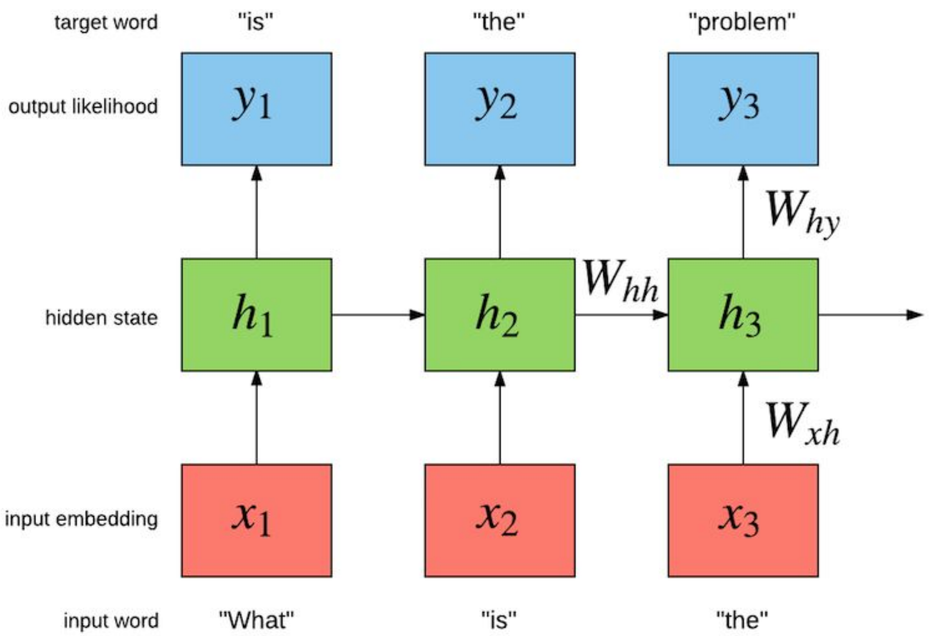
  $$p(x_1,x_2,\dots,x_T) = p(x_1) * p(x_2|x_1) * \dots * p(x_T|x_1,\dots,x_{T-1})$$

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

Large Language Models: A very brief introduction

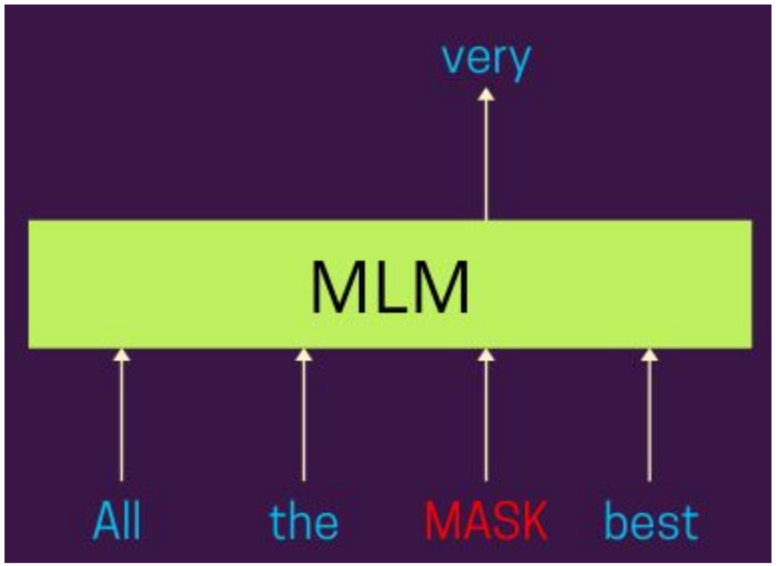
How to?

