Application of LLM-Augmented Knowledge Graphs for Wirearchy Management

Universitat Oberta de Catalunya

Xavier Ventura de los Ojos

Data Science Master's Degree
Data Analysis and Big Data

Project supervisor
Francesc Julbe López
Coordinating professor
Albert Solé Ribalta

Date of submission
June 2024

ACKOWLEDGMENTS

It was a great joy and privilege being able to allocate the proper time to embark in this thesis. This was possible only because I received the constant support of my wife Teresa, my sons Alba and Eudald and the rest of my family. During the last months this work was our new family member who set the agenda putting so many other things on hold…
So privileged to enjoy their love and understanding.

Thanks to my work colleague and mentor Jordi Mur-Petit who guided me through the process providing unvaluable hints and recommendations that I cannot quantify.

Thanks to my work line managers José Fernando López Bello and Pablo Espasandin who expressed complete support from day one while providing all sorts of relevant information around this fascinating area.

Finally, my gratitude to my project supervisor Francesc Julbe López for his wise and clear advice and all the indications and corrections. His guidance has been a very convenient guardrail. Thanks for such an easy and comfortable journey.

With gratitude,

Xavier Ventura.
Argentona – El Maresme (Catalonia) – March-June 2024.

SUMMARY OF THE FINAL PROJECT

Title of the project:	*Application of LLM-Augmented Knowledge Graphs for Wirearchy Management*
Author name:	*Xavier Ventura de los Ojos*
Project supervisor:	*Francesc Julbe López*
Coordinating professor:	*Albert Solé Ribalta*
Date of submission (MM/YYYY):	*06/2024*
Name of the degree:	Data Science Master's Degree
Topic of the final project:	*TFM-BD Data Analysis and Big Data*
Language:	*English*
Keywords:	*knowledge graph, LLM, Wirearchy*

Abstract

Today's organization structures are porous, where formal hierarchies are intertwined with dynamic networks of interactions. This situation was coined as "Wirearchy". These structures are often volatile making very difficult to have a reliable picture of the actual organization. This unclarity can make some decision process difficult when not risky. Even worse pieces of the full picture sit with a few individuals eventually jeopardizing this corporate knowledge.
Because the Wirearchy is not available in a shared system, gathering this information is time consuming and error prone.

To alleviate this problem, we propose leveraging two main technologies: Knowledge Graphs (KG) and Large Language Models (LLM).

Graphs are ideal to model and persist these domains due to the interconnected nature of the organizational structures. But then we still need to deal with two challenges. How to keep this graph up-to-date and reliable? How can it be easily exploited by non-IT population? To solve them, recent research indicates that state of the art LLMs can be of use via LLM-augmented KG and natural language question and answer techniques.

We propose to validate these statements by performing a proof of concept using graph engines and state of the art LLMs and determining its feasibility and performance against a test dataset.

Index

1. Introduction *2*
1.1. Context and motivation 2
1.2. Goals 3
1.3. Sustainability, diversity, and ethical/social challenges 3
1.4. Approach and methodology 4
1.5. Schedule 4
1.6. Summary of the outputs of the project 7
1.7. Brief description of the remaining chapters of the report 7
2. State of the art *8*
2.1. Summary 8
2.2. Knowledge Graph. 8
2.3. Graph engine. 8
2.4. Large Language Models. 9
2.5. How to query a KG using Natural Language. 9
2.6. How to enrich the KG 10
2.7. Frameworks 11
2.8. Blueprints from Graph vendors 11
2.9. Conclusions 12
3. Methods and resources *13*
3.1 Summary 13
3.2 Building blocks 14
3.3 Datasets 16
3.4 POCs Implementation 19
4. Results *33*
4.1 POC1 Performance (Query) 33
4.2 POC2 Performance (Graph Enrichment): 40
5. Conclusion and future work *47*
5.1 Conclusion 47
5.2 Future work 49
6. References *50*

1. Introduction

1.1. Context and motivation

Today's companies recognize that organization structures are porous. On the one side there is a bureaucratic structure based on a hierarchy with clear roles and responsibilities, on the other side there are complex structures in the form of dynamic networks of interactions with different power dynamics. Jon Husband defined "Wirearchy" as "*dynamic flow of power and authority, based on information, trust, credibility, and a focus on results, enabled by interconnected technology and people"* [1][2].

These structures can be very volatile, with limited scope or visibility but still relevant for the organization. Who is working on this project? Who solved this problem? Which teams and vendors are involved in this program? Who is responsible of this service? Who is using my product? Who are the current subject matter experts of this topic? Having inaccurate or inconsistent answers to the above questions can lead to wrong decision making in terms of resource allocation, vendor management or support service. When some of this information is just in the head of a few individuals then there is a risk of corporate knowledge loss when people leave the organization or just move to a new position.

When this Wirearchy is not available in a shared system, users are forced to manually search the answer to the previous questions by pulling data from disparate systems; sometimes looking into their own personal mailbox and eventually checking with their closest peers (yet another informal network). This becomes an inefficient task with high probabilities of yielding poor or inaccurate results.

To solve the problem, we propose leveraging the following technologies:

- **Knowledge Graph** (KG): Graphs are ideal to model and persist this data domain due to the interconnected nature of the organizational structures. There are two main approaches to graphical representation of data, Resource Description Framework (RDF) and Labelled Property Graphs (LPG). Both approaches will be considered.
  Once defined the proper graph model (ontology) we need to feed it. Data source could be either applications with proper interfaces or structured or semi-structured data files. In both scenarios data can be transformed and ingested using well stablished techniques. These cases will be out of the scope of this project.
  We are interested in the underlying organizational structure that is present either explicitly or implicitly in non-structured documents. These documents could range from office documents stored in a content management system or blob storage to posts in the intranet to emails.
- **Large Language Model** (LLM): Recent research has identified two main applications of this emerging technology that are relevant to us; augmenting KGs by identifying entities and their relationships from unstructured data [3] and Knowledge Graph Question and Answer (KGQA) natural language [4].

