

Specification	Implication
<i>Accurate, stable grounding</i>	<i>Return OMOP-standardised terms without hallucination or unpredictable model output</i>
<i>Respect OMOP-style constraints</i>	<p><i>Generalisable methods to enforce CDM conventions (stronger than just enum lists)</i></p> <p><i>Specify task-dependent preferences for term sets and concept hierarchies</i></p>
<i>Zero-shot, configurable pipeline</i>	<i>New targets and vocabularies specified declaratively without retraining</i>
<i>Reusable & portable</i>	<i>Support sharing of validated configurations - no re-development required</i>
<i>Operate on standard professional grade machines</i>	<p><i>Must produce workable results with models that can run without dedicated GPU i.e. typically no larger than ~3-7B parameters</i></p> <p><i>High-level abstraction to allow more powerful or bespoke models where resourcing and throughput/reasoning demands allow</i></p>
<i>Run under heavily restricted environments</i>	<p><i>Support linking to locally hosted models, vocabularies & configuration resources</i></p> <p><i>Integrate with securable model hosts</i></p>

Specification	Implication
Accurate, stable grounding	<i>Return OMOP-standardised terms without hallucination or unpredictable model output</i>
Respect OMOP-style constraints	<p><i>Generalisable methods to enforce CDM conventions (stronger than just enum lists)</i></p> <p><i>Specify task-dependent preferences for term sets and concept hierarchies</i></p>
Zero-shot, configurable pipeline	<i>New targets and vocabularies specified declaratively without retraining</i>
Reusable & portable	<i>Support sharing of validated configurations - no re-development required</i>
Operate on standard professional grade machines	<p><i>Must produce workable results with models that can run without dedicated GPU i.e. typically no larger than ~3-7B parameters</i></p> <p><i>High-level abstraction to allow more powerful or bespoke models where resourcing and throughput/reasoning demands allow</i></p>
Run under heavily restricted environments	<p><i>Support linking to locally hosted models, vocabularies & configuration resources</i></p> <p><i>Integrate with securable model hosts</i></p>

Semantically Grounded

Specification	Implication
Accurate, stable grounding	<i>Return OMOP-standardised terms without hallucination or unpredictable model output</i>
Respect OMOP-style constraints	<p><i>Generalisable methods to enforce CDM conventions (stronger than just enum lists)</i></p> <p><i>Specify task-dependent preferences for term sets and concept hierarchies</i></p>
Zero-shot, configurable pipeline	<i>New targets and vocabularies specified declaratively without retraining</i>
Reusable & portable	<i>Support sharing of validated configurations - no re-development required</i>
Operate on standard professional grade machines	<p><i>Must produce workable results with models that can run without dedicated GPU i.e. typically no larger than ~3-7B parameters</i></p> <p><i>High-level abstraction to allow more powerful or bespoke models where resourcing and throughput/reasoning demands allow</i></p>
Run under heavily restricted environments	<p><i>Support linking to locally hosted models, vocabularies & configuration resources</i></p> <p><i>Integrate with securable model hosts</i></p>

Semantically Grounded  **Configurable & Sharable**

Specification	Implication
Accurate, stable grounding	<i>Return OMOP-standardised terms without hallucination or unpredictable model output</i>
Respect OMOP-style constraints	<i>Generalisable methods to enforce CDM conventions (stronger than just enum lists)</i> <i>Specify task-dependent preferences for term sets and concept hierarchies</i>
Zero-shot, configurable pipeline	<i>New targets and vocabularies specified declaratively without retraining</i>
Reusable & portable	<i>Support sharing of validated configurations - no re-development required</i>
Operate on standard professional grade machines	<i>Must produce workable results with models that can run without dedicated GPU i.e. typically no larger than ~3-7B parameters</i> <i>High-level abstraction to allow more powerful or bespoke models where resourcing and throughput/reasoning demands allow</i>
Run under heavily restricted environments	<i>Support linking to locally hosted models, vocabularies & configuration resources</i> <i>Integrate with securable model hosts</i>

Semantically Grounded



Configurable & Sharable



Hostable & Secure

JOURNAL ARTICLE

Structured Prompt Interrogation and Recursive Extraction of Semantics (SPIRES): a method for populating knowledge bases using zero-shot learning

J Harry Caufield  , Harshad Hegde , Vincent Emonet , Nomi L Harris , Marcin P Joachimiak , Nicolas Matentzoglu , HyeongSik Kim , Sierra Moxon , Justin T Reese , Melissa A Haendel ... Show more

Bioinformatics, Volume 40, Issue 3, March 2024, btae104, <https://doi.org/10.1093/bioinformatics/btae104>

Published: 21 February 2024 Article history ▾

 PDF  Views ▾  Cite  Permissions  Share ▾

Abstract

Motivation

Creating knowledge bases and ontologies is a time consuming task that relies on manual curation. AI/NLP approaches can assist expert curators in populating these knowledge bases, but current approaches rely on extensive training data, and are not able to populate arbitrarily complex nested knowledge schemas.

- **Works OK, with some limitations**

- *Brittle parsing of outputs*
- *No OMOP-specific grounding files*
- *Issues working in restricted / airgapped environments*
- *Strict cardinality requirements don't reflect reality of clinical notes*

(N.B. for the most part, these limitations are specific to the implementation, not conceptual. i.e. solvable)

