



Engaging Content
Engaging People

A World
Leading SFI
Research
Centre



R2[RML]-ChatGPT Framework

Alex Randles and Declan O'Sullivan

ADAPT Centre for Digital Content, Trinity College Dublin, Ireland



HOST INSTITUTION



Trinity College Dublin
Coláiste na Tríonóide, Baile Átha Cliath
The University of Dublin

HOST INSTITUTION



PARTNER INSTITUTIONS



University College Dublin
An Coláiste Ollscoile, Baile Átha Cliath
Ireland's Global University



MTU
Ollscoil Teicneolaíochta na Mumhan
Munster Technological University



TUS



Maynooth University
National University of Ireland Maynooth



Ollscoil na Gaillimhe
UNIVERSITY OF GALWAY

Motivation

- Creating high-quality mappings requires a lot of background knowledge
- Mapping quality issues can exponentially multiply into resulting data

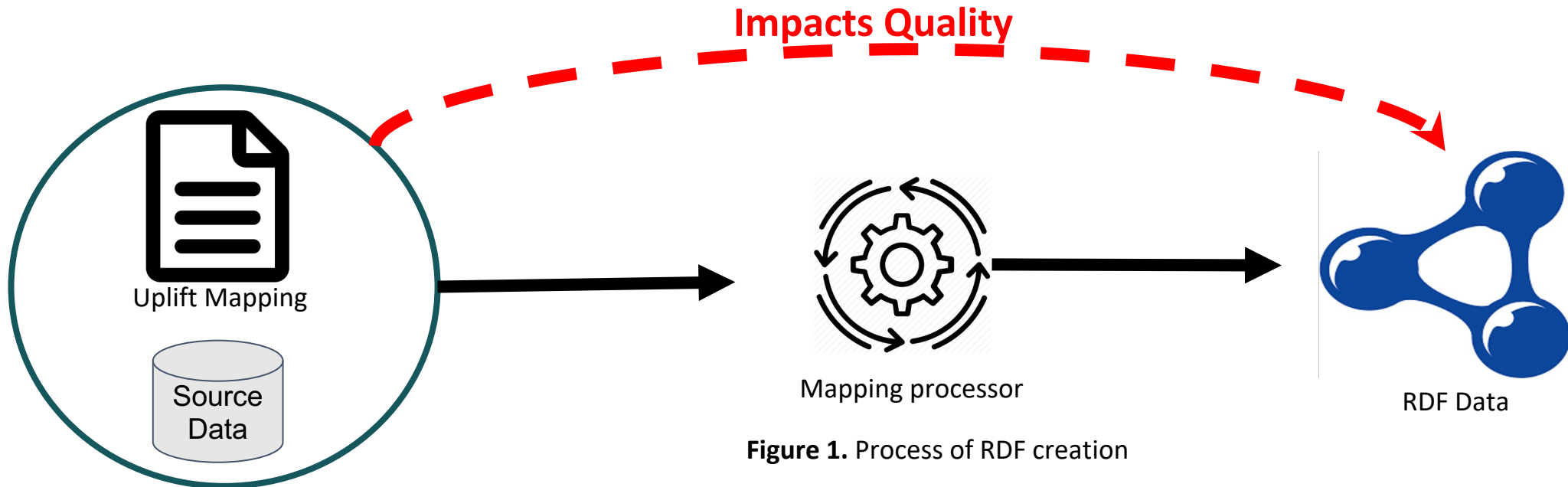


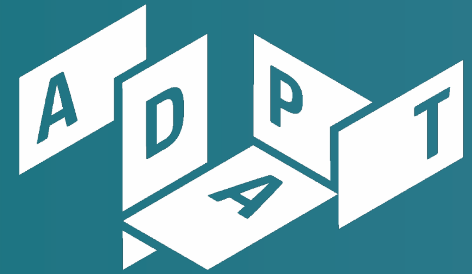
Figure 1. Process of RDF creation

- Existing quality assessment approaches often restricted to knowledge in mappings and used ontologies
- Diverse knowledge available in LLMs
- Support mapping engineers during mapping creation
 - What alternative concepts/ontologies are available?*
 - Create sample instances which use this mapping concept?*
 - Create a constraint to validate the range of this concept?*



