**These slides:**
https://bit.ly/resp-ai-od

# Getting Past Chit-chat with ChatGPT: Large Language Models and Structured Outputs

**Ondřej Dušek**

Responsible Use of AI in Universities
23.11.2023

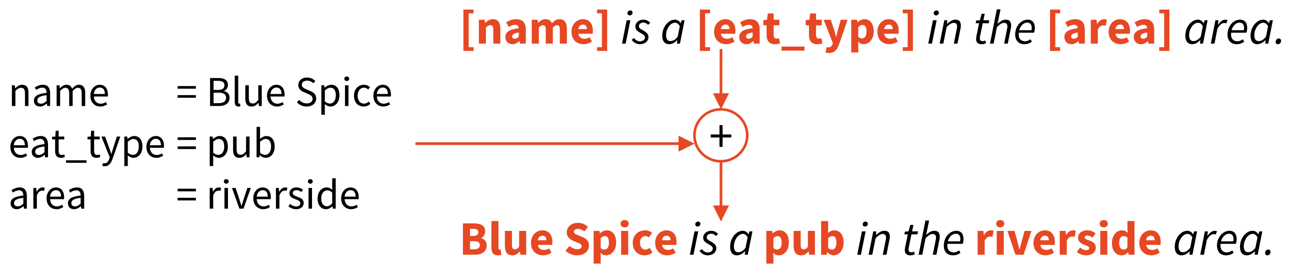
Thanks: Vojtěch Hudeček, Jonáš Kulhánek, Tomáš Nekvinda

Charles University
Faculty of Mathematics and Physics
Institute of Formal and Applied Linguistics

BY SA
unless otherwise stated

# Language Generation

- NLG = **some input** (data/context…) ➔ **text**
- Traditional approach: **templates** – fill in blanks
  - safe, but laborious & not so fluent



**[name]** *is a* **[eat_type]** *in the* **[area]** *area.*

name = Blue Spice
eat_type = pub
area = riverside

**Blue Spice** *is a* **pub** *in the* **riverside** *area.*

vs.

- **End-to-end**
  - feed data/context, get text word-by-word, left-to-right
  - neural net (Transformer → →) handles everything
- Very **fluent** & convincing outputs
- !Opaque & has **no guarantees on accuracy**
  - used essentially as a black box, internals unknown