☰ OntoGPT



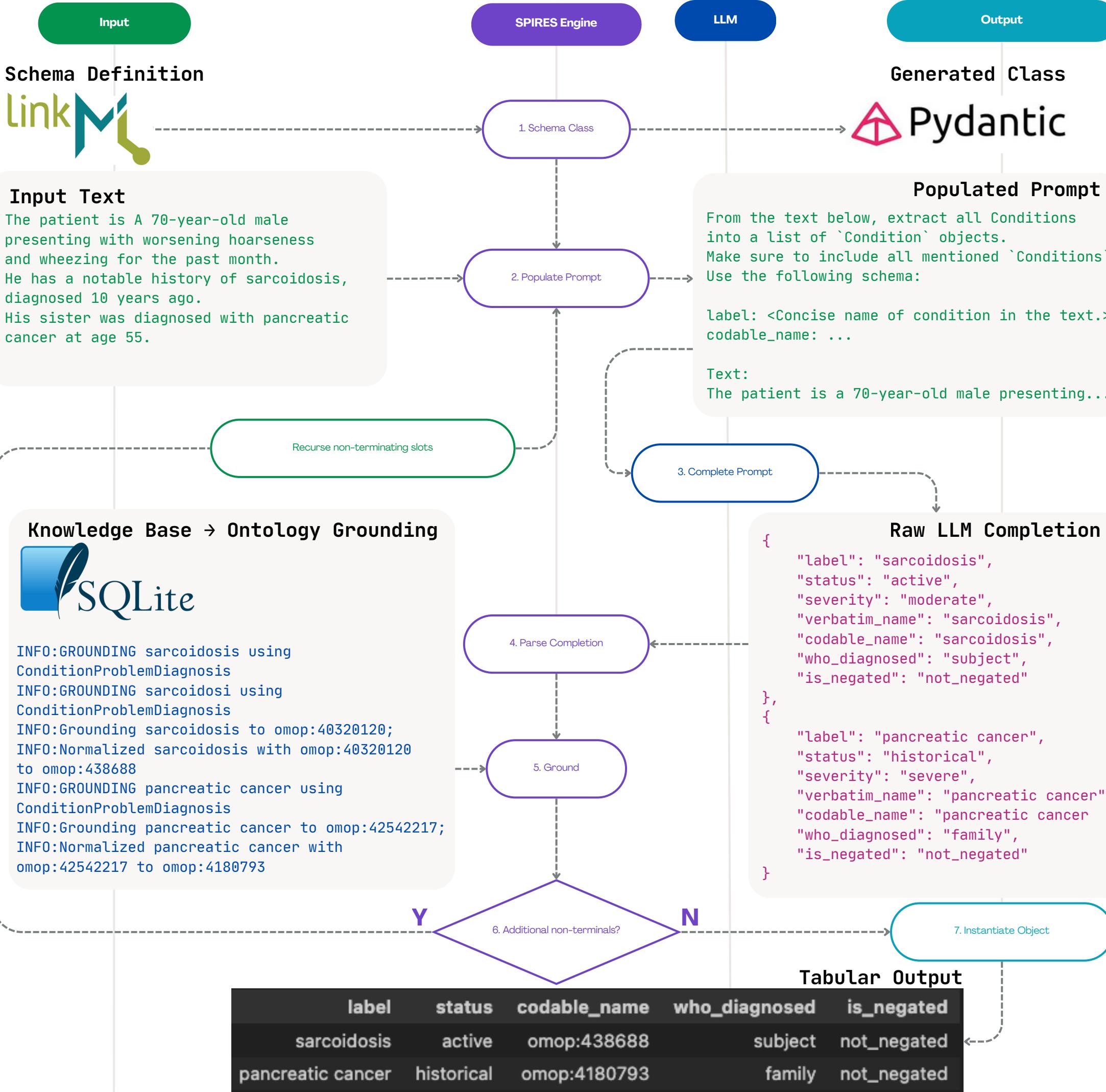
Introduction

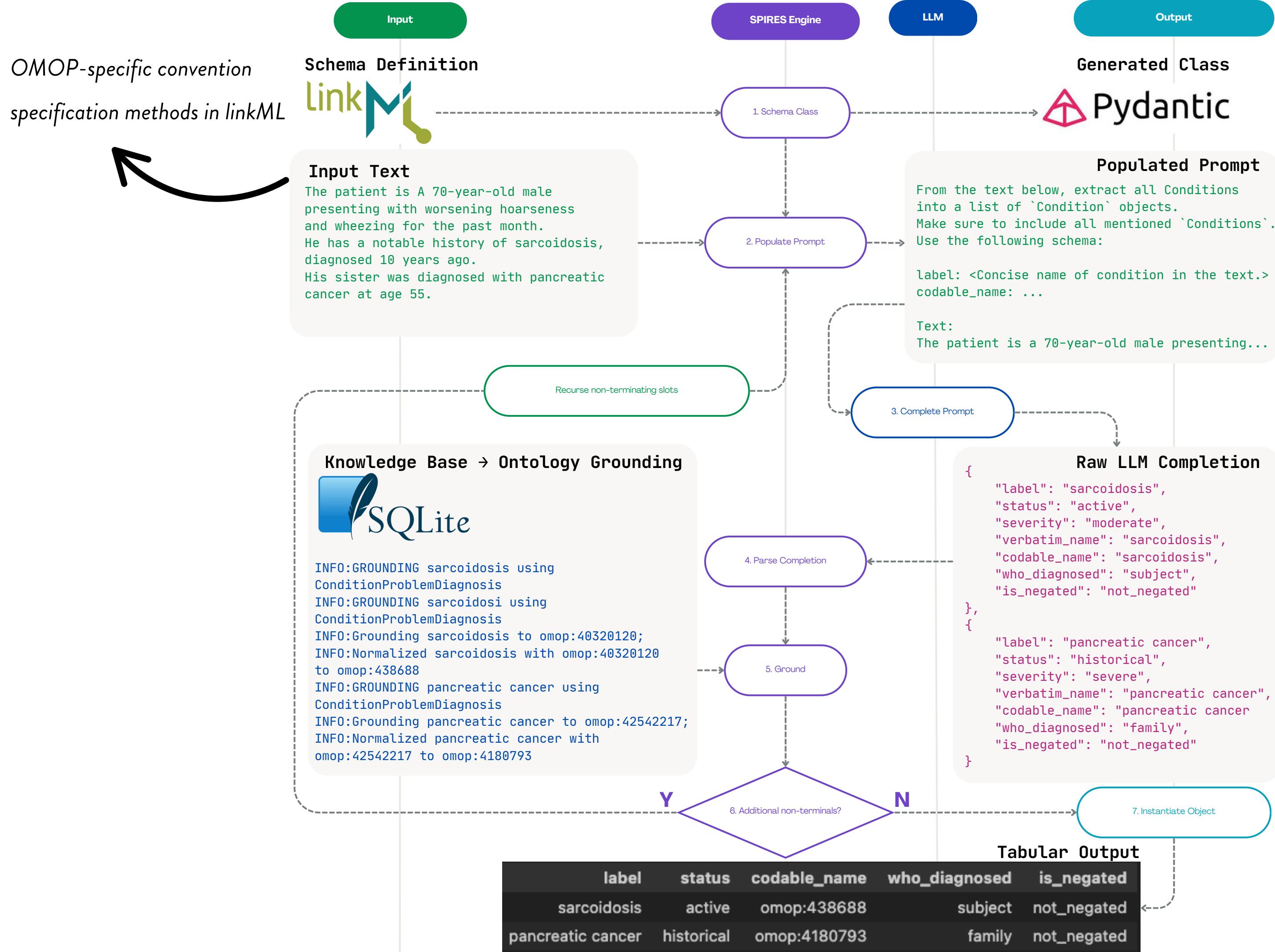
OntoGPT is a Python package for extracting structured information from text with large language models (LLMs), *instruction prompts*, and ontology-based grounding. It works well with OpenAI's GPT models as well as a selection of other LLMs. OntoGPT's output can be used for general-purpose natural language tasks (e.g., named entity recognition and relation extraction), summarization, knowledge base and knowledge graph construction, and more.

