From Unstructured Text to Knowledge Graphs: Towards More Reliable RAG Systems

Khadija ABATTANE

Introduction

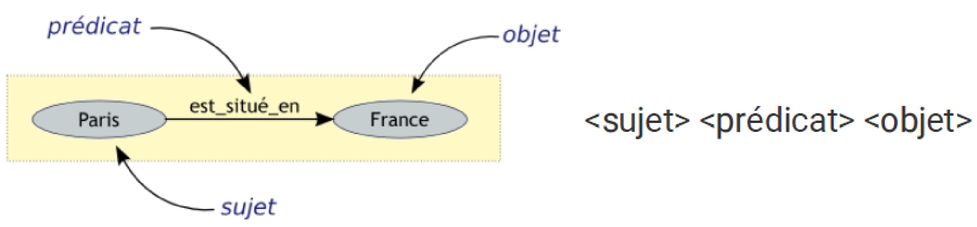
Automatic Knowledge Graph Construction

Transforming unstructured text into structured Knowledge Graphs (KGs) enables more organized and interpretable representations of information. This approach supports tasks like reasoning, decision-making, and question answering.

Our work focuses on automating the conversion of natural language into KGs, bridging the gap between raw text and structured knowledge.

While neural AI methods have recently dominated the field, KGs remain crucial due to their high reliability, fast information retrieval, and strong transparency and explainability, qualities still missing in most large language models.

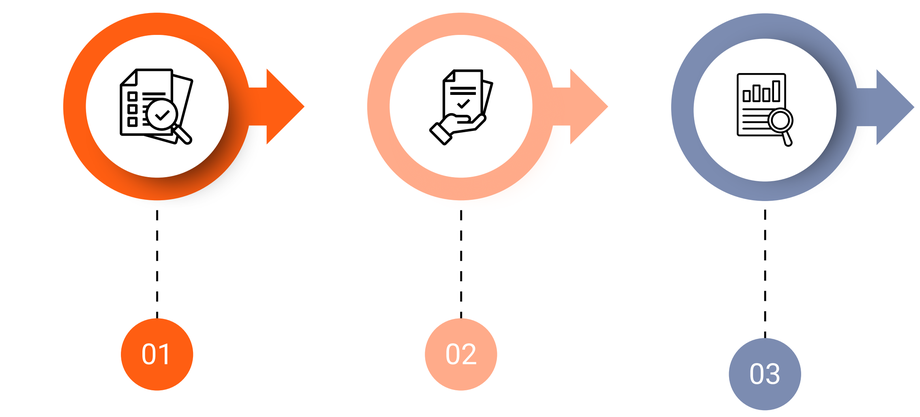
Triplets RDF (Resource Description Framework)



<sujet> <prédicat> <objet>

Keywords: Knowledge Graphs · Ontology · Neo4j · Large Language Models (LLMs)· information extraction

Method



Knowledge Extraction (KE): The corpus is segmented into coherent chunks and processed by an LLM.

Carefully designed prompts enforce:

- Predicate length: 1 to 3 words maximum (ideally 1 or 2).
- Nominal consistency: uniform use of the most complete and canonical form for each entity.
- Pronoun resolution: systematic replacement with the referenced entity.
- Atomicity of terms: identification of elementary concepts, avoiding the merging of distinct ideas.
- Typographic normalization: all components of the triplets, including proper nouns, are written in lowercase.
- Connectivity reinforcement: generation of inverse and implicit relations whenever the context allows it.
- Multiplicity of occurrences: each entity must appear in at least two triplets, with orphan entities connected through generic or inferred relations.

Knowledge Indexing (KI)

- Extracted triples are transformed from RDF into Neo4j Cypher statements.
- A knowledge graph is constructed where:
  - nodes represent entities (identified by a name property)
  - edges represent relations extracted from the text

Knowledge Querying (KQ)

- The resulting graph supports semantic exploration and relational analysis.
- Using Cypher queries, it becomes possible to:
  - detect paths between entities
  - perform aggregations over relation types

```
MATCH (s:Entity)-[r:est_associé_à]->(o:Entity {name:"Pompe à chaleur"})
RETURN s.name
```

Acknowledgements

I would like to thank the R&D department at EDF for providing a welcoming and supportive environment during my internship, which greatly contributed to my personal and professional growth.

I also warmly thank my internship supervisors for their guidance and valuable support throughout this work.

Corpus

The corpus used in this work originates from the regulatory documentation related to Energy Savings Certificates (CEE).

Decrees · **Standardized Forms** · **Technical Guidelines**

Results

We applied this knowledge extraction pipeline to 3 documents from the CEE corpus. Each document was fully processed: raw text was extracted from the PDF format, then split into 100-word segments with a 20-word overlap. For each segment, subject–predicate–object triples were automatically generated using Mixtral-8x7B-Instruct-v0.1

To assess the quality of the resulting graph, we defined several indicators aimed at measuring the **coherence** and **relevance** of the extracted entities with respect to the source text.