The intended outcome is to implement proof of concept (POC) in the form of a working solution to prove the applicability of KGs and LLMs to manage a complex and dynamic organizational structure (Wirearchy).

The solution will be then validated against a dataset (documents and challenging questions).

In a personal note, along my career over the last 30 years I've been working directly or indirectly for corporations of all shapes and sizes. My experience tells me that having a clear understanding of the "extended" organization's structure is key for an optimal engagement. This information is not easily available. On the other side I see a lot of potential around the evolving technologies in scope, hence my motivation to propose this project.

1.2. Goals

- State of the art:
    - Identify candidate LLMs, Graph DBs and tools required for the task.
- Proof of Concept:
    - Architecture: Architect a solution describing its components and its integration.
    - Data: Produce a test dataset consisting of a starting graph, a set of nonstructured documents and a set of questions.
    - Implementation: Produce a working product to test the chosen technologies.
- Conclude and propose future work.

1.3. Sustainability, diversity, and ethical/social challenges

Sustainability

The proposed system does not have a direct impact on sustainability perse. Still, having better insights of the organization structure could be an enabler for more sustainable production patterns (SDG 12 - Responsible consumption).

Ethical behaviour and social responsibility

The outcome of the project is too technical to have any direct impact in ethical/social aspects. Having a better understanding of one's organization could eventually help on any corrective action aimed to improve jobs conditions. Nevertheless, whether the impact is positive or negative will depend on the decisions taken based on the acquired knowledge.

Diversity, gender and human rights

In terms of gender, the target 5.5 of the SDG 5 – *Gender equality* aims to "women's full and effective participation and equal opportunities for leadership at all levels of decision-making in political, economic and public life". Indicator 5.5.2 is "Proportion of women in managerial positions".

If we incorporate both the gender and a flag indicating if a person is in a managerial position, then we could ask the system about how the organization is performing regarding 5.5.2.

When it comes to accessibility, the fact that we can query on the organization structure using natural language instead of technical languages such as GraphQL or Cypher will make it accessible to a wider range of users not limited to IT skilled employees.

1.4. Approach and methodology.

There are two possible approaches to develop this project:

- Perform a market analysis to identify any existing product such as HR or Content Management Solutions (CMS) that already fulfils the proposed requirements.
- Design / build from scratch: Propose an implementation strategy and identify existing building blocks (mainly open source). For the features that are missing or not fully working, identify the lines of work or investigation that would fill the gap in the future once a proper implementation is available.

Here the reasons why I consider that designing and building from scratch is the most convenient option:

- Rapidly evolving technologies: New LLMs are constantly emerging with new options being available. A commercial of the shelf solution would need to be updated to leverage any recent development.
- Flexibility: Integrate components from different open-source initiatives or vendors if required give more flexibility and room for innovation.
- Academic approach: I see more value in focusing on the state-of-the-art analysis of the core components rather than identify a closed solution.

1.5. Schedule

The project will require the following resources:

- Graph Engine: This could be either open-source systems or free versions of commercial offerings.
- LLMs: One or two of the many available options today. We'll look for the most cost-effective option for the task at hand.

- POC: Open-source frameworks required to build a basic UI and to integrate the KG and the LLMs.

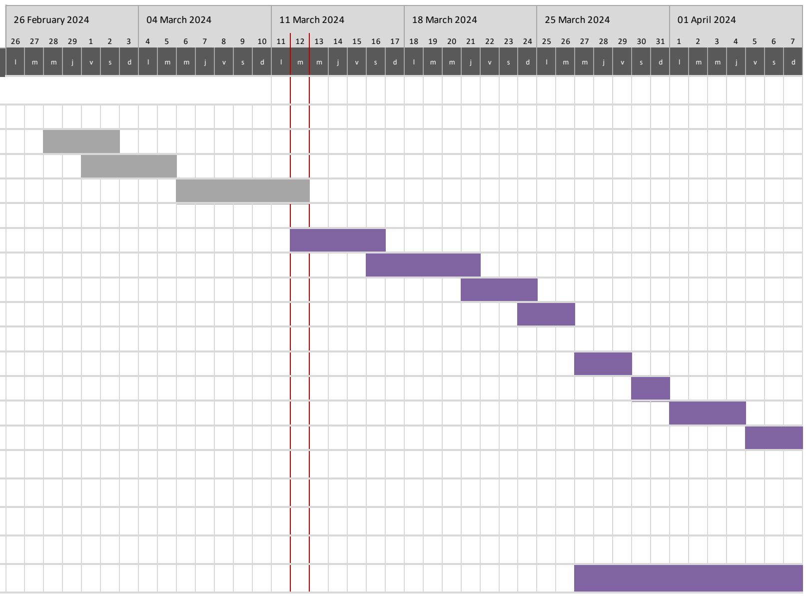
The project plan is as follows:

- M2 State of the Art
    - Identify core components (Graph): Determine the suitable Graph engine either RDF or LPG.
    - Identify core components (LLMs): Locate the appropriate LLMs for Graph augmentation and for Natural language query.
    - Identify core components (Frameworks): Find the needed libraries and to implement the POC.
    - Document the above findings with supporting papers. (Milestone)
- M3 Development
    - Integrate the building blocks: Draw the POC's sequence diagram and check for any missing component from the previous task.
    - Deploy Graph Engine: Setup the chosen Graph Engine.
    - Produce starting KG: Create the initial organizational structure in the form of a Knowledge Graph including both the ontology and the data.
    - Create Data set (I): Build the list of test questions using natural language.
    - Implement POC Phase 1 (query): Implement basic UI by integrating the chosen LLMs for natural language Q&A
    - Validate Phase 1: Validate LLM performance on the test questions.
        - Milestone: **POC Phase 1 completion**
    - Create Data set (II): Build a list of documents from where to infer the Wirearchy with the expected outcome from each document.
    - Implement POC Phase 2 (enrichment): Implement the artifacts that will enrich the KG from the text in the dataset leveraging the LLMs.
    - Validate Phase 2: Evaluate the performance by comparing the proposed changes from each document versus the expected results.
        - Milestone: **POC Phase 2 completion**
    - Project documentation (draft): Write the initial draft of the project documentation.
- M4 Project documentation
    - Project memory final draft.
    - Project memory final version.
    - Video recording of project presentation.