Design and Implementation of Framework

R2[RML]-ChatGPT Framework



Engaging Content
Engaging People

- R2[RML] mapping uploaded to framework
- Pre-processing involves retrieving distinct concepts and inputting into prompt templates into ChatGPT 3.5 turbo
- Post-processing extracts, validates and improves returned code



Workflow of Framework

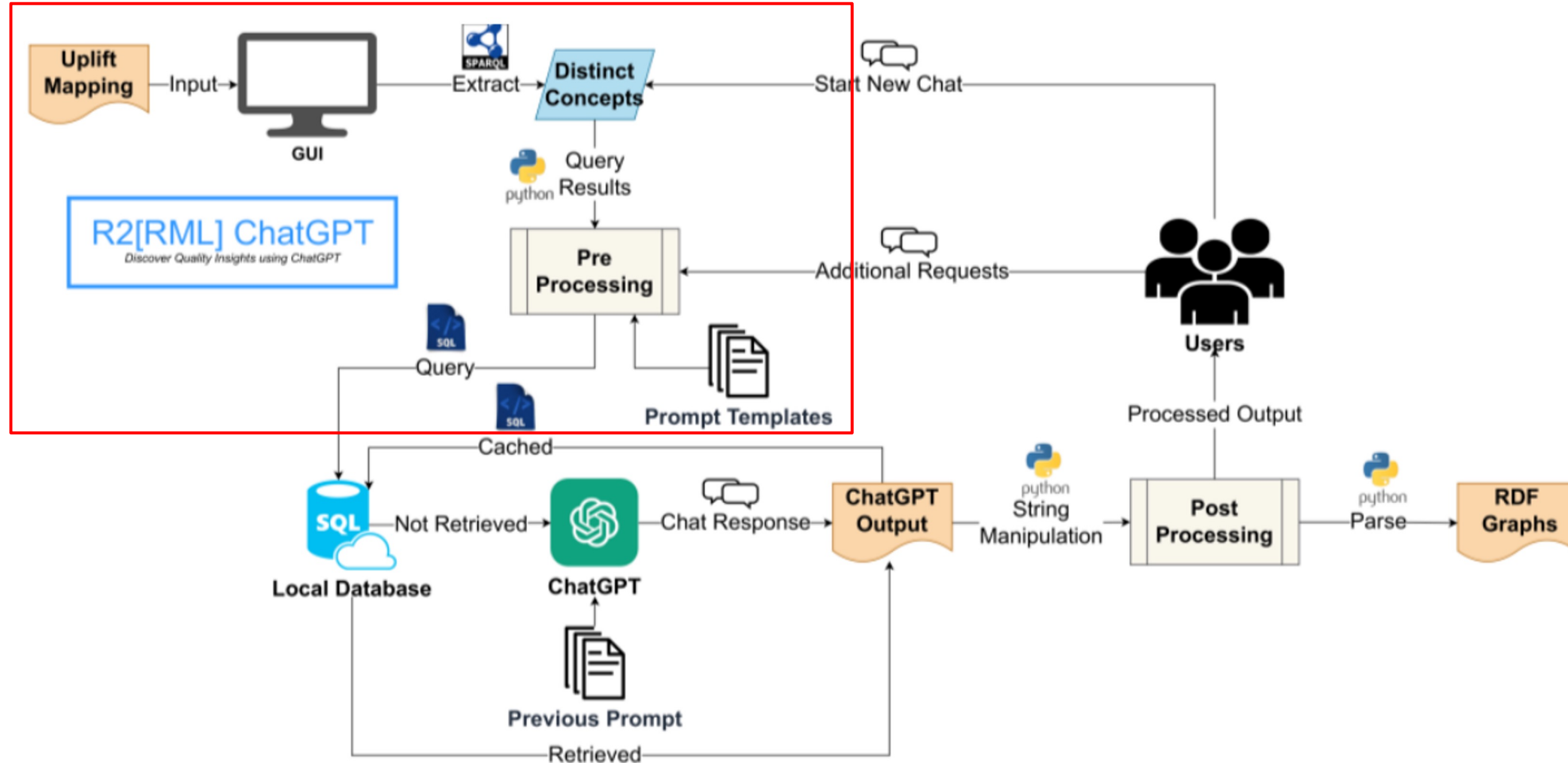


Figure 1: Workflow of the R2[RML]-ChatGPT Framework

Workflow of Framework

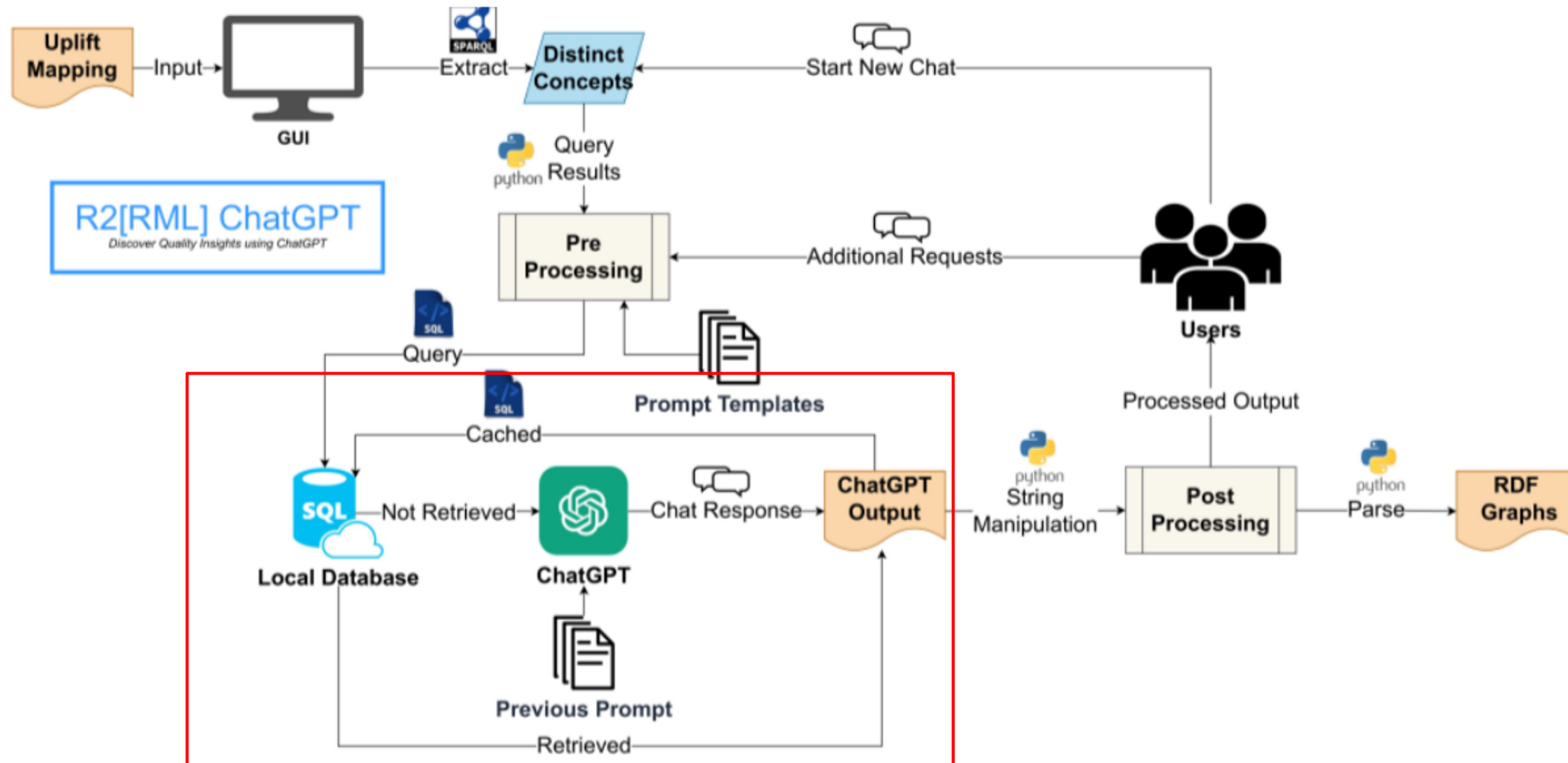


Figure 1: Workflow of the R2[RML]-ChatGPT Framework

Workflow of Framework