- **Coverage rate:** The coverage rate measures the proportion of words from the RDF triplets that also appear

in the source text.

- **Intersection rate:** The intersection rate evaluates the proportion of RDF nodes that appear in at least two distinct triplets

$$\text{Coverage Rate} = \frac{|\mathcal{W}_{\text{RDF}} \cap \mathcal{W}_{\text{TXT}}|}{|\mathcal{W}_{\text{TXT}}|}$$

$$\begin{cases} \mathcal{W}_{\text{RDF}} = \text{set of words extracted from RDF triplets} \\ \mathcal{W}_{\text{TXT}} = \text{set of distinct words from the source text} \end{cases}$$

$$\text{Intersection Rate} = \frac{|\{n \in \mathcal{N} \mid \#\text{triplets}(n) \geq 2\}|}{|\mathcal{N}|}$$

$$\mathcal{N} = \text{set of distinct RDF nodes (subjects, predicates, objects)}$$

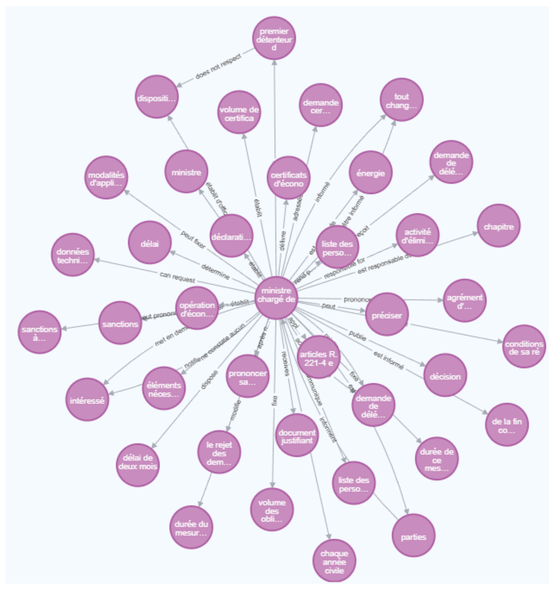
Document	Doc_1	Doc_2	Doc_3
Nombre total de triplets	1277	2805	1247
Nombre de chunks	103	240	109
Taille de chunks	100	100	100
Chevauchement (overlap)	20	20	20
Taux de couverture (%)	86.68	84.56	89.23
Taux d'intersection (%)	51.05	45.90	48.54

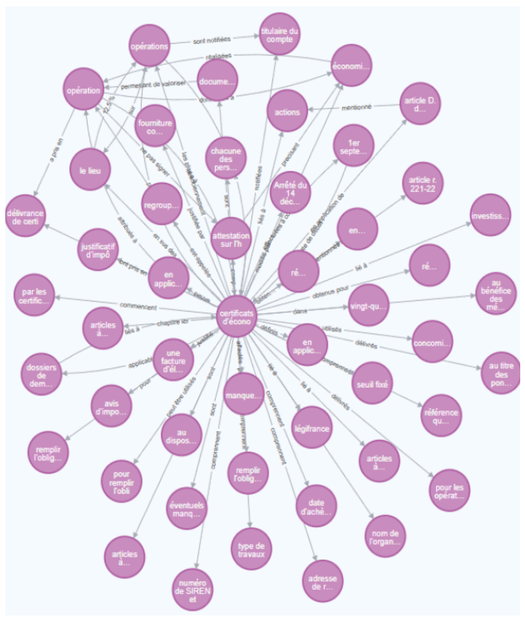
—**doc_1**: Arrêtédu 28 septembre2021 relatifaux contrôlesdans le cadre du dispositifdes certificatsd'économiesd'énergie.

—**doc_2**: Arrêtédu 4 septembre2014 fixantla listedes élémentsd'unedemandede certificatsd'économiesd'énergieet les documents à archiver par le demandeur.

—**doc_3**: Code de l'énergie: Version envigueurau 11 octobre2022.

Examples of a subgraphs centered on the entity"certificats d'économies d'énergie" and "ministre chargé de l'énergie", showing the main RDF relations extracted from the corpus





Conclusion and Perspectives

The results are promising and highlight the relevance of the proposed approach. However, several improvements remain to be explored:

Entity normalization
to merge different expressions referring to the same entity, improving graph coherence

Enhanced connectivity
by adding inverse and implicit relations where context allows

Ontology Schema
to provide a formal, reusable structure for organizing and reasoning over the extracted knowledge

References:

- McDermott, R.: From Unstructured Text to Interactive Knowledge Graphs Using LLMs (May 2025)
- Hendrik, H., Fauziati, S., Permanasari, A.E.: Enhancing Knowledge Graph Construction with Automated Source Evaluation Using Large Language Models (2024)