Project GANTT:

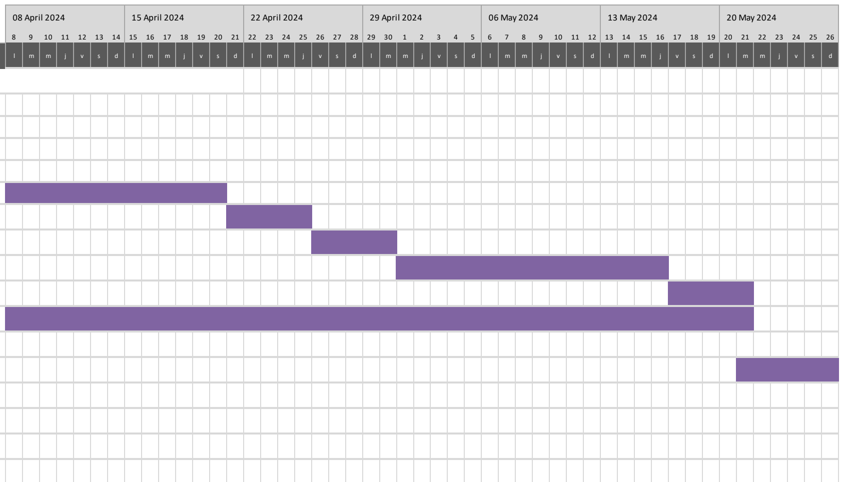
Week 1

TASK	START	END
M1 Thesis Definition		
Choose subject & initial research	28-2-24	2-3-24
First draft	1-3-24	5-3-24
Initial proposal	6-3-24	12-3-24
M2 State of the Art		
Identify core components (Graph)	12-3-24	16-3-24
Identifiy core compoments (LLMs)	16-3-24	21-3-24
Identify core components (Frameworks)	21-3-24	24-3-24
Dcoument findings with supporing papers	24-3-24	26-3-24
M3 Development		
Define architecture	27-3-24	29-3-24
Deploy Graph Engine	30-3-24	31-3-24
Create Data set (I)	1-4-24	4-4-24
Implement POC (query)	5-4-24	20-4-24
Validate Phase 1	21-4-24	25-4-24
Create Data set (II)	26-4-24	30-4-24
Implement POC (enrichment)	1-5-24	16-5-24
Validate Phase 2	17-5-24	21-5-24
Project documentation (draft)	27-3-24	21-5-24



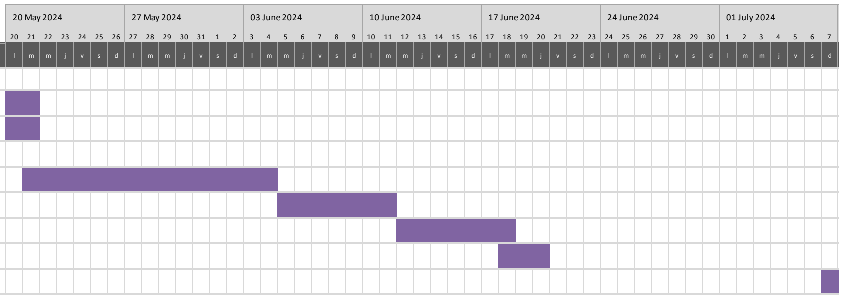
Week 7

TASK	START	END
M3 Development		
Define architecture	27-3-24	29-3-24
Deploy Graph Engine	30-3-24	31-3-24
Create Data set (I)	1-4-24	4-4-24
Implement POC (query)	5-4-24	20-4-24
Validate Phase 1	21-4-24	25-4-24
Create Data set (II)	26-4-24	30-4-24
Implement POC (enrichment)	1-5-24	16-5-24
Validate Phase 2	17-5-24	21-5-24
Project documentation (draft)	27-3-24	21-5-24
M4 - M5 Final delivarables & defense		
Project documentation (draft)	21-5-24	4-6-24
Project documentation (final)	5-6-24	11-6-24
Project presentation (video)	12-6-24	18-6-24
Final delivery	18-6-24	20-6-24
Project Defense	7-7-24	7-7-24



Week 13

TASK	START	END
M3 Development		
Validate Phase 2	17-5-24	21-5-24
Project documentation (draft)	27-3-24	21-5-24
M4 - M5 Final delivarables & defense		
Project documentation (draft)	21-5-24	4-6-24
Project documentation (final)	5-6-24	11-6-24
Project presentation (video)	12-6-24	18-6-24
Final delivery	18-6-24	20-6-24
Project Defense	7-7-24	7-7-24



1.6. Summary of the outputs of the project

To achieve the project goals, we'll produce the following deliverables:

- Project documentation including video presentation.
- Test datasets.
- POC implementation including source code and datasets.

1.7. Brief description of the remaining chapters of the report

The report will contain the following sections and subsections:

- Methods and resources
    - State of the art: Description of the process for choosing the building blocks for the POC: KG, LLMs and Framework.
    - Building Blocks: Identify the required building blocks used for the POC implementation.
    - Datasets: Explain the criteria used to define the datasets, list its content and describe in detail one or two samples.
    - POC implementation: Describe the implementation process with references to the selected components and the code base.
- Results
    - Performance (Query): Describe the testing and performance evaluation process with the detailed results for the natural language query.
    - Performance (Enrichment): Describe the testing and performance evaluation process with the detailed results for the KG enrichment.
- Conclusion and future work
    - Conclusion: Establish if the POC was successful in terms of the performance of the two requirements.
    - Future work: Identify areas for improvement and related lines of research to follow up.
- References.

2. State of the art

2.1. Summary

In this deliverable we perform an initial literature review that will be refined during the building phase. This review sets the foundation and helps fine tune the expectations around the two main project objectives: Enrich a domain specific Knowledge Graph (KG) using unstructured documents and enable a natural language Q&A engine on top of the previous KG.