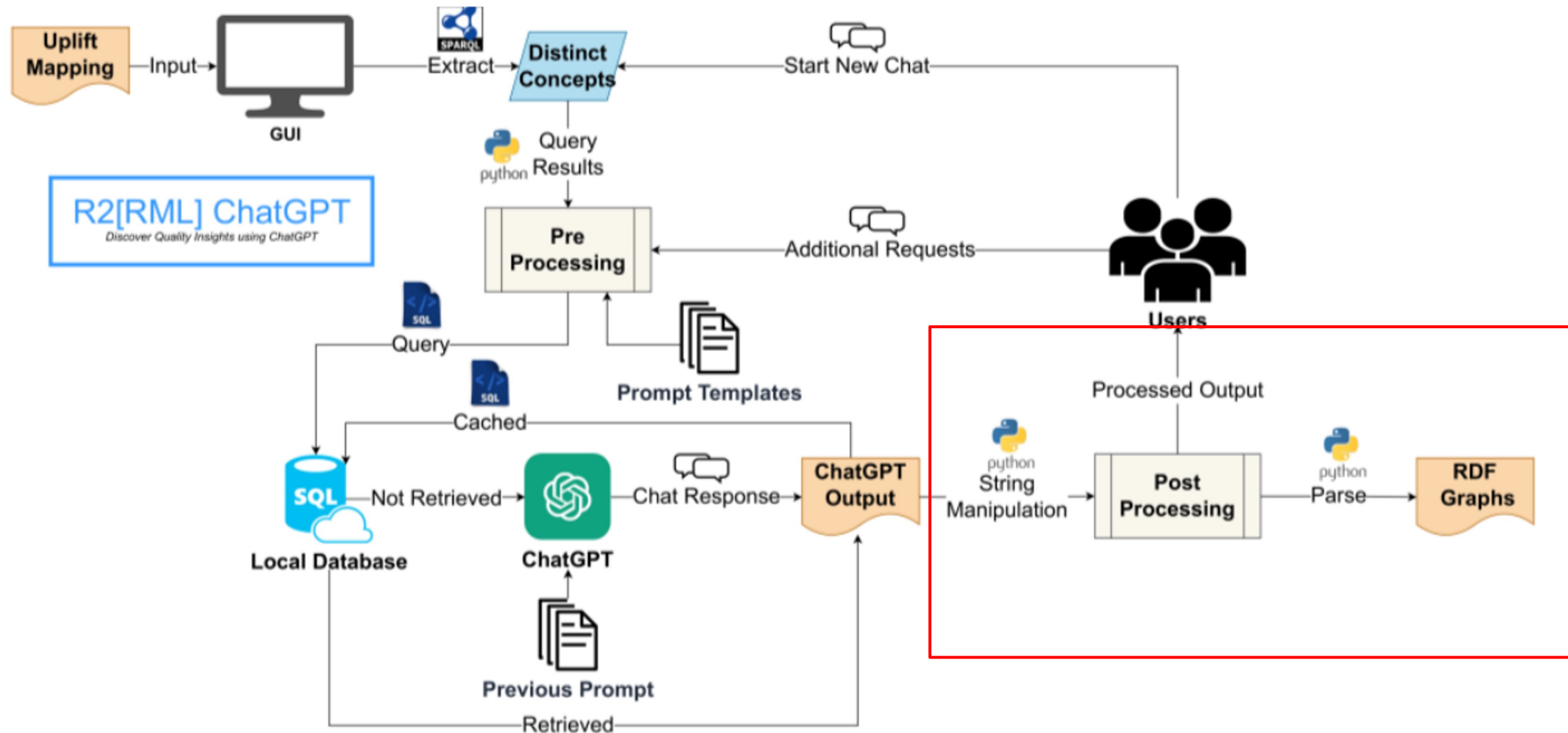


Figure 1: Workflow of the R2[RML]-ChatGPT Framework

- Several Python libraries used to implement framework

Library	Usage
Flask	Create web application for interface
SPARQLWrapper	Execute SPARQL queries
OpenAI	Communicate with ChatGPT
RDFLib	Syntax parser

