Analysis and development of a prompt engineering ontology

Graduating

- Simone Gramegna

Advisor

- Prof. Claudia d'Amato

Co-advisors

- Dr. Roberto Barile
- Dr. Andrea Nuzzolese

What is the context?

Context

- LLMs are advanced AI models, capable of executing different tasks
- Most popular LLMs: GPT, Gemini, Mistral and DeepSeek
- Prompt engineering involves design and optimization of prompts, producing the best possible response from an LLM
- Different prompting techniques have been developed

Main issue

- Fragmented resources on prompt engineering and LLMs
- Solution: a resource providing a complete perspective

Prompt Engineering Ontology

- Prompt Engineering Ontology (PEO): new resource for prompt engineering and LLMs
- PEO represents widely used concepts
- Linked Open Terms (LOT) methodology used in the development
- PEO is encoded in OWL
- PEO is available on Github, W3id and Bioportal
- Compliance with FAIR principles in Open science

- https://github.com/simonegramegna/peo
- http://w3id.org/peo
- https://bioportal.bioontology.org/ontologies/PEO_ONTOLOGY?p=summary

1. Does PEO provide comprehensive and consistent knowledge on LLMs and prompt engineering?

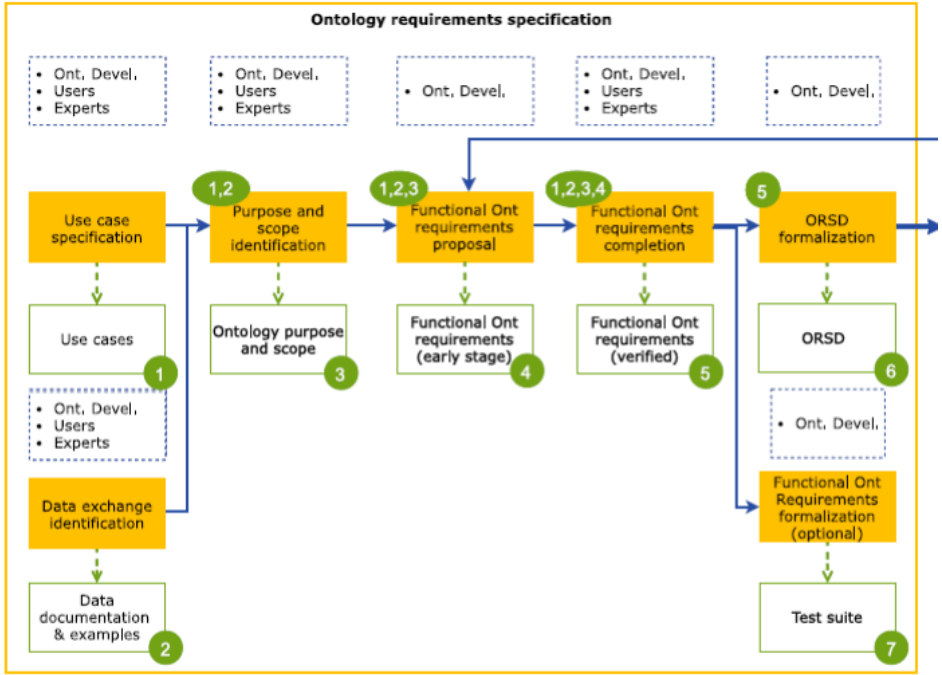
2. Can we use PEO to infer additional knowledge?

Design

- 10 use cases defined
- 10 data sources (papers, websites, repositories) identified
- Scope and purpose specification
- 16 competency questions created
- 3 non functional requirements identified

What is the output?