In both use cases we want to assess the feasibility of using general use LLMs as the core building block.

2.2. Knowledge Graph.

The Domain-specific Knowledge graphs survey [5] confirms that a KG is the perfect model to capture a Wirearchy. More specifically, the Wirearchy fits within the *Societies and politics* domain proposed be the author (Abu-Salih, 2021, p. 9):
Men who have established systems by making formal and informal decisions concerning the production, distribution, and the use of various resources. The formal representation of KGs provides an excellent mean to conceptualize relationships in social sciences and politics.

In the same paper, Abu-Salih proposes a taxonomy with the key aspects for KG construction (entities and relations).
Entity extraction comprises: i) Named Entity Recognition (NER) finding individuals, organizations, events and other relevant entities from unstructured data sources; ii) Named Entity Disambiguation (NED) by mapping the inferred entity to the factual real-world entity; and iii) Named Entity Linking (NEL) which assign a unique IRI or identifier to the entity.
Relation extraction aims to discover the semantic relationships between the identified entities.

2.3. Graph engine.

Once stablished the convenience of using a KG, we need to identify the proper architecture for its implementation, considering both the persistence and query capabilities.
There are two main architectures to consider: Resource Description Framework (RDF) designed in the early-2000 for standardization of data exchange on the web and Label Property Graph (LPG) as a model for storing Enterprise Content Management data. These have been seen as opposed models.
The recent publication of the RDF-star extension though aligns RDF with the LPG approach while keeping RDF’s benefits [6].
When it comes to the query capabilities, RDF graph is mostly queried using SPARQL where in the case of LPG, Cypher has been the de-facto standard, and it has been recently published as ISO/IEC 39074:2024 standard under the Graph Query Language (GPL) name [7].
Cypher or GPL has a richer expressivity which makes it a better candidate to produce and understand the queries for the users with a SQL background.

For the above reasons we'll consider LPG as the preferred technology for the implementation phase:

- Commercial: Neo4j Desktop (free tier) [8].

To build the initial KG ontology and dataset we will leverage the open data repository of the "Generalitat de Catalunya".

2.4. Large Language Models.

The aim of the study is to leverage LLMs for optimal KG enrichment and to provide natural language query capabilities. In this section we review the state-of-the art of LLMs in relation to KGs.

After the invention of Transformers by the Google Brain and Google Research teams in 2017 [9] there have been an explosion[1] on the usage of LLMs for a wide range of applications. With thousands of NLP models available today [10], we focus on our area of interest by following the work of Shirui Pan et al (2023) [3].

Their article is a forward-looking roadmap for unifying LLMs and KGs while leveraging their strengths and overcome their limitations. The roadmap is organized in 3 frameworks:

1) KG-enhanced LLMs; 2) LLM-augmented KGs; 3) Synergized LLMs + KGs

We are interested in the following methods:

KG-enhanced LLM inference → Retrieval-augmented knowledge fusion.
KG-enhanced LLM inference → KGs Prompting.
LLM-augmented KG construction → End-to-End KG construction.

2.5. How to query a KG using Natural Language.

Retrieval Augmented Knowledge Fusion: RAG is a popular method to inject knowledge into LLMs during inference. The key idea is to retrieve relevant knowledge from a large corpus, the KG, and then fuse the retrieved knowledge into LLMs.

KGs Prompting: By using the prompt, we can easily harness the power of LLMs to perform reasoning based on KGs without retraining the models. However, the prompt is usually designed manually, which requires lots of human effort.

(Li et al., 2023) uses a predefined template that is fed via the retrieval and re-rank of the most relevant triplets from KGs, which are then concatenated with questions to be fed into language models [11]. While this method can provide better accuracy outperforming complicated question and answer systems, it is constrained into multi-choice Q&A.

[1] Over 113000 citations in Google Scholar of the seminal paper "Attention is all you need."

2.6. How to enrich the KG.

LLM-augmented KG Construction: Knowledge graph construction involves creating a structured representation of knowledge within a specific domain. This includes identifying entities and their relationships with each other. The process of knowledge graph construction typically involves multiple stages, including 1) entity discovery, 2) coreference resolution, and 3) relation extraction. Fig 19 presents the general framework of applying LLMs for each stage in KG construction. More recent approaches have explored 4) end-to-end knowledge graph construction, which involves constructing a complete knowledge graph in one step or directly 5) distilling knowledge graphs from LLMs.

(Han et al., 2024) propose a framework for *End-to-End KG Construction*, Prompting with Iterative Verification (PiVe) [12], to improve graph-based generative capability of LLMs. They show how a small language model could be trained to act as a verifier module for the output of an LLM. But this method requires the training of a verifier module (LM) that is not applicable in our use case due to the data sources variability.

This work is inspired by Chain-of-thought.

(Wei et al., 2022) explore how generating a chain of thought—a series of intermediate reasoning steps—significantly improves the ability of large language models to perform complex reasoning. [13]
This paper combines the strengths of two ideas. First, arithmetic reasoning can be solved via natural language rationales. Second, LLMs offer in-context few-shot learning via prompting. Specifically, they explore the ability of language models to perform few-shot prompting for reasoning tasks, given a prompt that consists of triples: ⟨input, chain of thought, output⟩. A chain of thought is a series of intermediate natural language reasoning steps that lead to the final output.

The LLMs in scope of the POCs are:

- OpenAI GPT-3.5 Turbo.
- OpenAI GPT 4 Onmi.
- Anthropic Claude 3 Haiku.
- Anthropic Claude 3 Opus.

2.7. Frameworks

There are a few available frameworks or orchestration libraries to implement LLM based solutions referred by several projects and papers.

Here an extract from the Awesome-LLM[2] curated list:

- LangChain[3]: LangChain is a framework for developing applications powered by language models.
- LlamaIndex[4]: Production Ready Data Framework for LLM-applications.
- Haystack[5]: The Production-Ready Open-Source AI Framework.
- Embedchain[6]: Embedchain is an Open-Source Framework that makes it easy to create and deploy personalized AI apps.
- Chainlit[7]: Chainlit is an open-source Python package to build production ready Conversational AI.