Implementation of Framework

R2[RML] CHATGPT Discover Quality Insights using ChatGPT

Mapping Quality Insights

- owl:disjointWith
- geo:hasMetricSize
- prov:generatedAtTime**
- skos:Concept
- skos:example
- rdfs:Class
- owl:Class
- foaf:knows
- foaf:Person
- rdfs:comment

Press on a concept above to discover more.

Chat related to prov:generatedAtTime concept

Here is general information about the prov:generatedAtTime concept:

- The 'http://www.w3.org/ns/prov#generatedAtTime' concept is part of the PROV-O ontology, which is a standard.
- This property is critical as...
- The gene scientific...
- In RDF, this is a standard...
- Properly improve...

21:10:52

Can you provide me with key information related to the 'prov:generatedAtTime' used in RDF/OWL Technology?

Can you provide the 'rdf:type' value for the 'prov:generatedAtTime' concept defined in an ontology used in RDF/OWL Technology?

Similar Concepts Usage Alternatives **Type** Domain Range SPARQL Query #1 SPARQL Query #2

SPARQL Query #3 SHACL Shape #1 SHACL Shape #2 Sample Graph #1 Sample Graph #2

Ask a Question

Figure 4: Screenshot of Implementation displaying information related to concepts in a mapping

Implementation of Framework

R2[RML] CHATGPT Discover Quality Insights using ChatGPT

Mapping Quality Insights

- owl:disjointWith
- rdfs:label
- skos:Concept
- skos:example
- rdfs:Class
- owl:Class
- foaf:knows
- foaf:Person
- rdfs:comment

Press on a concept above to discover more.

Chat related to `rdfs:Class` concept

and how they interact with other classes in the ontology.

19:15:32

Certainly! Here's an example SPARQL query that retrieves all the classes in an ontology using the 'rdfs:Class' predicate: `''' SELECT ?class WHERE { ?class a rdfs:Class . } '''` This query selects all resources that have the RDF type 'rdfs:Class' and binds them to the variable '?class'. The 'a' predicate is shorthand for the 'rdf:type' predicate, which links a resource to its RDF type. The 'WHERE' clause specifies the pattern to match against the data, which in this case is any resource that has the type 'rdfs:Class'.

19:16:12

Response from ChatGPT

Code exportation and validation

Extract Code

Export Code (.rq, .ttl)