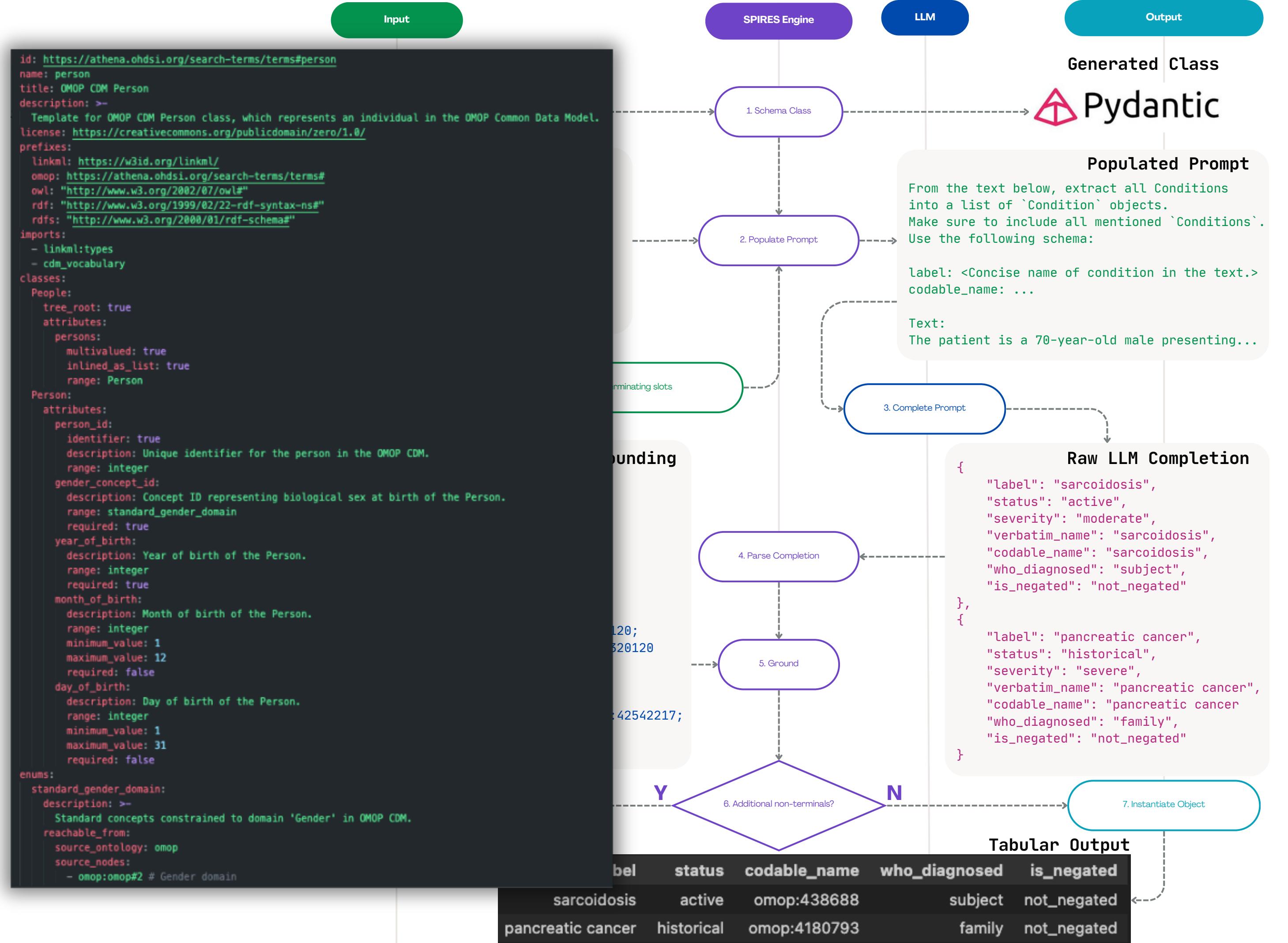
Methods

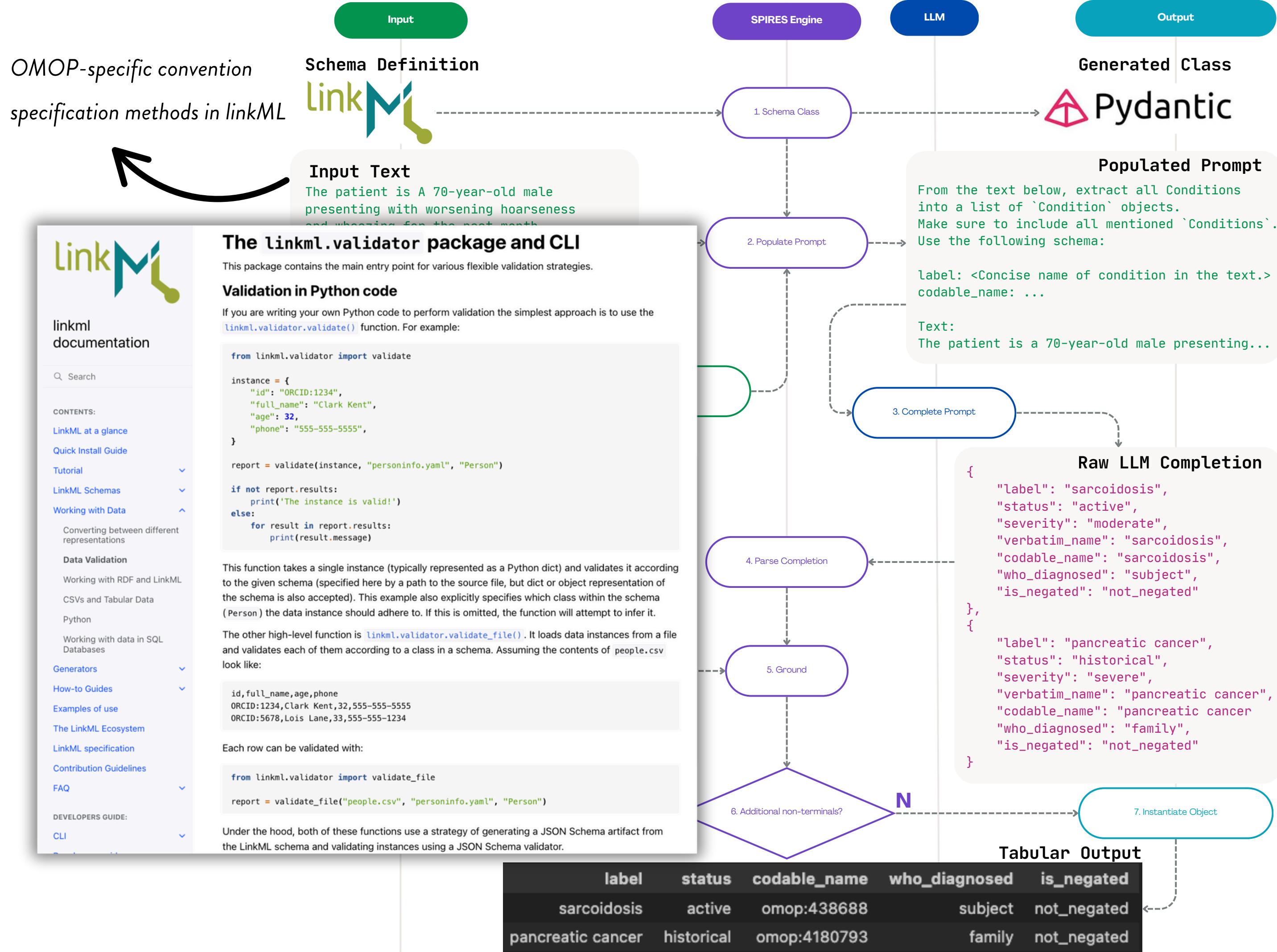
The primary extraction method currently implemented in OntoGPT is SPIRES:

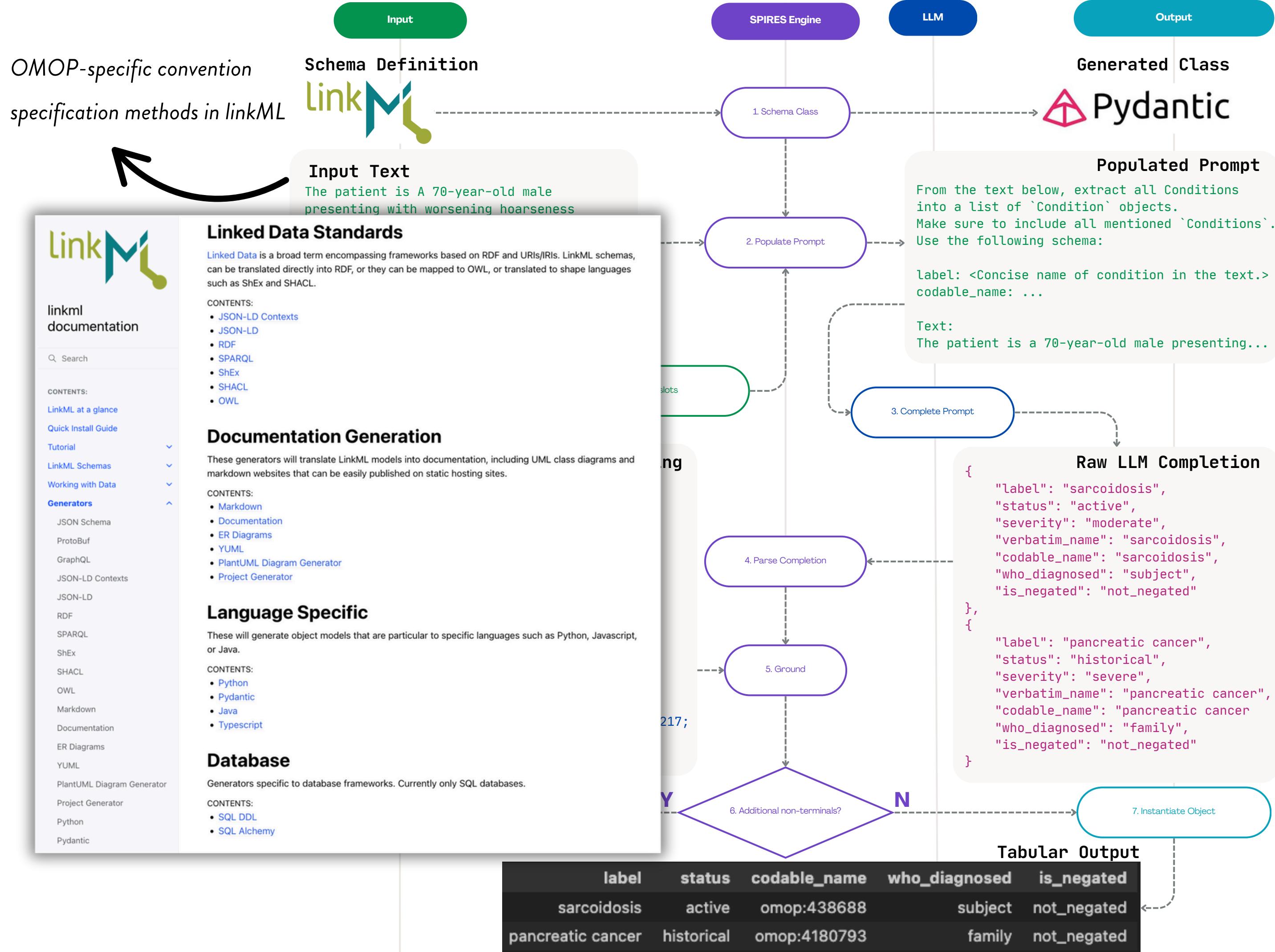
- SPIRES: *Structured Prompt Interrogation and Recursive Extraction of Semantics*
- A Zero-shot learning (ZSL) approach to extracting nested semantic structures from text
- This approach takes two inputs - 1) LinkML schema 2) free text, and outputs knowledge in a structure conformant with the supplied schema in JSON, YAML, RDF or OWL formats
- Uses OpenAI GPT models through their API, or one of a variety of LLMs on your local machine