- Ontology Requirements Specification Document (ORSD)



Ontology requirements specification workflow

Competency questions

- Competency questions (CQs) define an ontology from its functional perspective
- CQs cover main concepts in the ontology
- Good set of CQs responds to user's real questions
- Definition of a conceptual scheme starting from CQs
- CQs will be employed during the evaluation

1. What is prompt engineering?
2. What is a prompt?
3. What are prompting techniques?
4. What are image prompting techniques?
5. What are code prompting techniques?
6. Which task does a prompt solve?
7. Which prompts are generated using a prompting technique?
8. What are the responses that follow each prompt?
9. What are possible tasks?
10. Which tasks are related to the text?
11. What is a chat?
12. What is a large language model?
13. What types of large language models are available?
14. What are large language models architectures?
15. What are large language models capabilities?
16. What companies develop large language models?

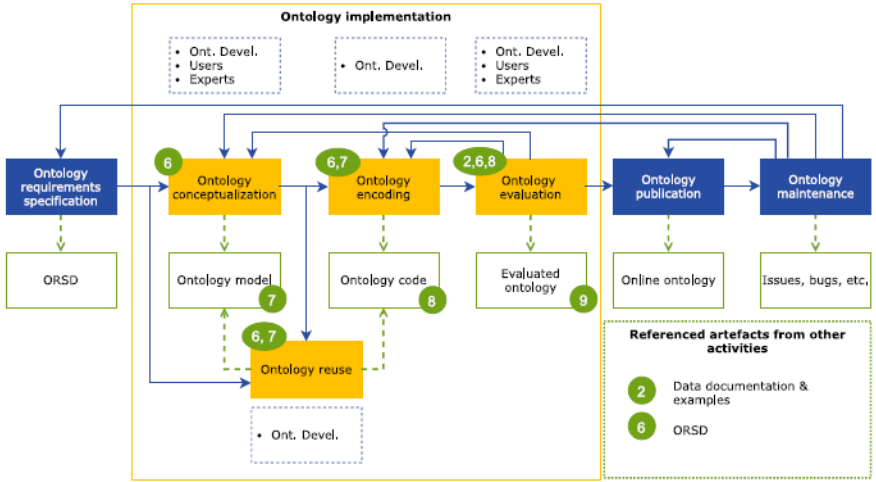
Conceptualization

Conceptualization phase:

- Exploration of possible reusable ontologies
- Definition of entities and relations
- Application of ontology design patterns