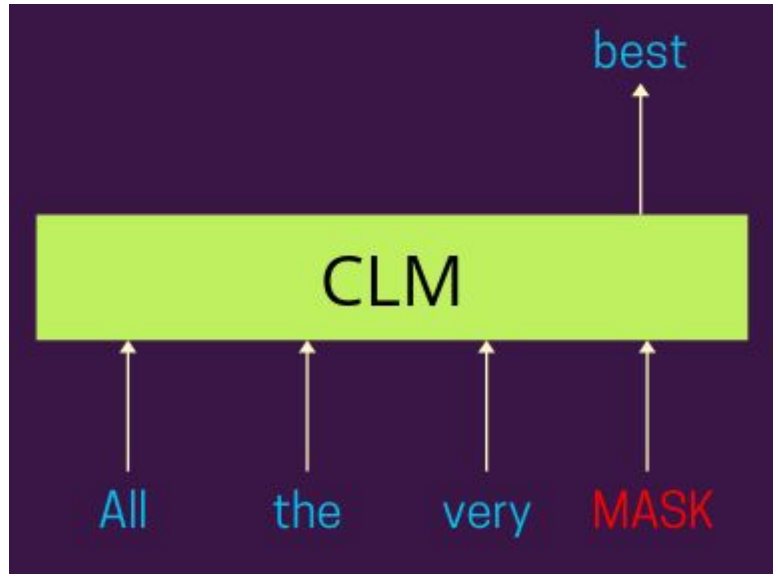
- Early models were n-gram count models (until 2010s)
- “Embedding” based models implemented in the mid-2010s
  - recurrent neural net based LMs
- Since 2018:
  - Transformer based LMs



Large Language Models: A very brief introduction

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





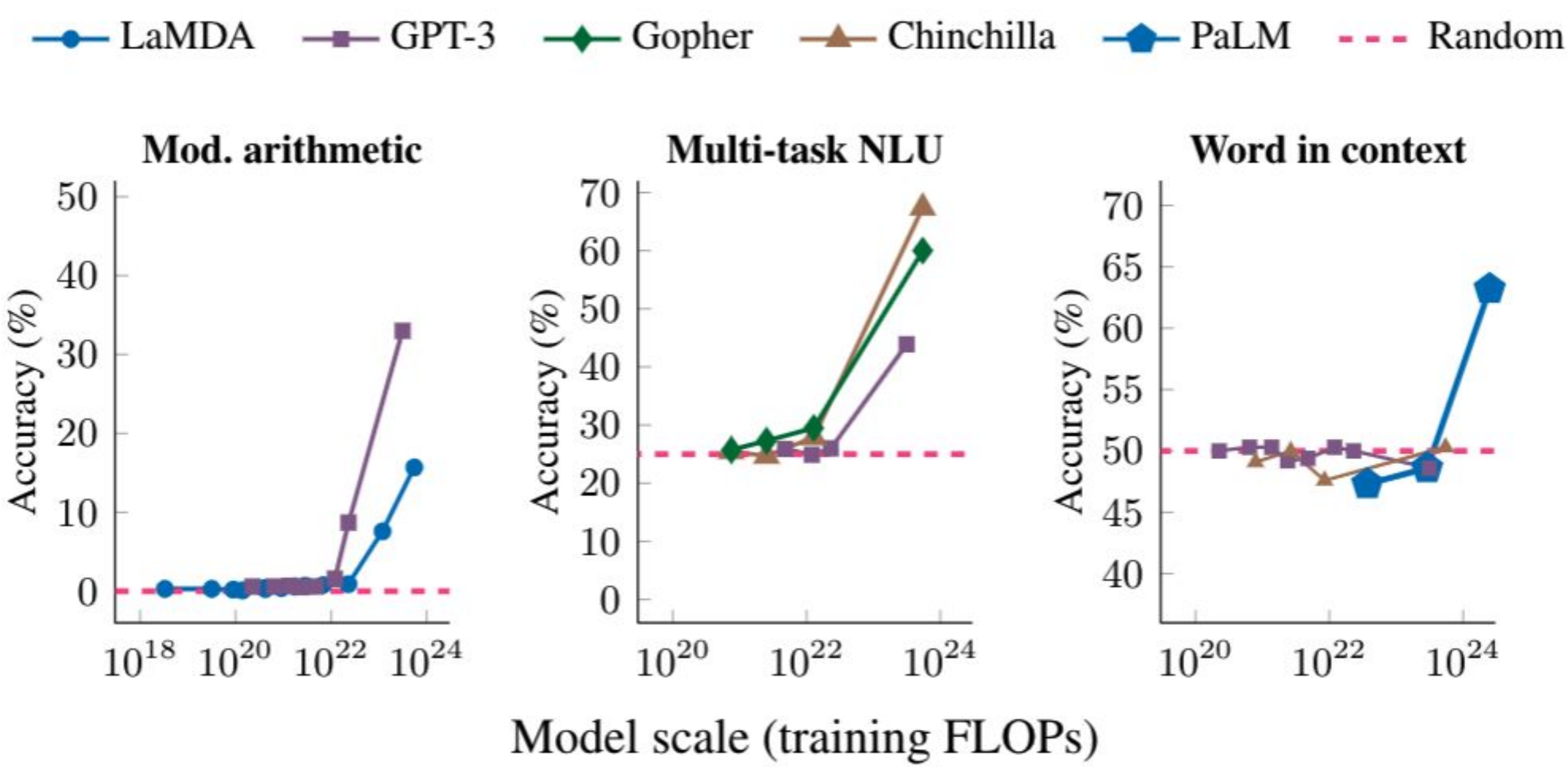
Large Language Models: A very brief introduction