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?class
WHERE {
  ?class a rdfs:Class .
}
```

Similar Concepts Provenance Usage Alternatives Domain Range Disjointness SPARQL Query #1

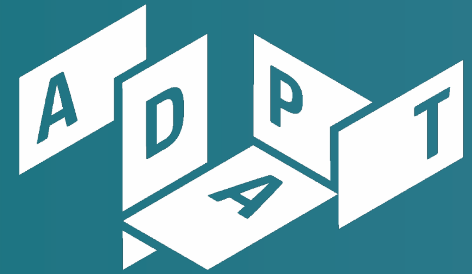
SHACL Shape #1 Sample Graph #1

Can you create a sample SPARQL query containing "rdfs:Class"?

Figure 5: Screenshot of code validation and exportation available on the framework

Experimentation

Validating Syntax and Semantics



Engaging Content
Engaging People

Experiment Overview

- **RQ1:** To what extent will ChatGPT produce semantically correct data for certain values in a declarative uplift mapping (e.g. type, domain, range and label)?
- **RQ2:** To what extent will ChatGPT produce syntactically correct RDF data and SPARQL queries?

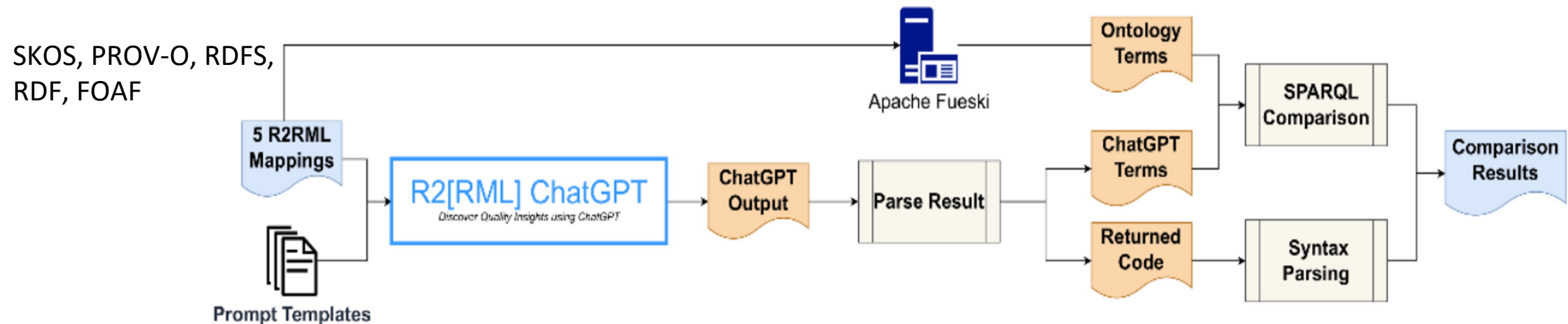


Figure 6: Overview of Activities involved in the Experimentation

- 4 ontology terms (type, domain, range and label) retrieved for concepts in mappings
- Returned terms inserted in SPARQL ASK queries
- Queries executed on namespace ontology for comparison

Semantic Correctness (RQ1) Results

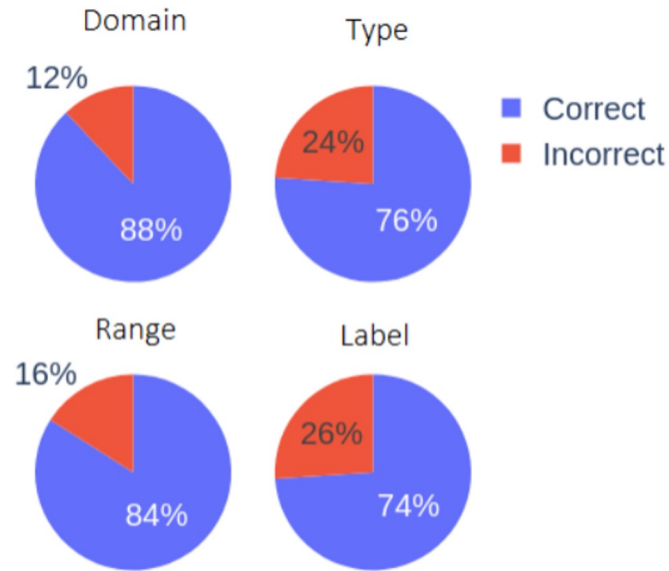
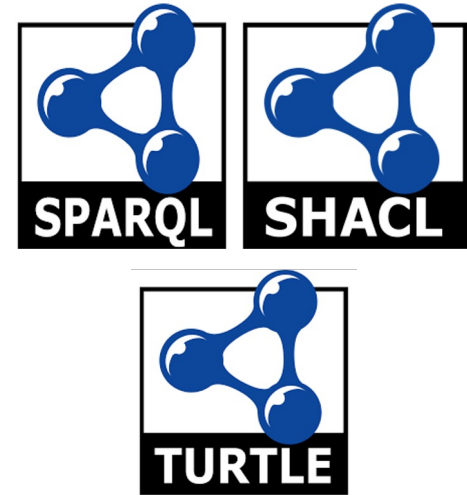


Figure 7: Results of correctness of each ontology term tested

- Domain and Range scored similar
- Type scored slightly worse with the name of tested concept returned in some cases