2.8. Blueprints from Graph vendors

The capabilities that LLMs can provide to provide additional capabilities did not go unnoticed to graph vendors who are incorporating them either a part of their products, in their roadmaps or in the form of blueprints:

- Stardog Voicebox[8]: Have a conversation with your data.
- GraphDB (ontotext)[9]: Talk to Your Graph (experimental).
- Neo4j[10]: GenAI Stack for Developers.
- NebulaGraph[11]: Database + Large Language Model

[2] https://github.com/Hannibal046/Awesome-LLM?tab=readme-ov-file#deploying-tools
[3] https://python.langchain.com/docs/get_started/introduction
[4] https://www.llamaindex.ai/open-source
[5] https://haystack.deepset.ai
[6] https://docs.embedchain.ai/get-started/introduction
[7] https://docs.chainlit.io
[8] https://stardog.ai
[9] https://graphdb.ontotext.com/documentation/10.6/talk-to-graph.html
[10] https://neo4j.com/blog/introducing-genai-stack-developers/
[11] https://www.nebula-graph.io/ai_llm

2.9. Conclusions

After the state-of-the-art analysis we observe that:

There is evidence from the scientific community that there are available LLMs capable of performing the required tasks: KG construction and Natural Language query. The models' performance is though tested under very specific conditions using reference datasets for comparison within several models and against previous architectures.

On the other hand, graph vendors claim that their databases can be easily queried using natural language by leveraging LLMs. In some cases, implementation details are provided for demonstration purposes.

Finally, there is a plethora of frameworks and libraries to build LLMs based applications.

In any of the above we have observed that the proposed "Wirearchy" is in fact properly solved or demonstrated.

We can conclude that we have the necessary building blocks but not an end-to-end solution that solves the problem statement.

The rest of the thesis will then consist of the actual implementation of a functional POC that proves our initial hypothesis.

3. Methods and resources

3.1 Summary

In this section we describe the method and scope of the thesis implementation. The core technologies Knowledge Graph and LLMs are proposed, and decision is made to build a POC from scratch to test them in the context of a Wirearchy.
Building a complete working solution in the sense of a Minimum Viable Product (MVP) is beyond the scope of the thesis due to the required effort. The proposal is to put in motion the referred technology and test it against a realistic dataset to determine the feasibility of this approach.

The implementation process is then:

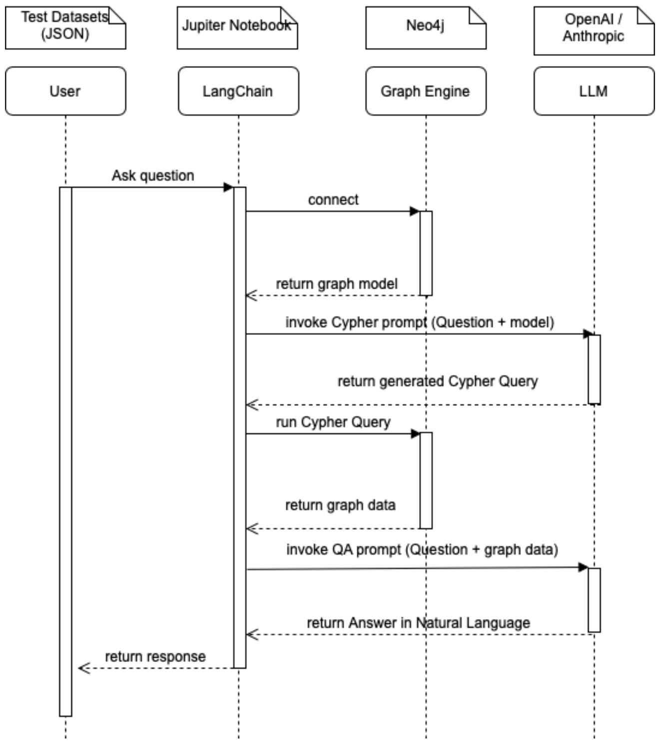
- Identify the required components: Preferred option open source or free products with exception made with the LLMs for which we require a subscription with the corresponding vendors.
- Produce a test dataset: We identify publicly available datasets that would reflect the concept of the Wierarchy.
- Deploy a graph engine and create a graph from the dataset.
- Implement two POCs in the form of two Jupyter notebooks that will serve as labs to validate the two hypotheses:
    - POC1: How to query a KG using Natural Language.
    - POC2: How to enrich the KG from unstructured documents.
- Perform the actual tests:
    - POC1: Run a set of question in batch using different configurations / LLMs.
    - POC2: Extract structured data from a set of public documents relevant to the graph and enrich it. The extraction is done using different models so that we can compare their performance.

In the section 4 Results, the results of the POCs are presented with the final conclusions and proposal for future work in section 5.

