

Transforming Unstructured Data to Structured: Standards, Logic, and Language Models

LaCo track | ESSLLI 2025 | Bochum

Zuzana Nevěřilová

Faculty of Arts, Masaryk University, Brno, Czechia

Day 5 | Aug 1, 2025

Connecting the two worlds

Strengths and Weaknesses

LLMs are very good at language generation. → They are understandable.

Logics can check consistency and describe facts precisely. → They are correct.

Understandable and correct?

LLMs and KGs

- can LLMs reason?
- LLMs help to build KGs
- KGs prevent LLMs' hallucinations
- LLMs are used to query KGs

LLMs and Reasoning

Ask LLMs to Reason

- LLMs have internal knowledge (encoded in their weights)
 - this knowledge is gained through consumption of huge amounts of texts
 - the knowledge is limited by the model cutoff date
- reasoning models perform a series of *thinking* steps
- sometimes, models are able to express [their uncertainty](#)
- specifically fine-tuned LLMs for logical reasoning exist (e.g., DeepMind [Selection Inference](#))

[Awesome LLM Reasoning Collection](#)

Example of LLM reasoning

Question: Verna loves to eat fruit. She bought three apples at \$1.50 each, five oranges at \$0.80 each, and six peaches at \$0.75 each. If she gave \$20, how much change did she receive?

Answer:

three apples cost $3 \times \$1.50 = \$\langle\langle 3 \times 1.5 = 4.50 \rangle\rangle 4.50$.

five oranges cost $5 \times \$0.80 = \$\langle\langle 5 \times 0.8 = 4 \rangle\rangle 4$.

four peaches cost $6 \times \$0.75 = \$\langle\langle 6 \times 0.75 = 4.50 \rangle\rangle 4.50$.

thus, verna paid a total of $\$4.50 + \$4 + \$4.50 = \$\langle\langle 4.5 + 4 + 4.5 = 13 \rangle\rangle 13$.

therefore, verna received $\$20 - \$13 = \$\langle\langle 20 - 13 = 7 \rangle\rangle 7$.

Example from [Kadavath et al., 2022](#)

LLM Overthinking

ollama run deepseek-r1 "Hello, world!"

Thinking...

First, the user said “Hello, world!” which is a common introductory phrase in programming. It’s often used to mark the beginning of a program or as a simple greeting.

As an AI assistant, I should respond helpfully and engagingly. Since this is a basic greeting, my response could be friendly and open-ended to encourage further interaction.

Possible ways to respond: - Acknowledge the greeting: Say hello back. - Be enthusiastic: Use an exclamation point or positive language. - Add context: Remind them that I’m here to help with various topics. - Ask a question: To keep the conversation going, like what they’re interested in.

I should also consider my identity. The system prompt says “You are a helpful AI assistant.” So, I need to stay true to that role.

The phrase “Hello, world!” might be referencing the famous first program in many languages that prints this out. But since it’s just saying hello, not asking for help with programming, I shouldn’t assume that unless they specify. Keep it simple and friendly. A good response could be: - Greet them back. - Introduce myself briefly. - Offer assistance or ask how I can help.

For example: “Hello! I’m your AI assistant here to help you with any questions or tasks. What’s on your mind today?” This maintains a helpful tone, confirms my role, and invites further conversation. If they meant it literally as a greeting in real life, I should respond accordingly but keep the digital context clear.

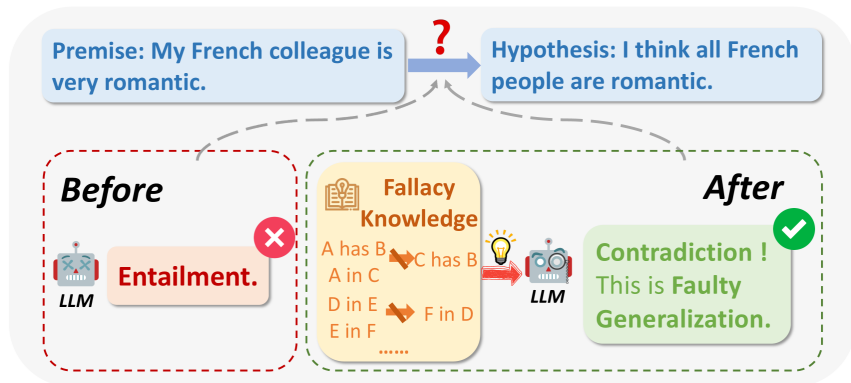
...done thinking.

Hello! That’s a great way to start—nice to meet you!