- Label scored worst with inferences resulting in incorrect results

Testing Syntactic Correctness (RQ2)

- 150 files containing Instances, SPARQL and SHACL generated
- Syntax validated using RDFLib parsers
- Invalid syntax improved by framework
 - Regular expressions applied on parser output
 - Added missing prefixes
 - Syntax validated again



Syntactic Correctness (RQ2) Results

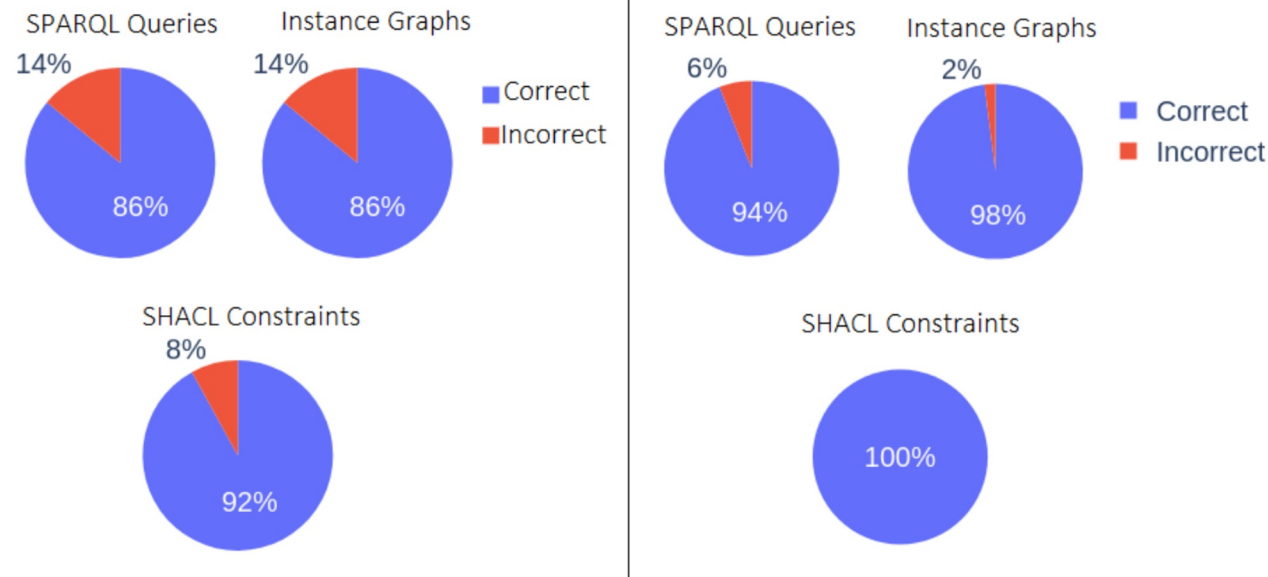
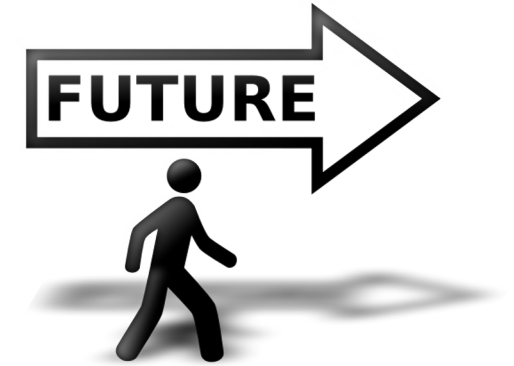


Figure 8: Results of occurrences of syntactic correctness for each category tested before (left) and after (right) post-processing by the framework

- Mean score of 42 (84%) out of 50 files correct
- SHACL constraints scored best which could be due to less prefixes
- Post-processing resolved most (14 out of 18) syntax issues

- Usability testing of framework
- Testing of other ontologies
- Extending support to interlink mappings
- Comparison of results with other LLMs and ChatGPT versions



Conclusion

- Labels scored worst with some inferences (e.g. `rdfs:rest` -> rest of list)
- Most common syntax problem was missing prefixes (14 out of 18)
- Level of ontology documentation could impact scores (rdfs better than skos)

alex.randles@adaptcentre.ie



CALL TO ACTION

**Get involved in evaluating the VRTI-KG Explorer in coming months
... no technical or historical expertise needed!**