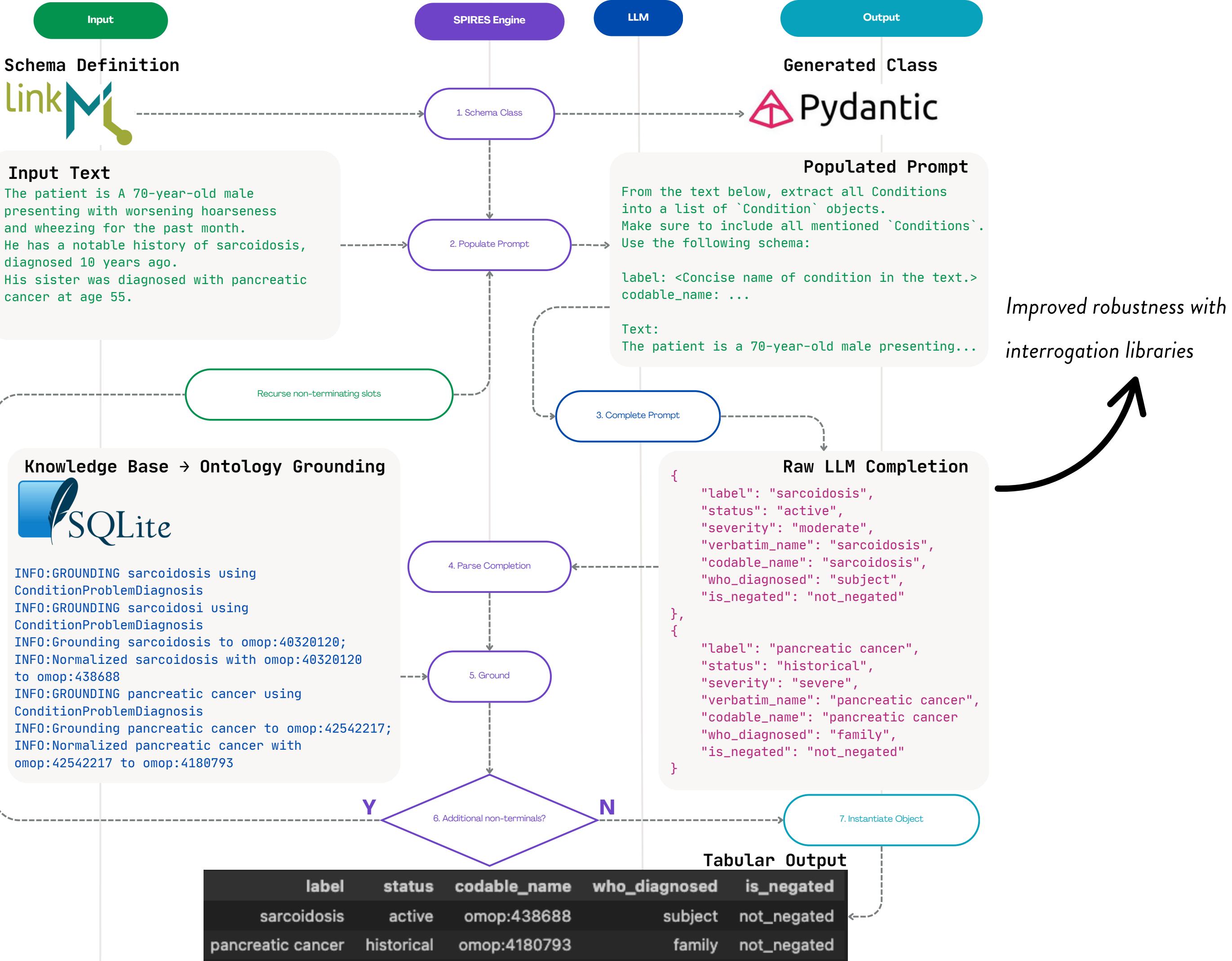


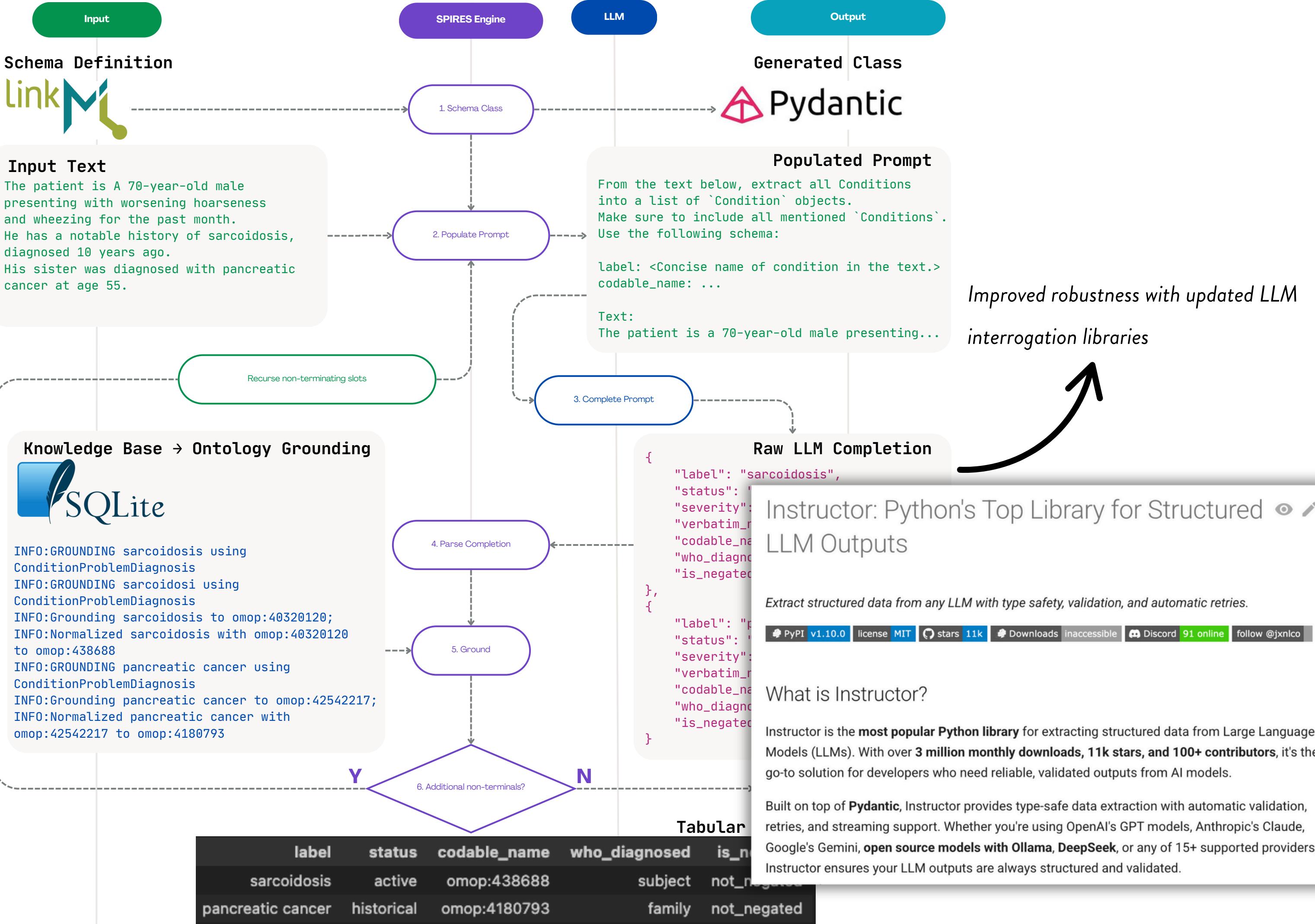


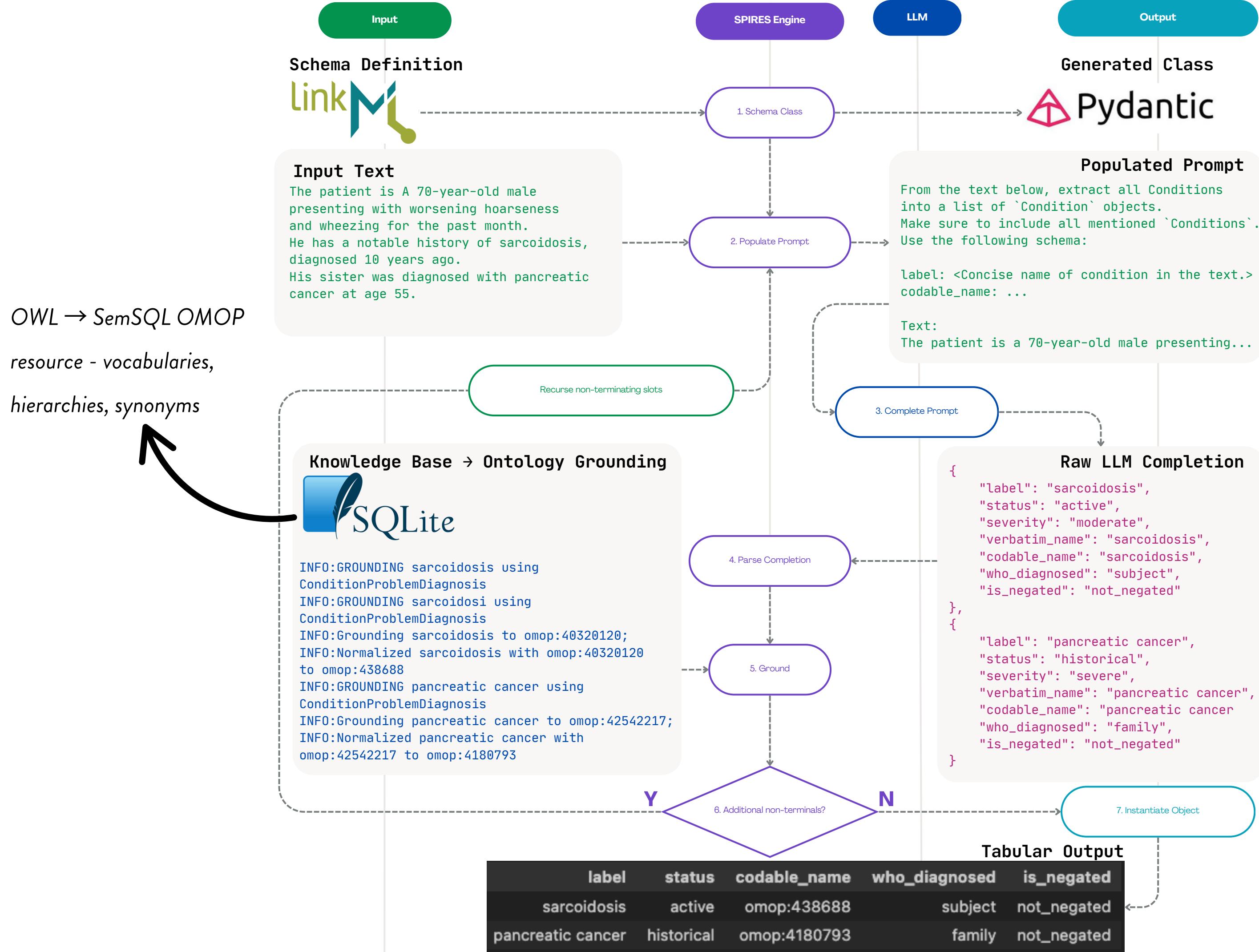


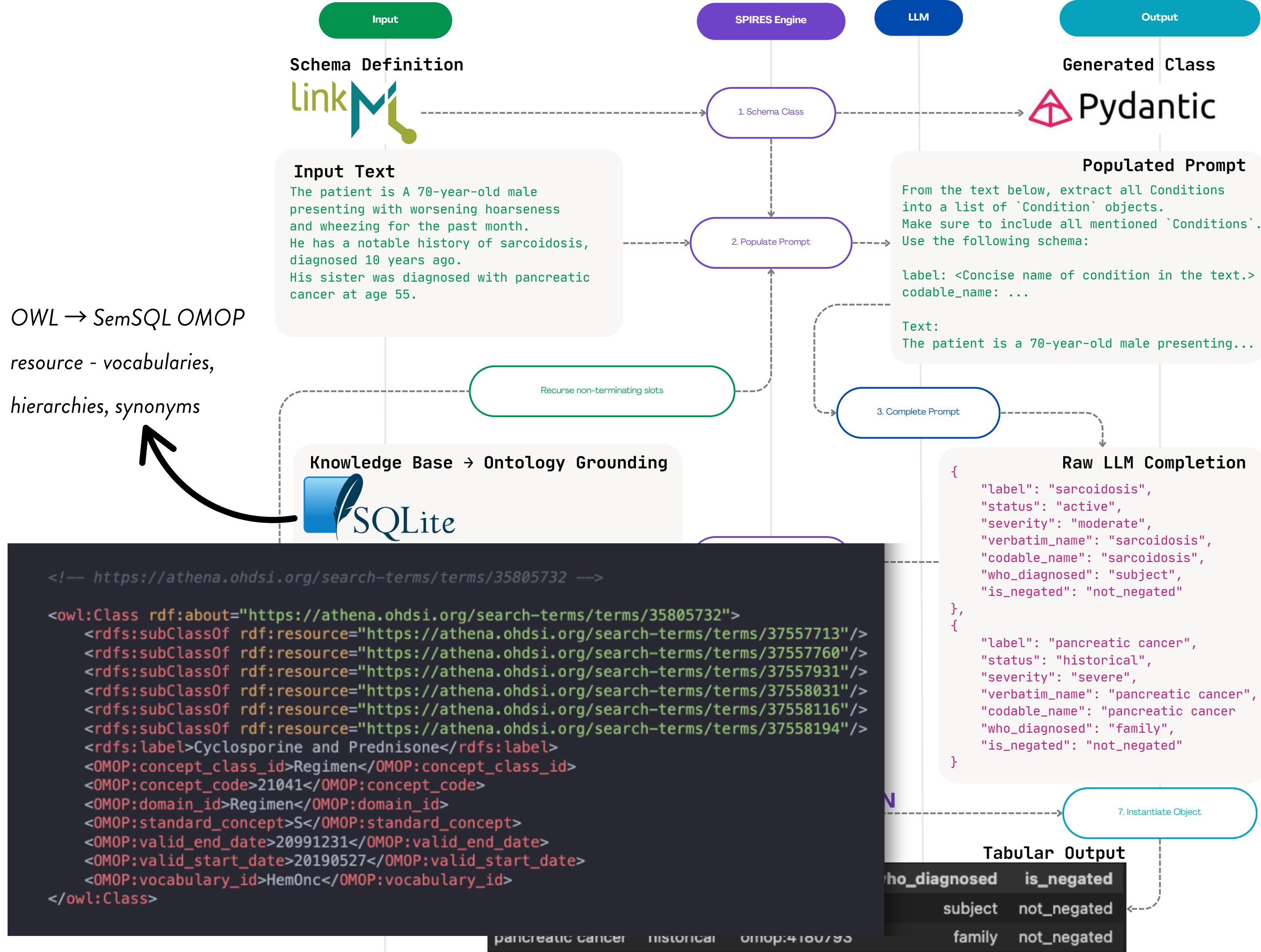


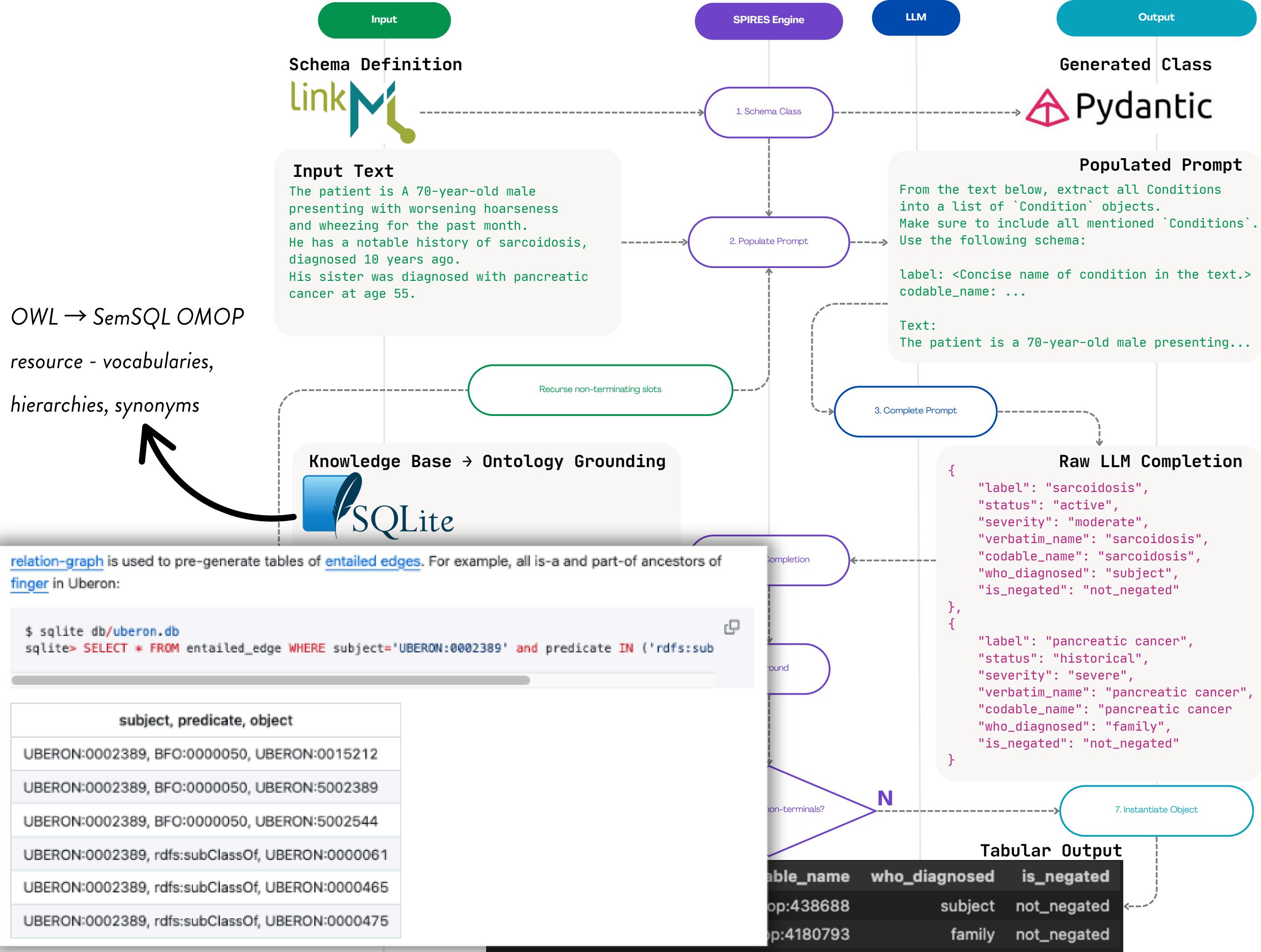


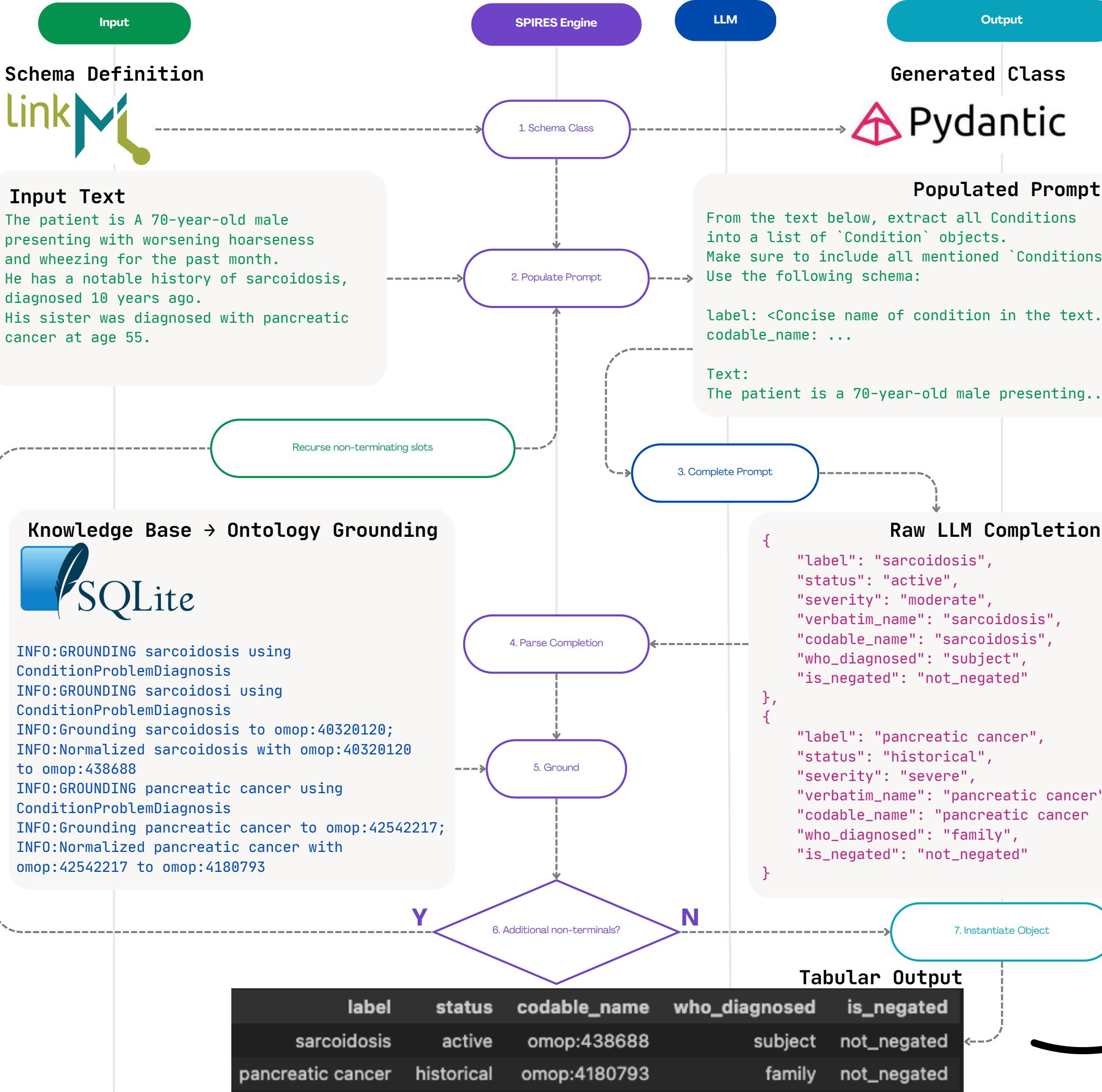




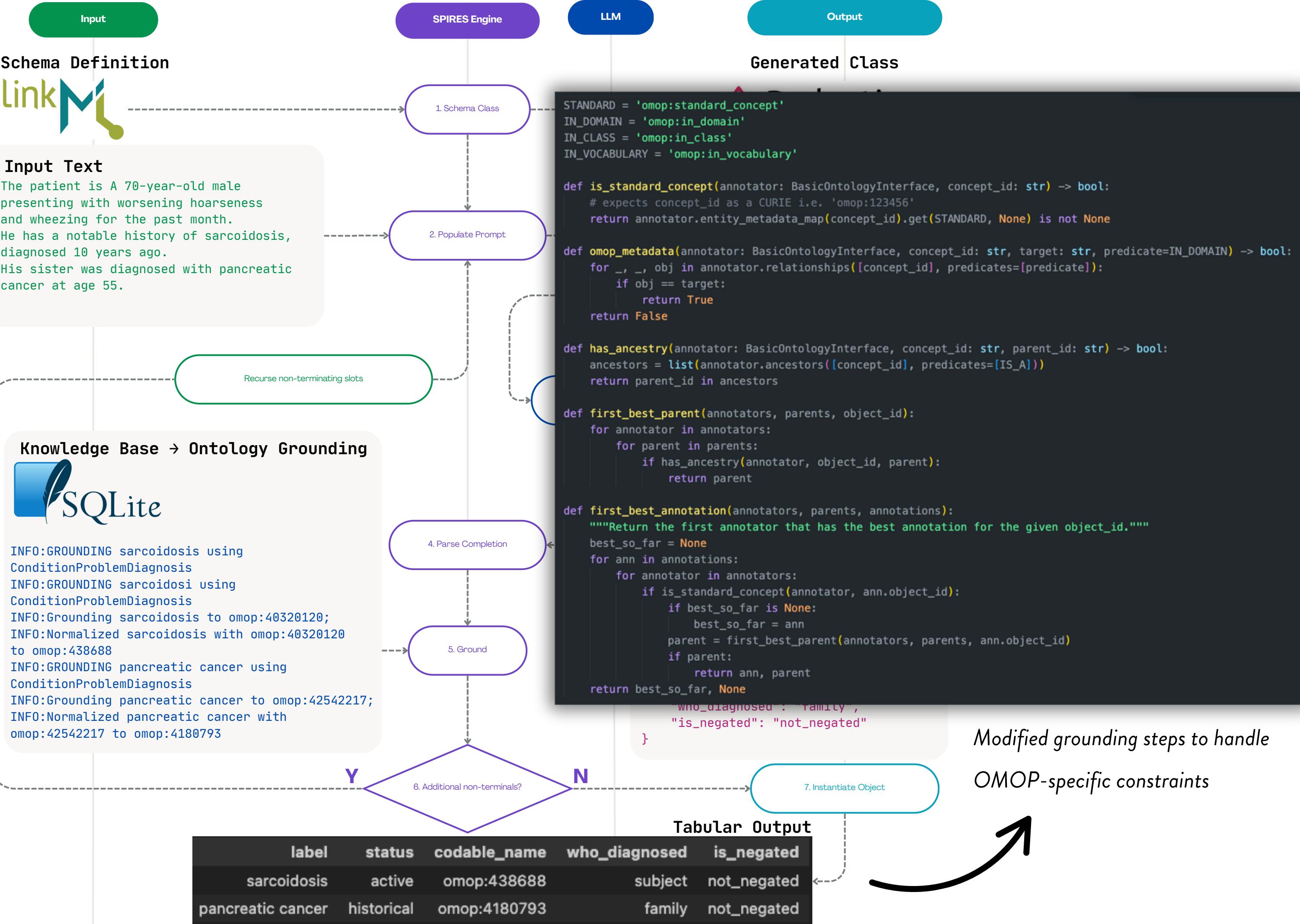






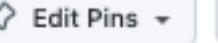
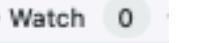


Modified grounding steps to handle
OMOP-specific constraints



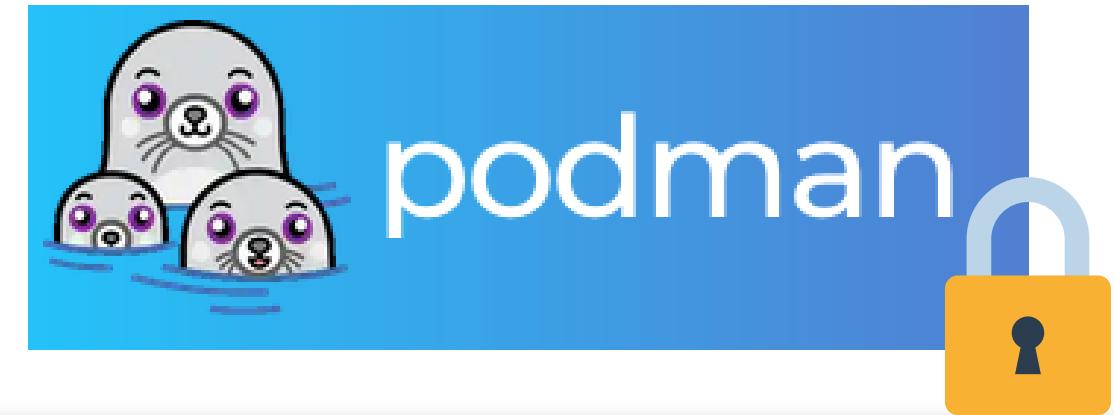
INPUT	llama3 label	IDs	evaluation	usagi label	IDs	evaluation	medspacy label	IDs	evaluation
<i>rectum + pelvis</i>	Pelvic region;Rectum structure	4044034;4144541	correct	PEL	4123163	invalid			no match
<i>(l) lung</i>	Lung structure	4213162	correct	Lung structure	4213162	correct	Entire lung	4111459	correct
<i>ph i prostate+sv</i>	Prostatic structure	4165732	valid	Prostatic structure	4165732	correct			no match
<i>glottis</i>	Glottis structure	4047227	correct	Glottis structure	4047227	correct	Entire glottis	4131315	correct
<i>l chest wall</i>	Chest wall structure	4193513	correct	Chest wall structure	4193513	correct	Entire chest wall	4109932	correct