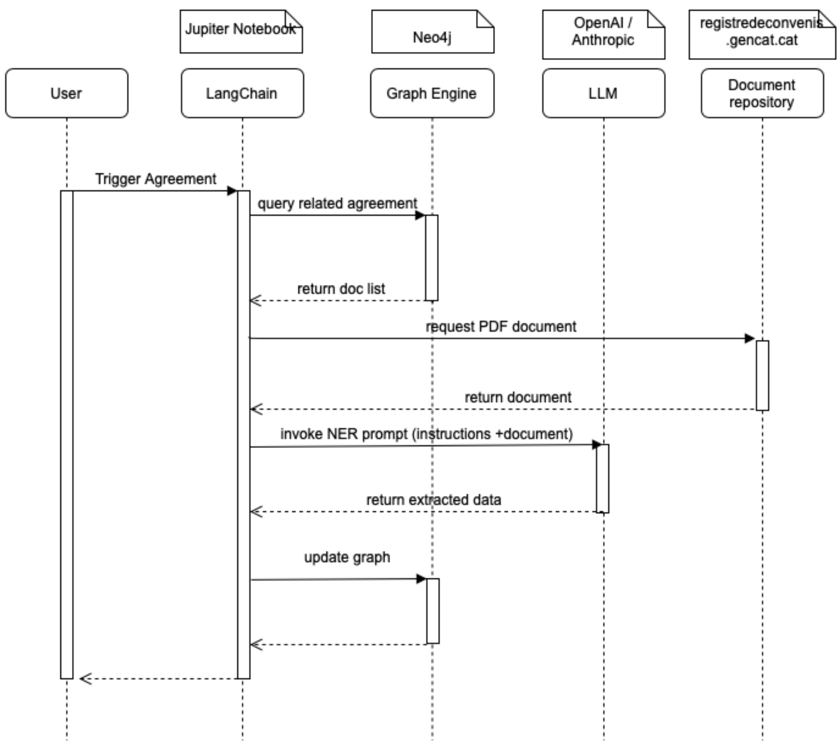
3.2 Building blocks

The solution consists of the implementation of these two sequence diagrams:

Query a Knowledge Graph using Natural Language



Enrich a Knowledge Graph extracting content from an unstructured document



The required components are:

- Graph Engine (Neo4j Desktop): Repository of the Knowledge Graph with Data Manipulation Language (DML), Data Definition Language (DDL), ETL and query capabilities.

- Graph Browsers: UI to visualize / query on top of the graph engine.
    - Neo4j Explorer (to execute CYPHER statements).
    - Neo4j Bloom (for interactive graph query / navigation).

- LLMs for both Natural Language query and Named Entity Recognition.
  We use 4 of the most popular multitask / multipurpose models with:

Vendor	**Model Name**	**Price (*) (Input)**	**Price (*) (Output)**
Open AI	GPT 3.5 Turbo	$0,50	$1,50
Open AI	GPT 4 Omni (**)	$5	$15
Anthropic	Claude3 Haiku	$0,25	$1,25
Anthropic	Claude3 Opus	$15	$75

  (*) Prices as per 1 million tokens. Price list checked on 9 June 2024.

  The list of models can be easily extended as per the POC design.

- LangChain: Framework for developing applications powered by LLMs.
- GraphCypherQAChain: Chain to interact with Neo4j graph.
- Jupiter Notebooks and other python modules: Execution environment to demonstrate the capabilities of the above components by implementing the corresponding sequence diagrams.
- Datasets:
    - Opendata csv and public PDF documents from Gencat.
    - Sample questions.
- Templates: Prompt templates for Cypher Generation.

(**) GPT 4. Omni was released on May 13, 2024, as an improved version of GPT-4 Turbo released Nov 6, 2023, and at half the cost. This shows how dynamic is this sector.

3.3 Datasets

After initial fruitless research on public graph databases looking for existing graphs that reflected an Wierarchy structure we moved to the standard approach of extracting structured files. Still, it is not easy to find public datasets reflecting internal organizational structures with people names, roles, departments and so on.

Turns out that because of legal obligations[12], the Catalan government makes available certain information on its organization and activities.

This dataset was chosen because:

License: *Open data licenses and terms of use are subject to public sector information reuse laws. In some cases, they may have intellectual property licenses, although they tend to be opened without conditions, as long as they remain unaltered and with mandatory citation of the source and its last update. For more information, you can consult the "Terms of use and licenses" section.*

Detailed information is available in the gencat portal here:

https://administraciodigital.gencat.cat/ca/dades/dades-obertes/informacio-practica/llicencies/

Fit for purpose: The files describe a large structure with over 70000 nodes:

- 8000 organizations
- 12000 groups of interests
- 4500 people
- 44000 events
- 100 agreements (only a small subset is loaded from the file with 36000 entries) with links to PDF documents.

The specific files used for the implementation, also available in the repo, were extracted from:

https://administraciodigital.gencat.cat/ca/dades/dades-obertes/les-dades-obertes/que-son-les-dades-obertes/

https://analisi.transparenciacatalunya.cat

The files in scope of the POC are:

- Tabular organization chart of the Generalitat de Catalunya[13] (org.csv)

  Organigrama dels organismes que componen la Generalitat de Catalunya i dels seus responsables, així com els organismes autònoms, empreses públiques i consorcis en format tabular.

[12] Llei 19/2014,del 29 de desembre,de transparència, accés a la informació pública i bon govern.
[13]https://analisi.transparenciacatalunya.cat/en/Sector-P-blic/Organigrama-tabular-de-la-Generalitat-de-Catalunya/czns-gsc6/about_data

- Register of interest groups in Catalonia[14] (grups.csv)

  Dades identificatives dels grups inscrits al Registre de grups d'interès de Catalunya que poden influir directament o indirectament en l'elaboració i l'aplicació de les polítiques públiques. Recull la informació sobre la finalitat i objectius del grup, la categoria i els àmbits d'interès des de la seva creació el 16 de febrer de 2017.

- Public agenda with interest groups of senior officials and management personnel[15] (agenda.csv)

  Publicitat activa de l'agenda pública amb grups d'interès dels alts càrrecs i personal directiu; subdirectors/es i llocs assimilats de la Generalitat de Catalunya.

- Registration of Collaboration and Cooperation Agreements[16] (convenis.csv)

  Dades sobre els convenis i encàrrecs de gestió de les Administracions públiques catalanes inscrits en el Registre de Convenis de Col·laboració i Cooperació. Dades a partir de l'1 de juliol de 2015.

Sample questions

To test the Natural Language query capabilities, we use the following list of questions which are stored in the `poc1_config/questions.json` file:

Id	**Question**
Q001	Qui és el president de la Generalitat de Catalunya?
Q002	Qui té actualment el càrrec de 'President de la Generalitat de Catalunya'?
Q003	Qui és el responsable de Direcció General de Turisme?
Q004	Quina és la estructura del Parlament de Catalunya?
Q005	Quines reunions s'han celebrat amb el Grup Universitat Oberta de Catalunya?
Q006	Qui es va reunir amb el grup Universitat Oberta de Catalunya?
Q007	Qui és Jaume Giró Ribas?
Q008	Quan es va reunir en Tomàs Roy Català amb el grup Universitat Oberta de Catalunya? I quin tema es va tractar?
Q009	Amb qui s'ha reunit en Tomàs Roy Català?
Q010	Llista les 10 persones amb més carrecs
Q011	Parlam sobre Elisenda Guillaumes Cullell?
Q012	Quina és la relació entre 'Miquel Salazar Canalda' i 'Joan Vintró Castells'? Descriu la relació pas a pas.
Q013	Quins grups s'han reunit per tractar sobre la sequera? Incloure el tema de la reunió i la data.