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
  - …

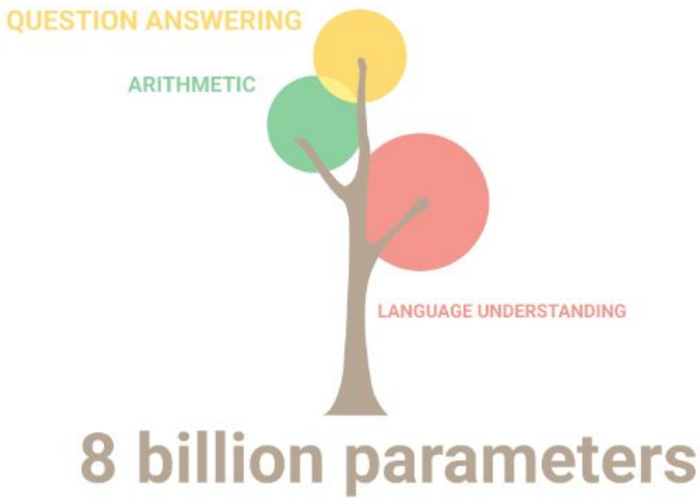
- As you make the **LMs bigger and bigger and bigger**
- If they are trained on **large enough datasets**
- with “emergent” abilities

Large Language Models: A very brief introduction



- with “emergent” abilities

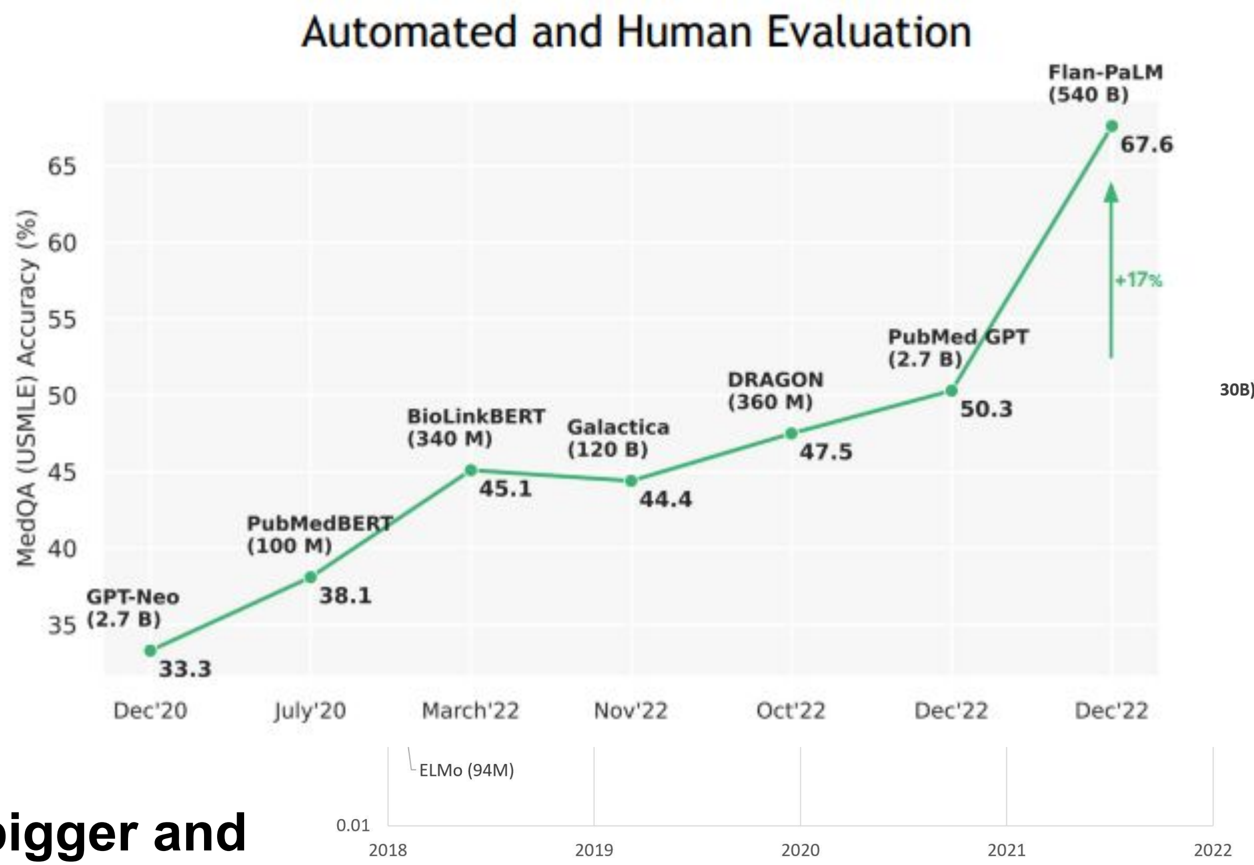
Large Language Models: A very brief introduction