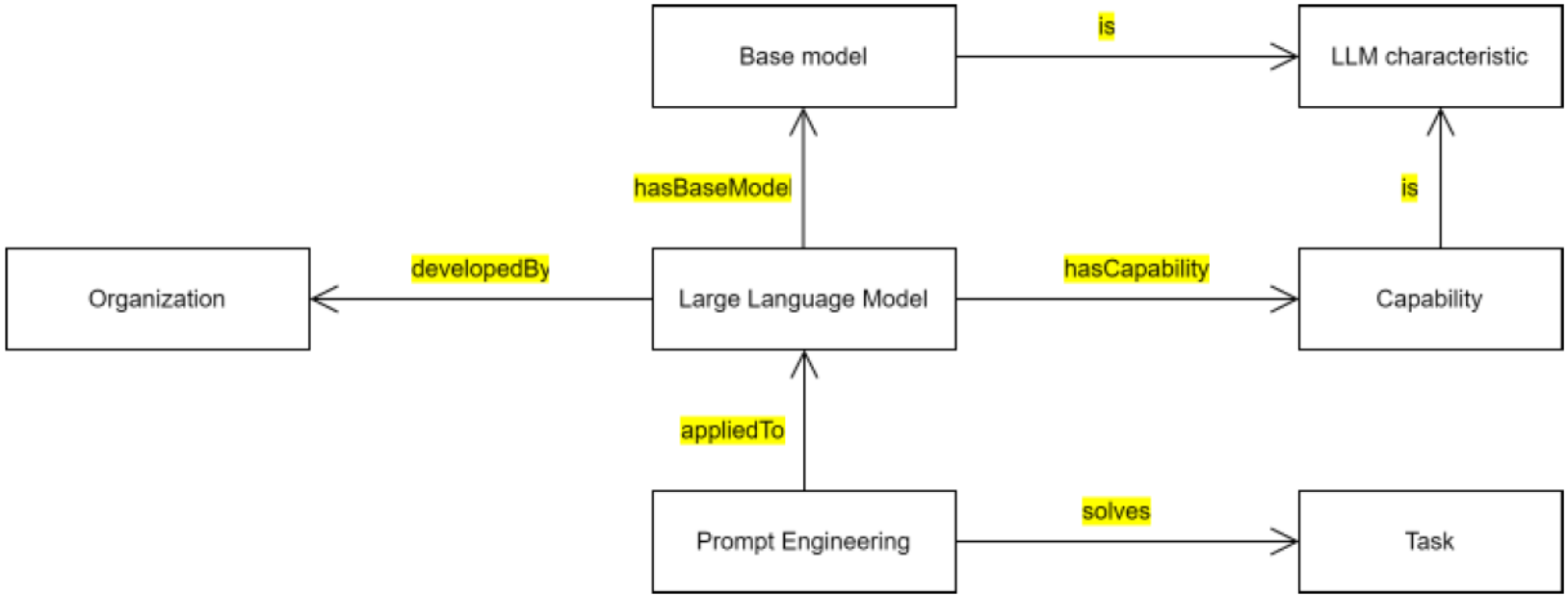
Ontology design patterns adopted

- Description pattern
- Sequence pattern
- Task execution pattern



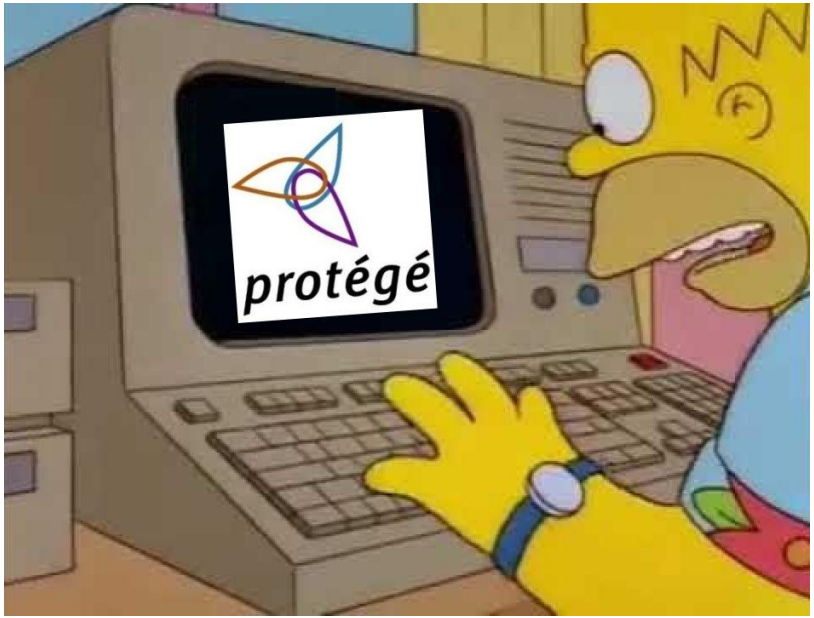
Ontology implementation workflow

Main concepts



Implementation

- Implementation of the defined scheme
- Protegé editor
- Definition of every class, object property and data property
- Inference rules (SWRL rules), inferring relations between instances
- Documentation using OWLDoc

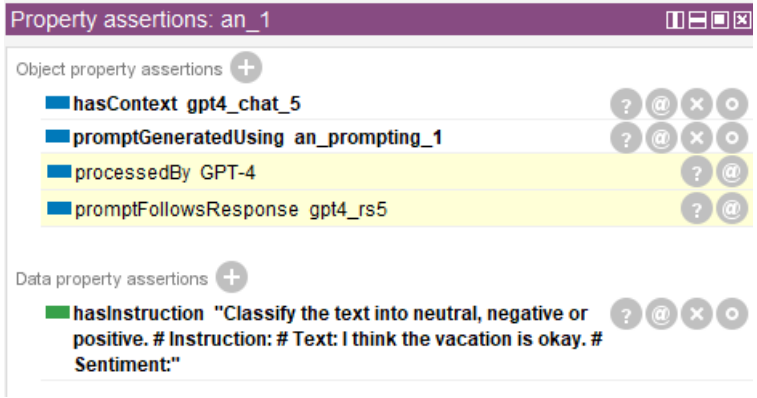


What is the output?