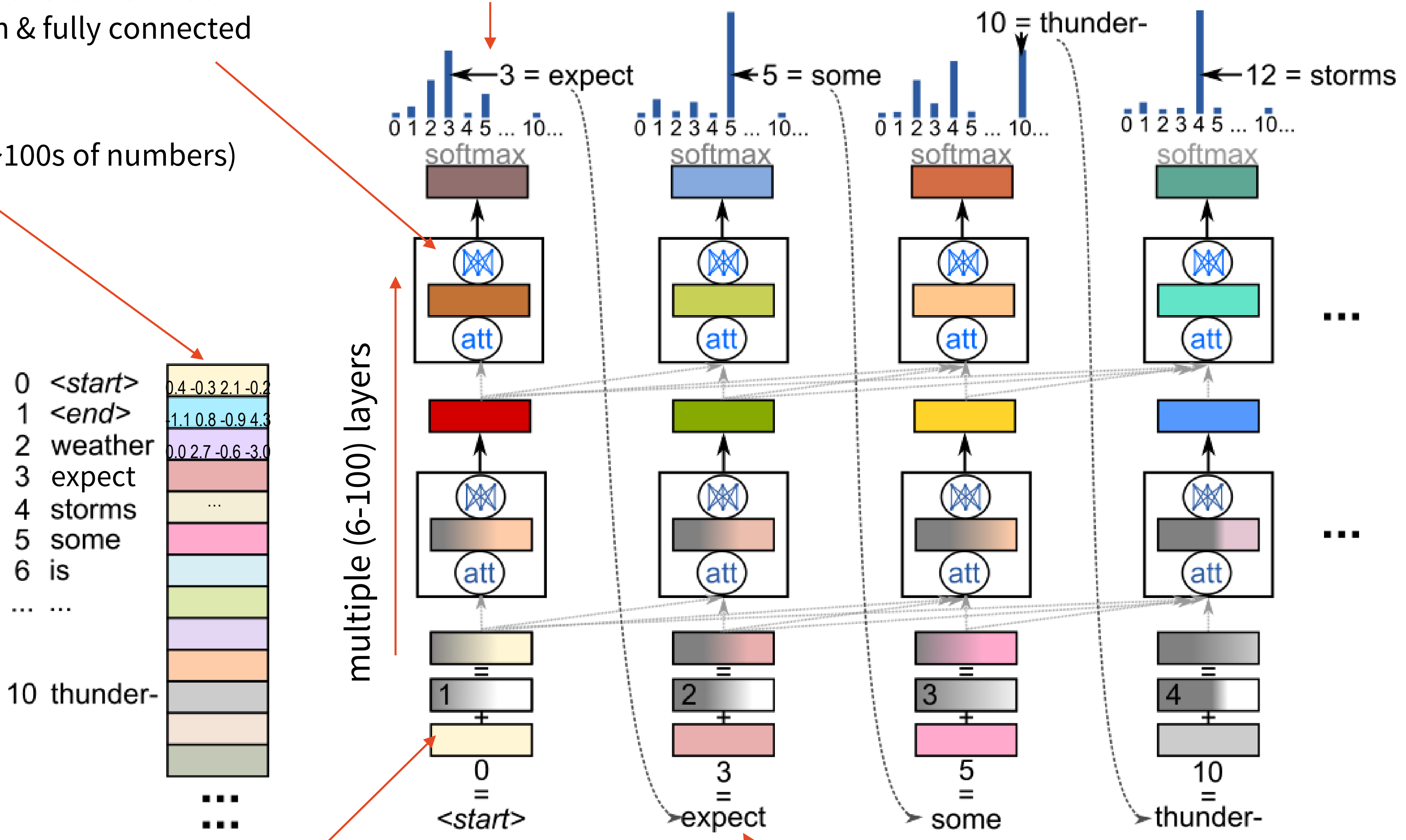
# Transformer neural language model

layer = Transformer block:
attention & fully connected

predicting next word (prob. dist.)

embeddings (~100s of numbers)

numbered
(sub)words



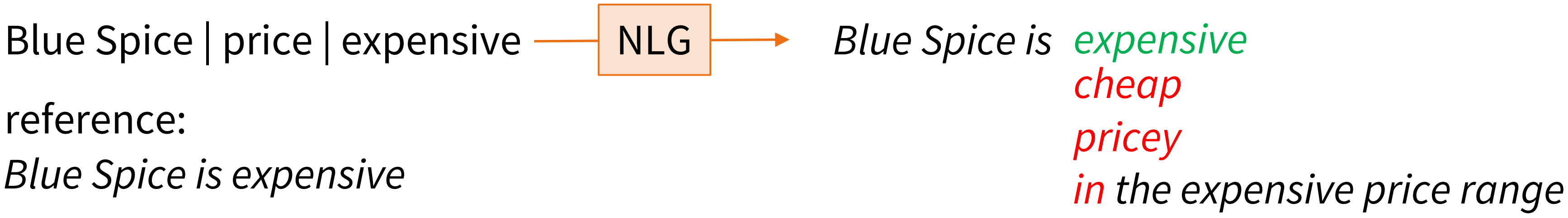
multiple (6-100) layers

encoding position

when generating: predicted word fed back in
when training/prompting: ignore predicted, use yours

# Training a Neural NLG System

- Reproduce sentences from data
  - replicate exact word at each position
- Fully trained from data
  - initialize model with random parameters
  - input example: didn’t hit the right word → update parameters