Large Language Models: A very brief introduction

Main insight in last few yea

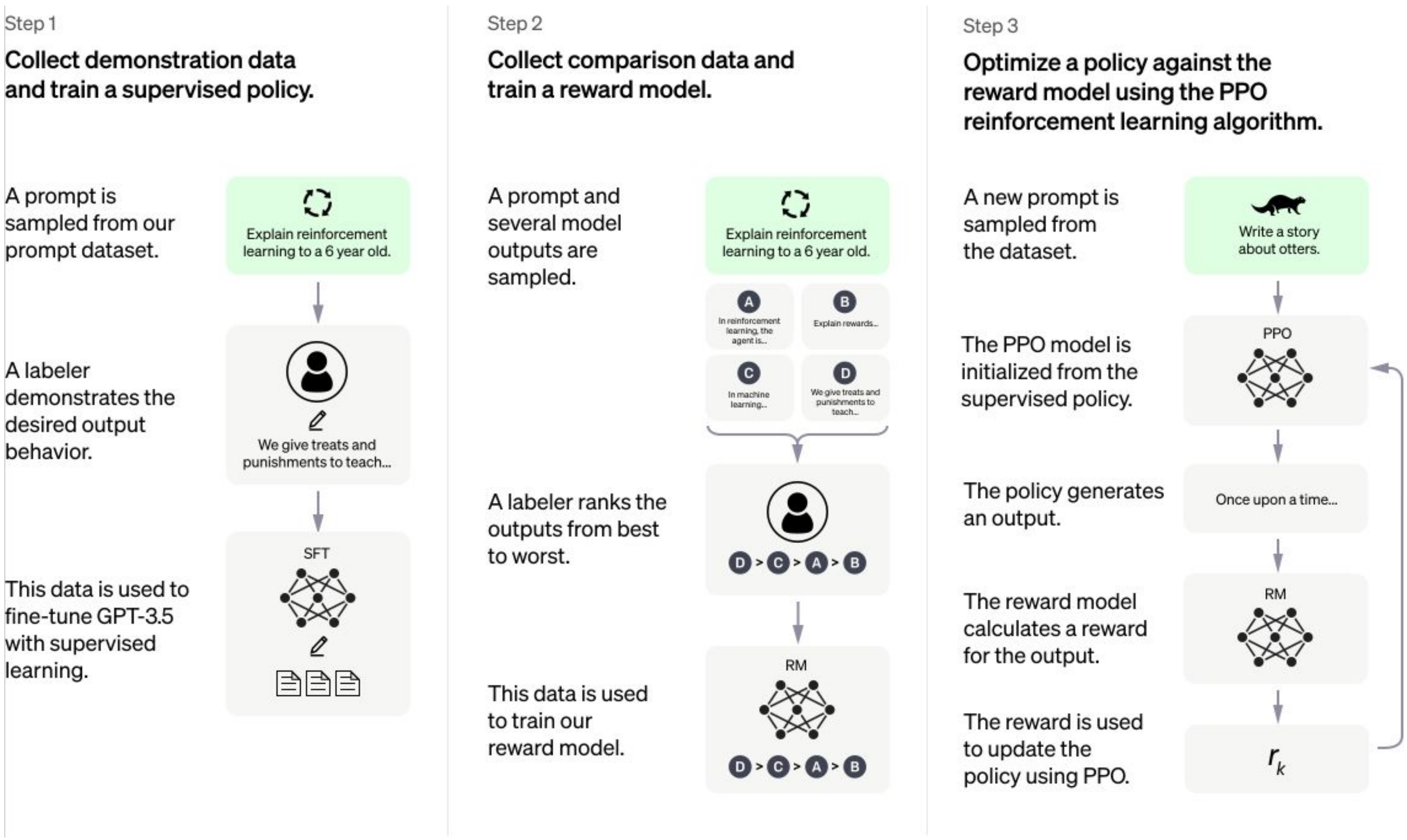
- LMs cannot only do te
  - part-of-speech tag
  - machine translatio
  - poetry generation
  - sentiment analysis
  - …