- PEO encoded in OWL

Entities population

- Instances creation of Large Language Model and Prompting technique classes
- Prompting techniques population: zero-shot prompting, few-shot prompting, emotion prompting, role prompting and analogical prompting
- Connection with other entities



Prompt individual with object properties and data properties

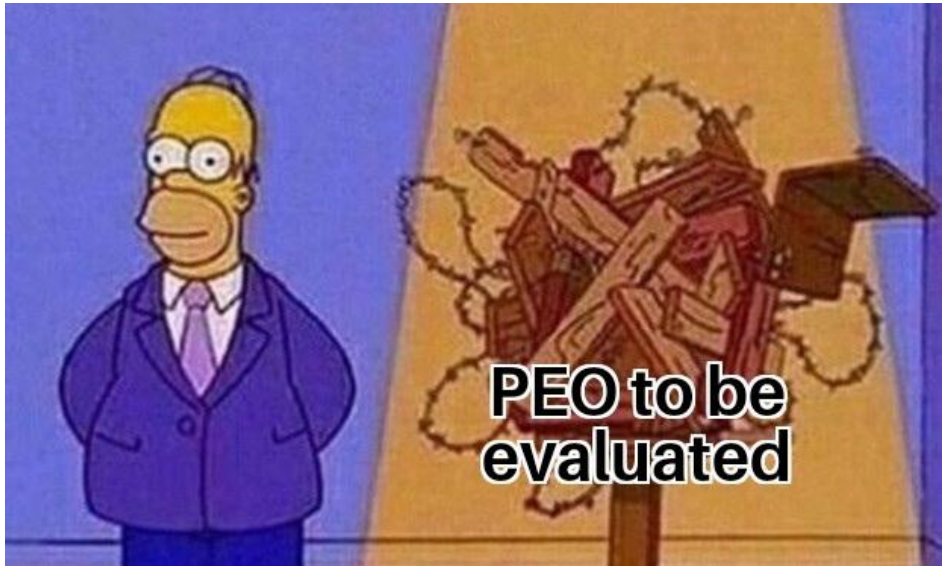
Ontology population using GPT-4

- Extends the original ontology
- Input: competency questions and PEO
- Output: new version of PEO

Evaluation setting

Evaluation on four dimensions, ensuring PEO quality.

- Ontology consistency check
- Standard metrics
- Static validation
- CQs addressing



What is the target?

- PEO evaluation

Consistency and metrics

Ontology consistency check

- HermiT reasoner execution
- No inconsistencies found

OntoMetrics

- Base metrics: count elements like classes, properties, and individuals
- Schema metrics: evaluate schema structure and hierarchy
- Knowledge base metrics: measure knowledge density and quality
- Class metrics: analyze class connectivity and detail
- Graph metrics: assess graph connectivity and centrality

- http://www.hermit-reasoner.com/
- https://ontometrics.informatik.uni-rostock.de/ontologymetrics/

Metrics

Property	Value
Axioms	2684
Logical axioms count	1695
Class count	126
Total classes count	126
Object property count	34
Total object properties count	34
Data property count	13
Total data properties count	13
Properties count	47
Individual count	352
Total individuals count	352
DL expressivity	SRIF(D)

Metric	Value
Attribute richness	0.103175
Inheritance richness	1.579365
Relationship richness	0.160338
Attribute class ratio	0.0
Equivalence ratio	0.007937
Axiom/class ratio	21.301587
Inverse relations ratio	0.390244
Class/relation ratio	0.531646