Blue Spice | price | expensive → NLG → *Blue Spice is expensive*

*cheap*

*pricey*

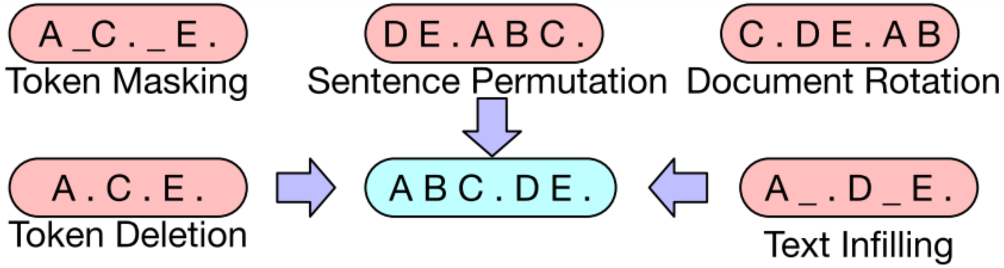
*in the expensive price range*

reference:
*Blue Spice is expensive*

- Very **low level**, no concept of sentence / text / aim

# Pretraining & Finetuning

1. **Pretrain** a model on huge data (simple language-based tasks)
    - predicting next word
    - reconstructing garbled texts
2. **Fine-tune** on your smaller data
    - same as training, but starting from a better model



(Lewis et al., 2020)
https://www.aclweb.org/anthology/2020.acl-main.703

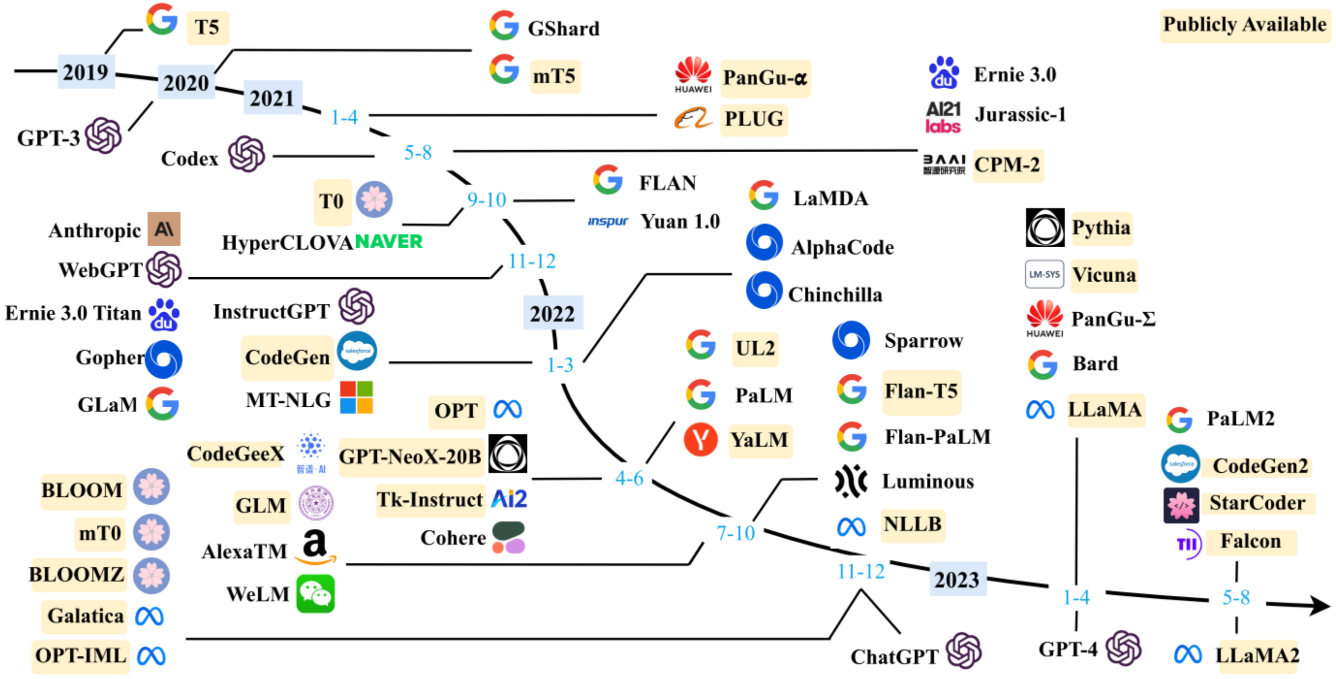
- Models free for download (https://huggingface.co/)
    - BERT/RoBERTa, GPT-2, BART, T5…
    - 100k-1B parameters – runs easily on regular GPUs

https://jlibovicky.github.io/2023/02/07/Otazky-a-odpovedi-o-ChatGPT-a-jazykovych-modelech.html