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

Large Language Models: A very brief introduction

ChatGPT

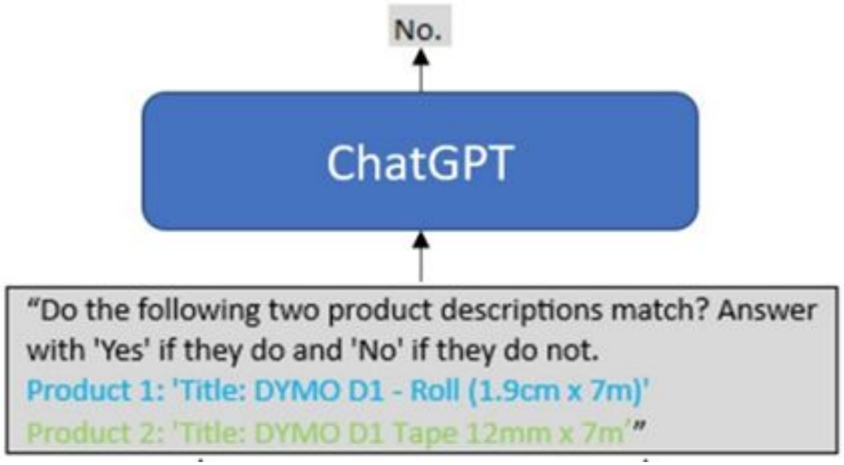


Prompt Engineering: A very brief introduction

- Prompt

  A prompt is natural language text

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



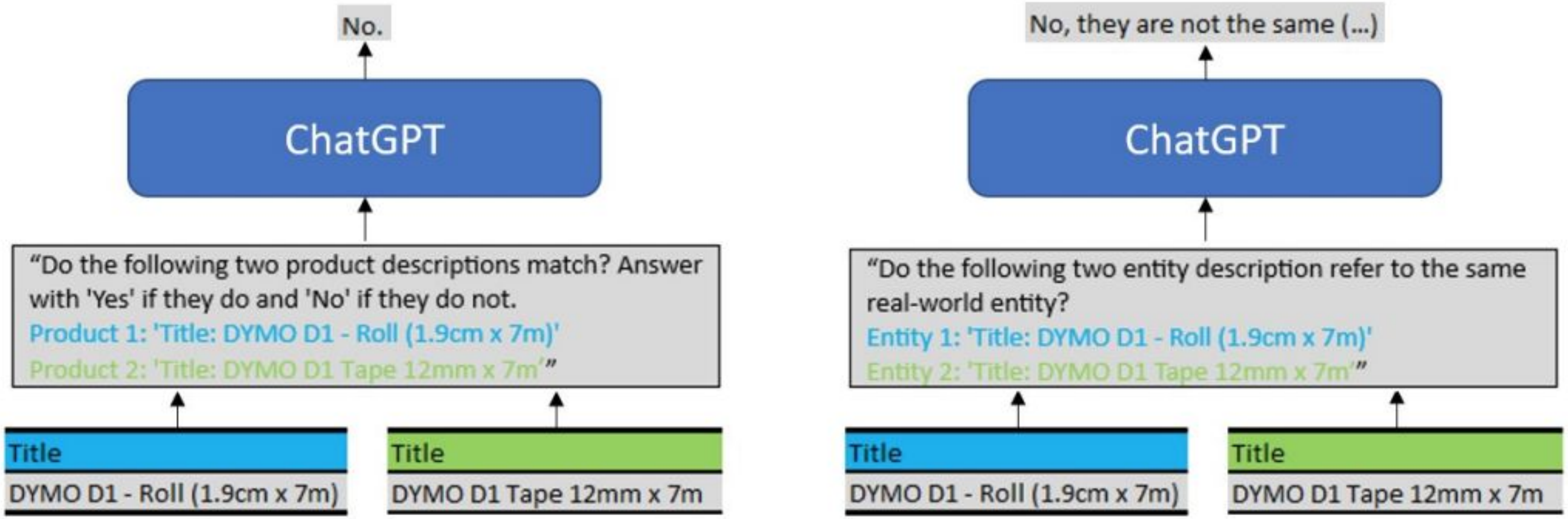
- Prompt Engineering

  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 strongly vary depending on the prompt formulation.
- Three patterns emerge:
  1. domain-specific wording leads to more stable results
  2. describing the task in simpler language works better
  3. forcing the model to answer with simple “Yes” or “No” is helpful