<i>lt femur</i>	Bone structure of femur	4323581	correct	Bone structure of femur	4323581	correct	Entire bone of femur	37115374	correct
<i>ph1 prostate</i>	Prostatic structure	4165732	correct	Prostatic structure	4165732	correct	Entire prostate	4110208	correct
<i>l/s spine</i>	lumbar	4045660	correct	Structure of vertebral column	4227378	valid	Entire vertebral column	4185891	valid
<i>ph2 prostate bed</i>	Prostatic structure	4165732	correct	Prostatic structure	4165732	correct	Entire prostate	4110208	correct
<i>distal oesophagus</i>	Esophageal structure	4140098	correct	Esophageal structure	4140098	correct	Esophageal structure	4140098	correct
<i>rtbreast+scf+imc+sib</i>	Breast structure	4298444	valid	Breast structure	4298444	correct			no match
<i>prostate + pelvis</i>	Prostatic structure;Pelvic region	4165732;4044034	correct	PEL	4123163	invalid	Entire prostate, Entire pelvis	4110208, 4041832	correct
<i>rt nasal ala</i>			no match	Lateral nasal artery	37157433	invalid			no match
<i>t11-l3</i>	thoracic;lumbar	4047490;4045660	correct	ST11	4159026	invalid	Level of the eleventh thoracic vertebra	4134469	near match