# Large language models (LLMs): Pretrain & Prompt

(Zhao et al., 2023)
http://arxiv.org/abs/2303.18223

- 10-100B parameters
  - hard(er) to run (OPT, LlaMa, Falcon…)
  - or API only (GPT-3/4, ChatGPT, Bard…)
- architecture mostly the same
- pretrained on more data

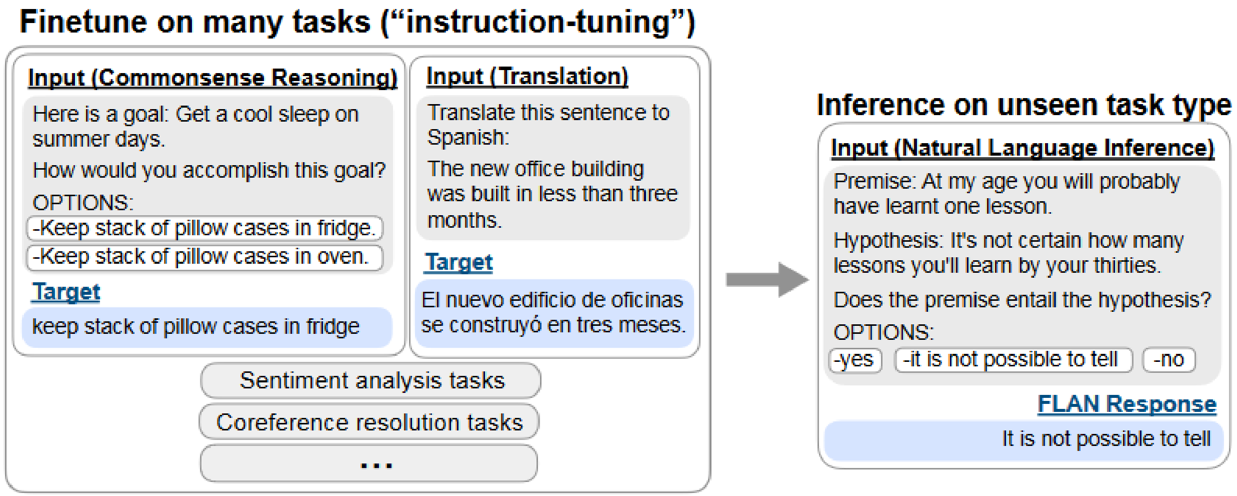


- **prompting:** feed in context / examples / question, get reply
  - finetuning can help, but it's expensive & has less effect than in smaller LMs
  - **prompt engineering** may be required
  - still the same problems– hard to control

# Instruction Tuning / RL from Human Feedback

(Wei et al., 2022) https://arxiv.org/abs/2109.01652

- LLMs finetuned for prompting
  - **instructions** (task description) **+ solution** in prompts
  - “in-domain” for what it’s used later
  - large datasets available



**Finetune on many tasks (“instruction-tuning”)**

**Input (Commonsense Reasoning)**

Here is a goal: Get a cool sleep on summer days.

How would you accomplish this goal?

OPTIONS:

-Keep stack of pillow cases in fridge.

-Keep stack of pillow cases in oven.

**Target**

keep stack of pillow cases in fridge

**Input (Translation)**

Translate this sentence to Spanish:

The new office building was built in less than three months.

**Target**

El nuevo edificio de oficinas se construyó en tres meses.

**Inference on unseen task type**

**Input (Natural Language Inference)**

Premise: At my age you will probably have learnt one lesson.

Hypothesis: It's not certain how many lessons you'll learn by your thirties.

Does the premise entail the hypothesis?

OPTIONS:

-yes -it is not possible to tell -no

**FLAN Response**

It is not possible to tell

- RL improvements on top (~InstructGPT/ChatGPT/…):
  - rating model based on human ratings gives rewards for outputs
  - main improvement: **reward is global**, not token-by-token
  - can be safer (low reward for bad behavior)