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

Results: In-Context Learning

Selection heuristic	Shots	P	R	F1	Δ F1	Cost (¢) per pair	Cost increase	Cost increase per Δ F1
ChatGPT-zeroshot	0	71.01	**98.00**	82.35	-	0.14	-	-
ChatGPT-random	6	78.33	94.00	85.45	3.10	0.77	450%	145%
	10	79.66	94.00	86.24	3.89	1.13	707%	182%
	20	78.95	90.00	84.11	1.76	2.07	1379%	783%
ChatGPT-handpicked	6	76.19	96.00	84.86	2.51	0.72	414%	165%
	10	80.00	96.00	87.27	4.92	1.00	614%	125%
	20	79.66	94.00	86.24	3.89	2.03	1350%	347%
ChatGPT-related	6	80.36	90.00	84.91	2.56	0.68	386%	151%
	10	**89.58**	86.00	87.76	5.41	1.05	650%	120%
	20	88.46	92.00	**90.20**	7.85	1.97	1307%	167%
GPT3.5-handpicked	10	61.97	88.00	72.72	-9.63	10.54	7429%	771%
	20	61.43	86.00	71.67	-10.68	19.71	13979%	1309%
GPT3.5-related	10	67.69	88.00	76.52	-5.83	10.04	7071%	1213%
	20	61.43	86.00	71.67	-10.68	20.34	14429%	1351%

- Performance increase of **~3%** F1 with just small number of examples
- Best performance: **20 related** examples lead to **~8%** F1 increase
- Increased performance comes with a **cost increase** of > 100% per gained percentage point of F1

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
- Intrinsic understanding of product features necessary

Results – Domain Knowledge

Table 5: Matching Knowledge results

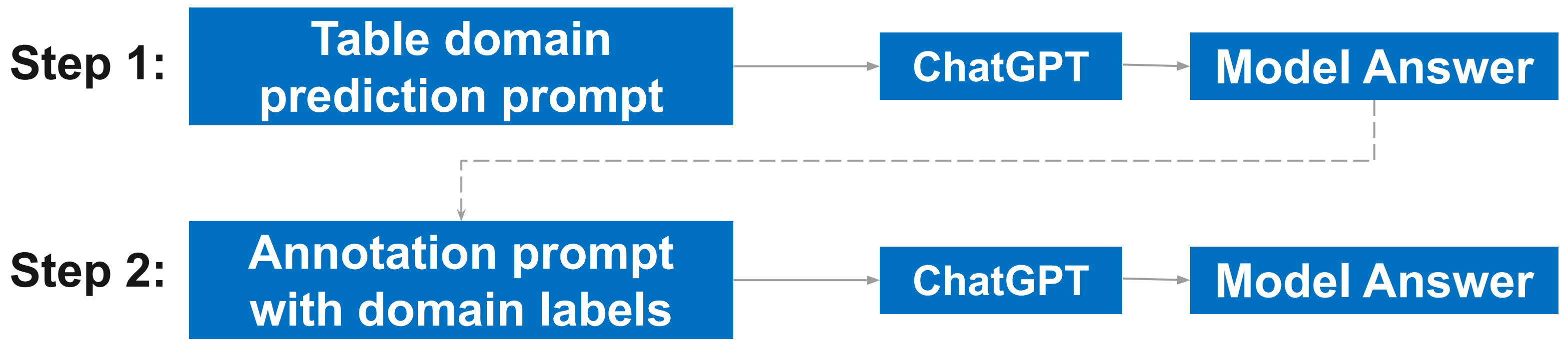
Prompt	Shots	P	R	F1	Δ F1	Cost (¢) per pair	Cost increase	Cost increase per Δ F1
ChatGPT-zeroshot	0	71.01	**98.00**	82.35	-	0.14	-	-
ChatGPT-zeroshot with rules	0	80.33	**98.00**	88.29	5.94	0.28	100%	17%
ChatGPT-related	6	80.36	90.00	84.91	2.56	0.68	386%	151%
	10	89.58	86.00	87.76	5.41	1.05	650%	120%
	20	88.46	92.00	**90.20**	7.85	1.97	1307%	167%
ChatGPT-related with rules	6	90.70	78.00	83.87	1.52	0.79	464%	305%
	10	90.91	80.00	85.11	2.76	1.17	736%	267%
	20	**91.11**	82.00	86.32	3.97	2.09	1393%	351%