[14] https://analisi.transparenciacatalunya.cat/en/Legislaci-just-cia/Registre-de-grups-d-inter-s-de-Catalunya/gwpn-de62/about_data
[15] https://analisi.transparenciacatalunya.cat/Sector-P-blic/Agenda-p-blica-amb-grups-d-inter-s-dels-alts-c-rre/hd8k-y28e/about_data
[16] https://analisi.transparenciacatalunya.cat/Sector-P-blic/Registre-de-Convenis-de-Col-laboraci-i-Cooperaci-/exh2-diuf/about_data

Id	**Question**
Q014	Amb quins grups s'ha parlat més sobre la sequera?
Q015	Quines persones han estan més actives en relació a la sequera? Mostra la llista de les 5 persones per any.
Q016	Quins organismes han mantingut més reunions per tractar sobre la sequera?
Q017	Quina va ser la reunió amb més participants l'any 2023?
Q018	Què saps del conveni 2022/9/0304?
Q019	Qui ha signat el conveni 2022/9/0304? Inclou tots els convenis relacionats. Llista les entitats i el nom de la persona i el seu càrrec.
Q020	Llista totes les persones, organitzacions i grups que han signat el conveni 2022/9/0304. Inclou tots els convenis relacionats amb el 2022/9/0304.

One very interesting feature of the LLMs is scope is the capacity of manage content in different languages. In our case the questions are in Catalan.
What is more interesting is that the prompts templates are written in English which means that the models can deal with prompts with mixed content. This is particularly impressive in the case of the POC2 Named Entity Resolution.

3.4 POCs Implementation

The POC implementation consist of the following components:

- Datasets for graph.
- Sample questions, prompt templates and config files.
- Graph Database.
- Jupyter Notebooks:
    - POC 1: How to query a KG using Natural Language.
    - POC 2: How to enrich a KG extracting content from unstructured documents.

The POC codebase, datafiles and test results are available in the tfm-uoc-wirearchy repository in GitHub.[15]

Folders	
images	README.md (screenshots)
neo4j	Datafiles and cypher scripts for graph creation: ***<datasource_file>.csv***: dataset in csv format ***load_<step>.cypher***: Cypher code for the graph creation via the CSV files ingestion.
poc1_answers	JSON files with the results / answers to the test questions (POC1)
poc1_config	Config files for the POC1: ***configurations.json***: list of test scenarios (LLM + prompt) ***prompt_template_%.txt***: prompt templates for the CYHPER generation chain. ***questions.json***: list of test questions.
poc2_extracts	***<agreement_code>.json*** files with the results of the NER process (POC2). the files are stored in subfolders named after the LLM used for each run of the data extraction test: **run01_gpt35**: extracts using gpt-3.5-turbo **run02_gpt4o**: extracts using gpt-4o **run03_haiku**: extracts using claude-3-haiku **run04_opus**: extracts using claude-3-opus
Files	
poc1 query graph.ipynb	Jupyter Notebook with POC 1: How to query a KG using Natural Language.
poc2 enrich graph (agreements).ipynb	Jupyter Notebook with POC 2: How to enrich a KG extracting content from unstructured documents
README.md	Readme document.
LICENSE.txt	License

The first step is to create a Graph from the datafiles in scope.

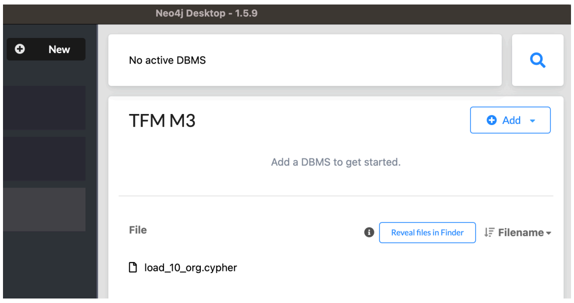
The project requires the Neo4j Graph Database. Neo4j Desktop is a very convenient (free) option to work with Neo4j: https://neo4j.com/docs/desktop-manual/current/

Graph Creation

The step-by-step instructions of the graph creation and the initial data load are available in the Graph Database section of the repository README file [15]. The same instructions are reproduced here for convenience:

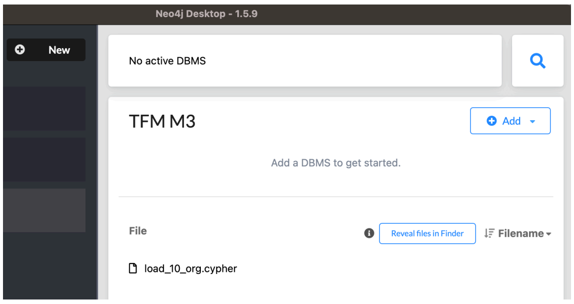
Follow these steps to create the graph and load the CSV files downloaded from the Open Data portal.

- Open the Neo4j Desktop application.
- Create a New project from directory. Choose the "neo4j" folder that contains the csv files and the cypher scripts.
- Select the newly created project. You can rename it by clicking on the project's name.

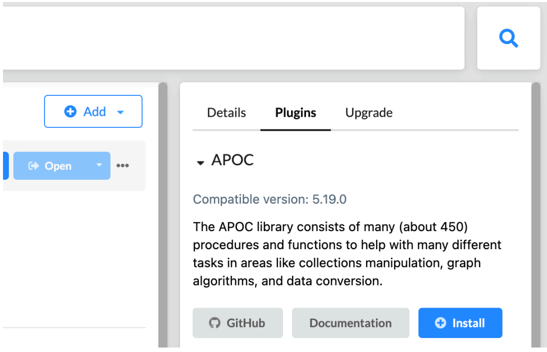


- Add a new Local DBMS using version 5.18.1 or above.