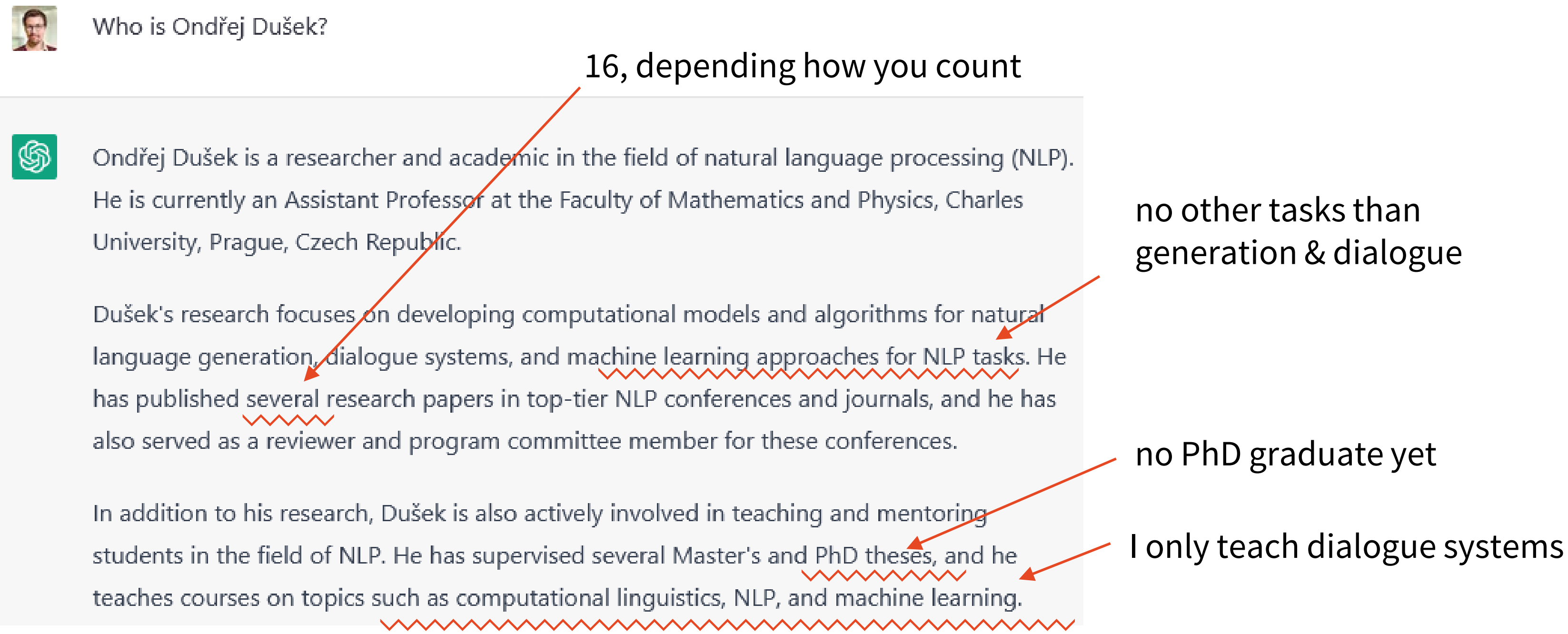
(Ouyang et al., 2022) http://arxiv.org/abs/2203.02155
https://openai.com/blog/chatgpt

(Rafailov et al., 2023) http://arxiv.org/abs/2305.18290

https://nlpnewsletter.substack.com/p/instruction-tuning-vol-1

# LLMs Caveats

- RLHF training scheme ~ be **convincing** but **not necessarily true**
- Not reliable for QA: only uses information it memorized, “hallucinates”



Who is Ondřej Dušek?

Ondřej Dušek is a researcher and academic in the field of natural language processing (NLP). He is currently an Assistant Professor at the Faculty of Mathematics and Physics, Charles University, Prague, Czech Republic.

Dušek's research focuses on developing computational models and algorithms for natural language generation, dialogue systems, and machine learning approaches for NLP tasks. He has published several research papers in top-tier NLP conferences and journals, and he has also served as a reviewer and program committee member for these conferences.

In addition to his research, Dušek is also actively involved in teaching and mentoring students in the field of NLP. He has supervised several Master's and PhD theses, and he teaches courses on topics such as computational linguistics, NLP, and machine learning.

16, depending how you count

no other tasks than generation & dialogue

no PhD graduate yet

I only teach dialogue systems

- Can use information provided in the prompt though (→ →)

# Task-oriented Dialogue

- **Assistant:** fulfill user requests (book a hotel/restaurant/taxi etc.)
- MultiWOZ: benchmark for multiple connected domains
  - 10k dialogues, extensive annotation (but noisy!)