- Matching rules lead to increase in ~9% Precision and ~6% F1
- Similar but not as strong effect as providing related in-context examples
- Rules are cheaper to derive, cost of a query is lower

Multi-Step-Pipelines

- **Approach:** Split task into multiple prompts, e.g. for table annotation
  1. predict domain/type of complete table
  2. perform annotation using reduced set of domain-specific labels

- **Advantages:**
  1. save token space for large vocabularies
  2. simplify the annotation task as the model chooses from smaller set of labels



Impact of the LLM/Prompt Combination

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

- Zero-shot performance of GPT4 is similar to ChatGPT using related in-context examples
- Falcon-40b model based on Llama not good enough for the task
- StableBeluga2 model based on Llama2 already achieves good performance
- The gap between OpenAI and open-source models is closing ☺
- The effectiveness of a prompt depends on the LLM ☹
- So, you always need to compare prompt/LLM pairs

2. Seminar Topics and Topic Assignment

- The seminar features literature as well as experimental topics.
- The goal of the **literature topics** will be to summarize the state of the art concerning the application and evaluation of LLMs.
- The goal of the **experimental topics** will be to verify prompt engineering techniques by applying them to tasks beyond the tasks used in the respective papers.

Topics

1. Literature Topic: Explainability of LLMs

- Student: Jannis Höppner
- Mentor: Christoph Leiter

- Some papers as starting point
- Yao et al., Tree of Thoughts: Deliberate Problem Solving with Large Language Models
- Turpin et al., Language Models Don't Always Say What They Think: Unfaithful Explanations in Chain-of-Thought Prompting
- Lanham et al., Measuring Faithfulness in Chain-of-Thought Reasoning
- Radhakrishnan et al., Question Decomposition Improves the Faithfulness of Model-Generated Reasoning

Topics

2. Literature Topic: Efficiency of LLMs

- Student: Flavjo Tomori
- Mentor: Daniil Larionov

- Some papers as starting point
- Lee et al., Surveying (Dis)Parities and Concerns of Compute Hungry NLP Research
- Touvron et al., LLaMA: Open and Efficient Foundation Language Models
- Dettmers et al., QLoRA: Efficient Finetuning of Quantized LLMs
- Hsieh et al., Distilling Step-by-Step! Outperforming Larger Language Models with Less Training Data and Smaller Model Sizes
- Gu et al., Knowledge Distillation of Large Language Models

Topics

3. Literature Topic: Agent-Based Modeling via LLMs

− Student: Stiliana Jano

− Mentor: Ran Zhang

- Some papers as starting point
- Park et al., Generative Agents: Interactive Simulacra of Human Behavior
- Li et al., CAMEL: Communicative Agents for "Mind" Exploration of Large Scale Language Model Society
- Boiko et al., Emergent autonomous scientific research capabilities of large language models
- Zhuge et al., Mindstorms in Natural Language-Based Societies of Mind
- Wang et al., Interactive Natural Language Processing

Topics

4. Literature Topic: LLMs for the Social Sciences

− Student: Saloni Wade

− Mentor: Steffen Eger

- Some papers as starting point
- Ziems et al., Can Large Language Models Transform Computational Social Science?
- Feng et al., From Pretraining Data to Language Models to Downstream Tasks: Tracking the Trails of Political Biases Leading to Unfair NLP Models
- Hartmann et al., The political ideology of conversational AI: Converging evidence on ChatGPT's pro-environmental, left-libertarian orientation

Topics

5. Literature Topic: Limitations of LLMs

− Student: Zeynep Eroglu

− Mentor: Steffen Eger

- Some papers as starting point
- Frieder et al., Mathematical Capabilities of ChatGPT
- Borji, A Categorical Archive of ChatGPT Failures
- Wang et al., Large Language Models are not Fair Evaluators
- Schick et al., Toolformer: Language Models Can Teach Themselves to Use Tools
- Bang et al., A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity

Topics

6. Literature Topic: LLMs for Education+Science

− Student: Daniel Delev

− Mentor: Steffen Eger

- Some papers as starting point
- Baidoo-Anu et al., Education in the Era of Generative Artificial Intelligence (AI): Understanding the Potential Benefits of ChatGPT in Promoting Teaching and Learning
- Choi et al., ChatGPT Goes to Law School
- Boiko et al., Emergent autonomous scientific research capabilities of large language models
- Meyer et al., ChatGPT and large language models in academia: opportunities and challenges