Limitations

It's very challenging for current LLMs to perform complex reasoning:

- multi-hop reasoning
- know their reasoning limits
- overlooking of logical fallacies (e.g., [Li et al., 2024](#))



A Promising Approach: Circuit Tracing

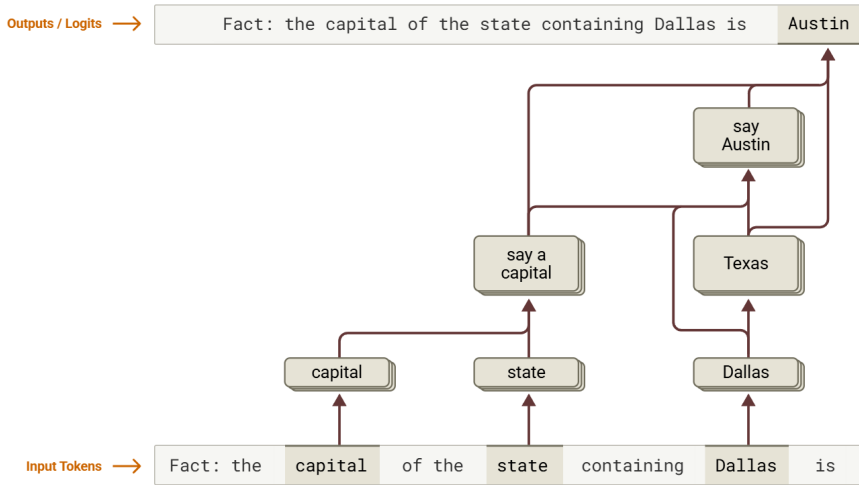


Image from [Lindsey et al., 2025](#)

LLMs build KGs

LLMs on their Way from Natural Language to Description Logic

Model “Sisters reunited after 10 years at the checkout line in Safeway”

- create/update the knowledge base
 - does the representation reflect common understanding?
 - how to represent implicit knowledge (i.e., not present in the input data)?
 - what existing ontologies/vocabularies can be used?

Natural Language → Formalization

The holy grail of the Natural Language Processing (NLP).

Historically:

- formal grammars
- named entity recognition and linking
- information extraction
- ...

Even with the LLMs, it's not easy. [ChatGPT conversation](#) —

Challenges

- proceed with the **default reading** only (solve ambiguity)
- force the model to follow the modeling **design choices**
- prevent the model to **fabricate** the facts

**KGs support LLMs with formalized
knowledge**

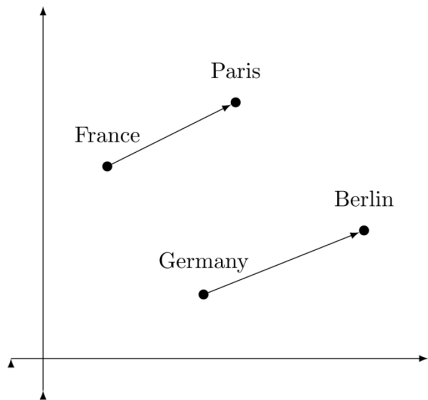
Retrieval Augmented Generation (RAG) prevents Hallucinations

RAG is a method to:

- provide the up-to-date knowledge to the model
- provide context specific knowledge
- prevent hallucinations

Side Note: Vector Embeddings

- vector = sequence of numbers
- length of the sequence = **dimension**
- semantic vectors encode meaning



Retrieval Augmented Generation - RAG

- **Retrieval** = knowledge is encoded so that the model understands it
 - chunks of knowledge are encoded into **embedding vectors** and stored in a database
 - at query time, RAG compares the query embedding vector with the database and answers with the most similar chunk
- The difficult part(s):
 - how to make the chunks
 - is the encoding part correct?
 - is the vector comparison the best way?
- Generation = the easier part for LLMs

RAG: Retrieval Example

- chunking is based on chunk length, text structure, etc.

firstName	lastName	age	city
Jane	Doe	26	Bochum
John	Smith	25	Bochum

Example of suboptimal chunking:

	Doe	26	Bochum
John			

How old is John?

Combine Knowledge Graphs with RAG

- encode *correct* chunks of knowledge - the KG statements

→ Graph RAG

- KG triples to sentences (chunks of knowledge)
- KG provides relevant subgraphs to answer the query
- LLM converts the query into a graph query

LLMs asks the KGs

Data Querying with LLMs

LLMs can produce SPARQL queries:

- no need to learn SPARQL (really?!?)
- different wordings for the same question
- let's try this in the hands-on session

Summary

- LLMs have reasoning capabilities.
- Knowledge graphs have advanced reasoning, but they are difficult to build and maintain.
- Hybrid approaches can benefit from LLMs' good ability to generate:
 - LLMs can convert text to triples (to some extent)
 - LLMs can query KGs (generate the SPARQL queries)
 - LLMs can express the facts in the KG in natural language
- KG can support the retrieval in RAG.
- Logic inside KG can check consistence and correctness of the answers.

Conclusion

Back to the initial question (from the *Fountains of paradise*):

“What was the rainfall in the capital of the world’s smallest national state on the day when the second largest number of home runs was scored in college baseball?”

Currently, this question cannot be answered by a simple web search nor a large language model.

However,

- there are ways how to formalize knowledge about rainfall, geography, and sports
- there are standards how to link such knowledge
- there are ways how to query a knowledge base
- ... even with the natural language

During the Course

Let's check whether you:

- learned what **knowledge graphs** are
- realize what is the difference between tabular and graph data and how the latter can be obtained from the former
- recognize some parts of the large idea of the **Semantic Web** during your web searches
- know what *good* data mean
- understand that **description logic** is a subset of first order predicate logic
- understand how can knowledge be stored in RDF/OWL **triples**
- know what type of new knowledge can be **inferred** in knowledge graphs
- are able to query huge knowledge bases such as DBpedia or Wikidata
- see advantages and disadvantages of LLMs in reasoning
- know about the good (and bad) things in both worlds