(Budzianowski et al., 2018)
https://aclanthology.org/D18-1547/

1. user: I am looking for a train from Cambridge to London Kinks Cross.
state: *{train {departure = cambridge, destination = london kings cross}}*
DB: *{train (70) {…}}*
[count] [departure] [destination]
system: There are 70 trains departing from Cambridge to London Kings Cross. What day would you like to travel?

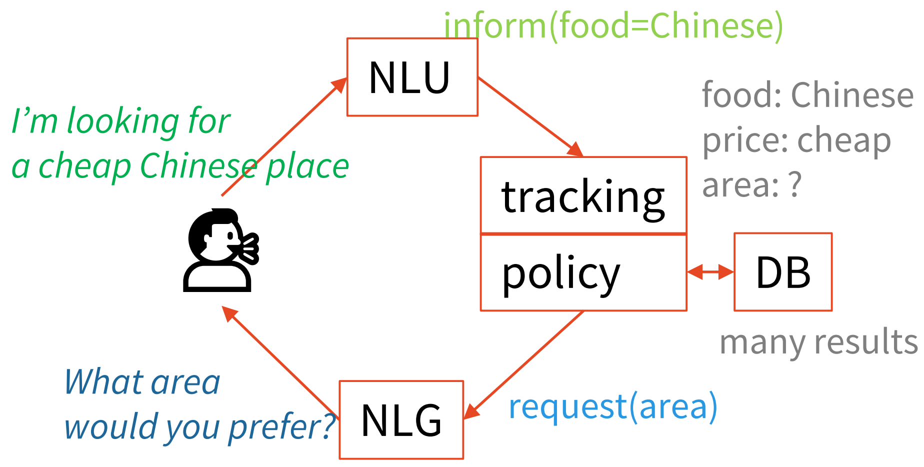
2. user: I would like to leave on Saturday after 18:45.
state: *{train {day = saturday, departure = cambridge, destination = london kings cross, leave at = 18:45}}*
DB: *{train (3) {arrive by = 19:51,21:51,23:51; id = TR0427,TR0925,TR4898; leave at = 19:00,21:00,23:00; … }}*
[id] [leave_at] [arrive_by]
system: TR0427 leaves at 19:00 and arrives by 19:51 . Would you like to book this train?

3. user: Yes, I would like to book it for eight people.
[reference]
system: I have booked it for you. Your reference number is 00000057. Is there anything else I can help you with?

4. user: I am also looking for an expensive restaurant in the centre.
belief: *{restaurant {area = centre, price range = expensive} train {...}}*
DB: *{restaurant (33) {area = centre (33); name=Curry Garden, ...; ...}, ...}*
[count] [price_range] [area]
system: There are 33 expensive restaurants in the centre. Is there a particular type of food you would like?

# End-to-end Neural Dialogue

- Traditional: separate components (NLU→DM→NLG)
  - trained separately, possibly optimization by RL
- **End-to-end models**
  - single neural LM for NLU&DM&NLG
  - word-by-word response generation