7. Literature Topic: Multimodality an LLMs

− Student: Thuy Nghiem

− Mentor: Steffen Eger

- Some papers as starting point
- Liu et al., Visual instruction tuning
- Video-LLaMA: An Instruction-tuned Audio-Visual Language Model for Video Understanding
- InstructBLIP: Towards General-purpose Vision-Language Models with Instruction Tuning

Topics

8. Experimental Topic: Chain-of-Thought Prompting

− Student: Maria Schlüter

− Mentor: Keti Korini

- Wei, Jason, et al. "Chain-of-thought prompting elicits reasoning in large language models." Advances in Neural Information Processing Systems 35 (2022): 24824-24837.
- Kojima, Takeshi, et al. "Large language models are zero-shot reasoners." Advances in neural information processing systems 35 (2022): 22199-22213.
- Zhang, Zhuosheng, et al. "Automatic chain of thought prompting in large language models." arXiv preprint arXiv:2210.03493 (2022).

Topics

9. Experimental Topic: Knowledge Generation Prompting

− Student: Deidamea Bajri

− Mentor: Alexander Brinkmann

- Liu, Jiacheng, et al. "Generated knowledge prompting for commonsense reasoning." arXiv preprint arXiv:2110.08387 (2021).

- W. Yu, D. Iter, S. Wang, Y. Xu, M. Ju, S. Sanyal, C. Zhu, M. Zeng, and M. Jiang. 2023. Generate rather than Retrieve: Large Language Models are Strong Context Generators. In ICLR2023

Topics

10. Experimental Topic: Tree of Thoughts Prompting

− Student: Kilian Hüllen

− Mentor: Ralph Peeters

- Yao, Shunyu, et al. "Tree of thoughts: Deliberate problem solving with large language models." arXiv preprint arXiv:2305.10601 (2023).
- Long, Jieyi. "Large Language Model Guided Tree-of-Thought." arXiv preprint arXiv:2305.08291 (2023).
- Besta et al. “Graph of Thoughts: Solving Elaborate Problems with Large Language Models” arXiv preprint arxiv.org/abs/2308.09687 (2023)

Topics

11. Experimental Topic: Plan-and-Solve Prompting

− Student: Joel Arenz

− Mentor: Keti Korini

- Wang, Lei, et al. "Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models." arXiv preprint arXiv:2305.04091 (2023).
- Chen, et al. “Symphony: Towards Natural Language Query Answering over Multi-modal Data Lakes” CIDR, 2023.
- https://python.langchain.com/docs/modules/agents/agent_types/plan_and_execute

12. Experimental Topic: Data Fusion using LLMs

- Student: Petra Revesz
- Mentor: Alexander Brinkmann

- Ahmad, Mohammad Shahmeer, et al. "RetClean: Retrieval-Based Data Cleaning Using Foundation Models and Data Lakes." arXiv preprint arXiv:2303.16909 (2023).
- Narayan, Avanika et. al. 2022. Can Foundation Models Wrangle Your Data? VLDB2022 (4), 738–746.
- Jens Bleiholder and Felix Naumann. 2009. Data fusion. ACM Comput. Surv. 41, 1, Article 1 (January 2009), 41 pages. https://doi.org/10.1145/1456650.1456651

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 technique from the paper
  2. present evaluation task and results from the paper
  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
   - Structure of your paper
2. Description of Experimental Design
   - What is your 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?
- Important
  - Read survey articles!
  - Read conference or journal papers
- Some relevant surveys
  - Zhao, et al.: A survey of Large Language Models. arXiv:2303.18223
  - Mialon, et al.: Augmented Language Models: a Survey. arXiv:2302.0784
- Textbook on how to write a thesis
  - Zobel: Writing for Computer Science, 3rd Edition, Springer 2014.

Citing Different Types of Publications

- Journal article
  - Good to cite, current research results
  - Survey articles (very good for an overview)
- Conference and workshop paper
  - Good to cite, current research results
- Books (sometimes cited)
  - Textbooks
  - Collections of articles/papers => Cite specific paper in book
- 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
- Slide sets (especially from our lectures)
  - Never cite!

How to Find Relevant Publications?

- Use Standard Search Engines
- **Use Google Scholar**
  - we use it a lot ourselves
- Search Engines of the University's library
  - see slides from the library course
- **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)

4. Questions?