Metric	Value
Absolute root cardinality	7
Absolute leaf cardinality	109
Absolute sibling cardinality	126
Absolute depth	318
Average depth	2.52381
Maximal depth	4
Absolute breadth	126
Average breadth	7.0
Maximal breadth	33
Ratio of leaf fan-outness	0.865079
Ratio of sibling fan-outness	1.0
Tangledness	0.269841
Total number of paths	126
Average number of paths	31.5

Metrics

Class Axiom Type	Count
SubClassOf axioms count	199
Equivalent classes axioms count	1
Disjoint classes axioms count	3
GCICount	0
HiddenGCICount	0

Object Property Axiom Type	Count
SubObjectPropertyOf axioms count	33
Equivalent object properties axioms count	0
Inverse object properties axioms count	16
Disjoint object properties axioms count	0
Functional object properties axioms count	0
Inverse functional object properties axioms count	0
Transitive object property axioms count	2
Symmetric object property axioms count	0
Asymmetric object property axioms count	0
Reflexive object property axioms count	0
Irreflexive object property axioms count	1
Object property domain axioms count	33
Object property range axioms count	33
SubPropertyChainOf axioms count	0

Data Property Axiom Type	Count
SubDataPropertyOf axioms count	12
Equivalent data properties axioms count	0
Disjoint data properties axioms count	0
Functional data property axioms count	7
Data property domain axioms count	11
Data property range axioms count	12

Individual Axiom Type	Count
Class assertion axioms count	352
Object property assertion axioms count	390
Data property assertion axioms count	580
Negative object property assertion axioms count	0
Negative data property assertion axioms count	0
Same individuals axioms count	0
Different individuals axioms count	0

Annotation Axiom Type	Count
Annotation axioms count	5
Annotation assertion axioms count	459
Annotation property domain axioms count	0
Annotation property range axioms count	0

OOPS! validation

Three pitfalls found in PEO:

- P04: Unconnected ontology elements
- P34: Untyped class
- P41: No license declared

OOPS! Pitfalls resolution

- P04: connected unconnected classes
- P34: Substituted SWRL rules with property chains
- P41: Declared license (CC-BY-4.0)

OOPS! main page

SPARQL queries

- Translation of the 16 competency questions into SPARQL queries used as unit test
- State-of-art approach to verify whether PEO meets the requirements
- Goal: check whether PEO contains the expected content
- Execution of SPARQL queries using python and jupyter notebook
- Comparison of the output of each SPARQL query with the expected result

CQ1: What is prompt engineering?

```
[41]: cq1 = """
SELECT DISTINCT ?property ?value
WHERE {
    <https://w3id.org/peo#PromptEngineering> ?property ?value .
}
"""

results_cq1 = execute_query(peo, cq1)
print_results(results_cq1)
```

CQ1 SPARQL query

```
[4]:
```

	property	value
0	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Class
1	http://www.w3.org/2000/01/rdf-schema#comment	Practice of crafting and optimizing input prom...
2	http://www.w3.org/2000/01/rdf-schema#label	Prompt engineering

CQ1 SPARQL query result

Research questions

1. Does PEO provide comprehensive and consistent knowledge on LLMs and prompt engineering?

Yes, PEO formalizes key concepts, relationships and applications of LLMs, prompting techniques and related concepts in a structured ontology.

2. Can we use PEO to infer additional knowledge?

Yes, PEO allows inferring additional knowledge by reasoning over LLMs and their connections with prompts and prompt engineering techniques.

Conclusions

- PEO conceptualizes LLMs and prompt engineering
- The conceptualization is machine-processable

Future developments

- PEO-based AI agent
- Automatic population using LLMs
- Integration with external ontologies
- PEO ontology web application

Thanks for the attention!

Simone Gramegna


ANY QUESTIONS?