<i>t9-l3</i>	thoracic	4047490	correct	T9-T10 rotator thoracis	4077547	invalid			no match
<i>upper pelvis</i>	Pelvic region	4044034	correct	PEL	4123163	invalid	Entire pelvis	4041832	correct
<i>rt parietal</i>	Brain structure	4133034	correct	Structure of left parietal bone	37158682	near match			no match
<i>right pelvis</i>	Pelvic region	4044034	correct	Structure of right renal pelvis	4184440	invalid	Entire pelvis	4041832	correct
<i>(r) breast/low axilla</i>	Breast structure;Axillary region structure	4298444;4238919	correct	Axillary region structure	4238919	correct	Entire breast	4108283	correct
<i>thyroid</i>	Thyroid structure	4321375	correct	Thyroid structure	4321375	correct	Thyroid structure	4321375	correct

 **grounding_dockerised_mwe** Private

 gkennos gguf dl script	dc02b4f · 15 hours ago	 18 Commits
 docker no do not try build vilm on an m4... yesterday		
 image-service-demo rest of image functionality last week		
 inference-wrapper no do not try build vilm on an m4... yesterday		
 omop-spires-demo works with podman now but not ramalama yet 2 days ago		
 populate_containers gguf dl script 15 hours ago		
 .dockerrcignore works with podman now but not ramalama yet 2 days ago		
 .gitignore gguf dl script 15 hours ago		
 DONTREADME.md no do not try build vilm on an m4... yesterday		
 README.md no do not try build vilm on an m4... yesterday		



[README](#) [Code of conduct](#) [Contributing](#) [MIT license](#) [Security](#)  


ramalama

RamaLama strives to make working with AI simple, straightforward, and familiar by using OCI containers.

Description

RamaLama is an open-source tool that simplifies the local use and serving of AI models for inference from any source through the familiar approach of containers. It allows engineers to use container-centric development patterns and benefits to extend to AI use cases.

RamaLama eliminates the need to configure the host system by instead pulling a container image specific to the GPUs discovered on the host system, and allowing you to work with various models and platforms.

SPIRES paper



OntoGPT Library



OMOP-links



Semantic SQL



Ramalama



linkML docs



Python Instructor



These slides



More detailed write-up



georgina.kennedy@unsw.edu.au