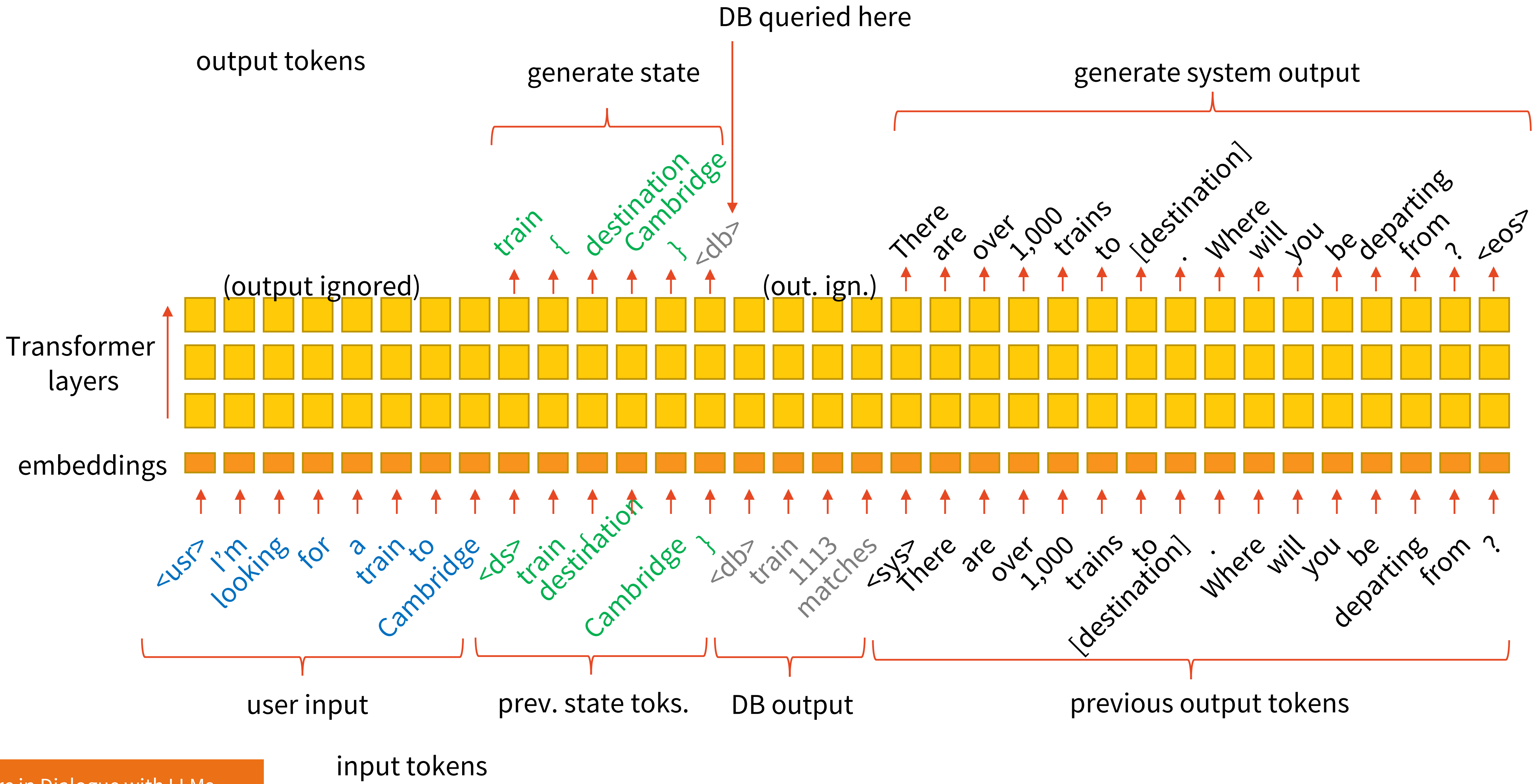
**AuGPT:** finetuned GPT-2 LM (~100M params)

- Multi-step, all word-by-word:
  1. feed in dialogue context
  2. generate dialogue state (as text)
  3. query DB, feed in DB results as text
  4. generate response

(Kulhánek et al., 2021)
http://arxiv.org/abs/2102.05126
https://github.com/ufal/augpt

# End-to-end Neural Dialogue with GPT-2

(Kulhánek et al., 2021)
http://arxiv.org/abs/2102.05126


DB queried here
output tokens
generate state
generate system output
train { destination Cambridge } <db>
There are over 1,000 trains to [destination] . Where will you be departing from ? <eos>
(output ignored)
(out. ign.)
Transformer layers
embeddings
<usr> I'm looking for a train to Cambridge
<ds> train destination Cambridge }
<db> train 1113 matches
<sys> There are over 1,000 trains to [destination] . Where will you be departing from ?
user input
prev. state toks.
DB output
previous output tokens
input tokens

# Performance

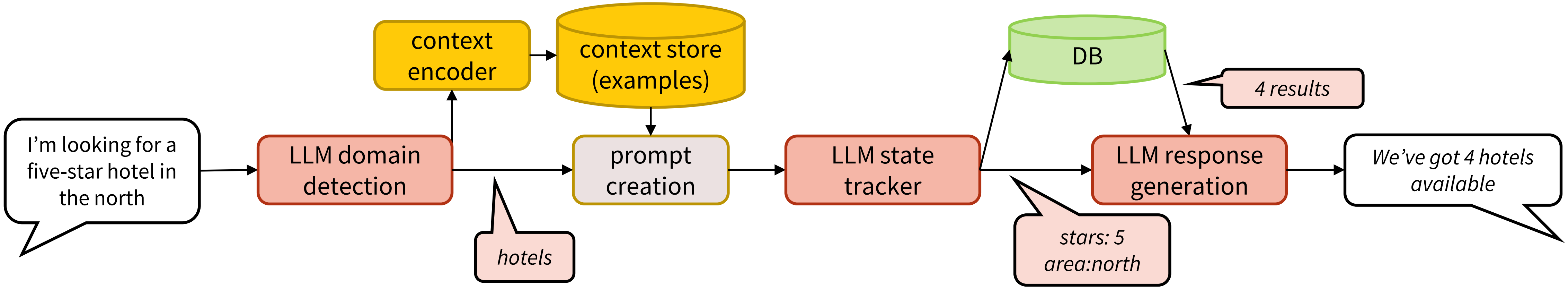
- Dialogue success (=user gets what they wanted)
  - crowdsourcing: 62%
  - experts trying hard: 87%
- **Hallucinates** sometimes
  - may generate factually incorrect outputs, hard to control
  - → data cleaning, consistency checks
- **Needs a lot of data** & annotation (MultiWOZ = 10k)
  - costly, may be noisy
  - → **LLM prompting**?

# Dialogue with LLMs

(Hudeček & Dušek, 2023)
https://aclanthology.org/2023.sigdial-1.21

| | |
|---|---|
| instruction | Definition: Capture values from a conversation about hotels. Capture pairs “entity:value” separated by colon and no spaces in between. Separate the “entity:value” pairs by hyphens. Values that should be captured are: |
| domain description | - “pricerange”: the price of the hotel<br>- “area”: the location of the hotel<br>… |
| examples | --- Example 1 ---<br>…<br>--- |
| dial. history | Assistant: “Hello, how can I help you?”<br>… |
| user input | Customer: “I am looking for a five-star hotel in the north” |

- How good are LLMs if we require structure?
  - slots / DB are given
  - no finetuning ~ **prompting only**
  - ChatGPT, Tk-Instruct, Alpaca… (7-20B params)
- A few examples in prompt (context store)
  - wide application potential
- Still the same idea: **context** → **state** → **DB** → **response**
  - additional step needed: domain detection



# Results

- Domain detection – very good, ChatGPT almost perfect
- Belief tracking – pretty bad
  - much worse than SotA
  - ChatGPT best: ~50% F1, others 7-33%
- Responses: OKish
  - experts trying hard: 76% ChatGPT, 64% TkInstruct
- More potential with prompt engineering

# Conclusions

- LLMs are powerful & can work well…
  - if you provide data on the input
  - if you optimize your prompts
- So far, ChatGPT/GPT4 are better than open LLMs
  - new LLMs coming up every week (Llama2, Falcon, Mistral, …)
  - OpenAI closed models may have seen some of the data (~is it zero-shot?)

## Future work

- look into data leakage effect
- more transparency ~ prompting, interpretable latents
- constraining – alignments, decoding-time "critic"

(Lango & Dušek, 2023)
https://arxiv.org/abs/2310.16964

# Thanks

## Contacts:

**Ondřej Dušek**
**odusek@ufal.mff.cuni.cz**
**https://tuetschek.github.io**
**@tuetschek**

## Thanks:

Vojtěch Hudeček

Jonáš Kulhánek

Tomáš Nekvinda

## Links

**These slides:** **https://bit.ly/resp-ai-od**

Papers: http://arxiv.org/abs/2102.05126
https://aclanthology.org/2023.sigdial-1.21

Supported by: European Research Council (ERC StG No. 101039303 NG-NLG), Charles University projects GAUK 302120, 373921, SVV 260698,
TA ČR TL05000236 AI asistent pro žáky a učitele.
Using LINDAT/CLARIAH-CZ Research Infrastructure resources (Czech Ministry of Education, Youth and Sports project No. LM2018101).