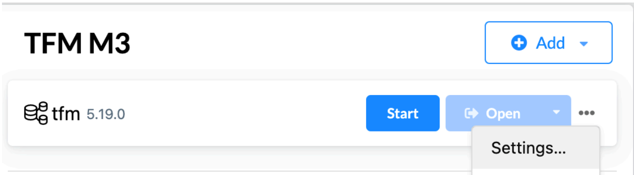
Note: Carefully write down the provided database name and password.



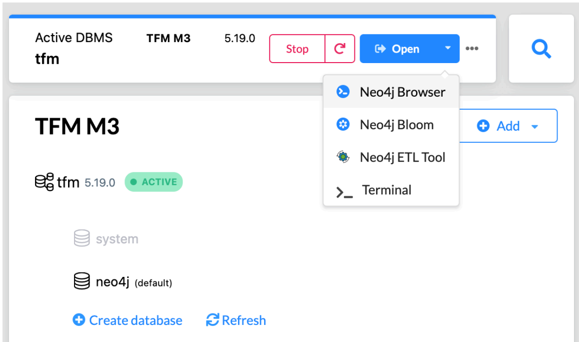
- Once the database is created, we need to install the APOC plugin. Click on the database, then on the right panel click on "Plugins". Open the APOC section and click "Install".



- Change the value of the "`server.directories.import`" setting. Replace the value "import" with the absolute path where the CSV files are located.
- Add a new setting "`db.transaction.timeout=60s`" to prevent any wrongly created query to continue executing after 1 minute. You might need longer execution time for the data load below. In this case set the "`db.transaction.timeout=10m`"



- Start the database. Once started you'll have access to Neo4j Browser and Bloom applications.



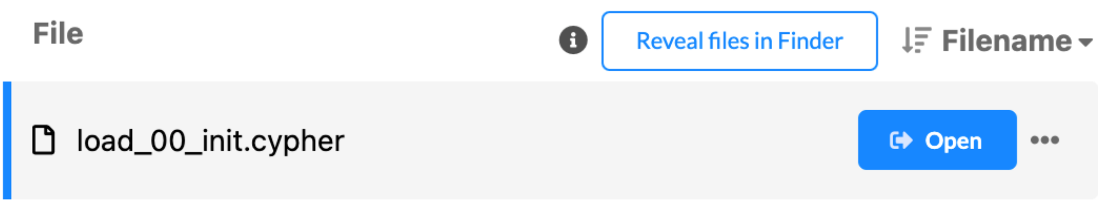
Loading the CSV files

The following CYPHER files are provided for the graph creation from the CSV files. They must be executed in the following sequence.

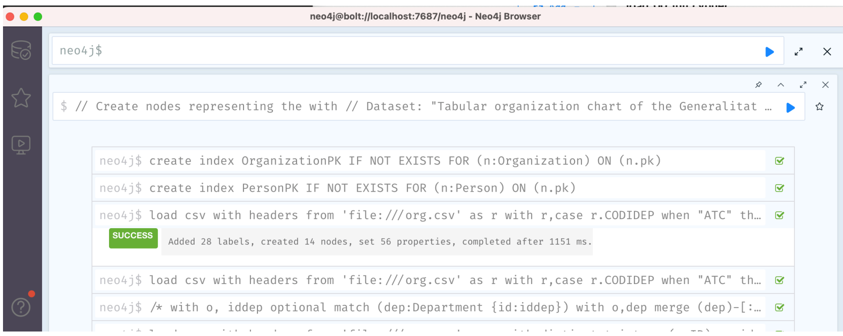
Filename	CSV	Description
load_00_init.cypher		Initialize the database by deleting ALL nodes.
load_10_org.cypher	org.csv	Creates the "Department", "Organization" and "Person" nodes. Creates the CHILD_OF and RESPONSIBLE_OF relationships
load_20_grups.cypher	grups.csv	Creates the "Group" nodes.
load_30_agenda_org.cypher	agenda.csv	Creates additional "Organization" nodes and update properties.

Filename	CSV	Description
load_31_agenda_event.cypher	agenda.csv	Creates "Events", additional "Group" and "Person". Creates the PARTICIPATE relationship.
load_32_agenda_role.cypher	agenda.csv	Creates the REPONSIBLE_OF relationship
load_40_convenis.cypher	convenis.csv	Creates the "Agreement" nodes and the "LINKED_TO" relationship.

To execute each Cypher script click on filename and then the "Open" button.



This will open the script in the Neo4j Browser interface. Click the Run button (blue triangle) and wait for the script execution.
You can see the outcome of each command by clicking on each line in the output window.



- Change the database timeout setting to 60 seconds "`db.transaction.timeout=60s`".

IMPORTANT:

It is possible for the LLM to produce Data Manipulation Language (DML) statements such as **INSERT, MERGE or DELETE** if it is instructed to do so:

This is using the text prompt to send the LLM not a question but instructions to produce a CYPHER statement to manipulate or even delete the complete graph.

One possible solution to protect the system from this specific prompt injection attack is setting a less privileged (read only) connection to Neo4j that would not be authorized to produce any unwanted change on the graph.

It is not in the scope of the POCs to implement any protection, but this is a MUST in the context of the security and privacy of any actual application implementation.

Once all the scripts are successfully executed, a graph with the following schema will be available:

Graph nodes:

- Department: Refers to the structure of the Government of Catalonia.
- Organization: Refers to the structure of the Government of Catalonia.
- Group: Refers to groups of interests who lobbies with the Government. It could be companies, schools, ngo or any other association.
- Person: Refers to public servants or individuals representing groups of interests.
- Event: Refers to events where Government and Groups of interest meet for different reasons.
- Agreement: Refers to formal binding agreements between Government and groups of interests.

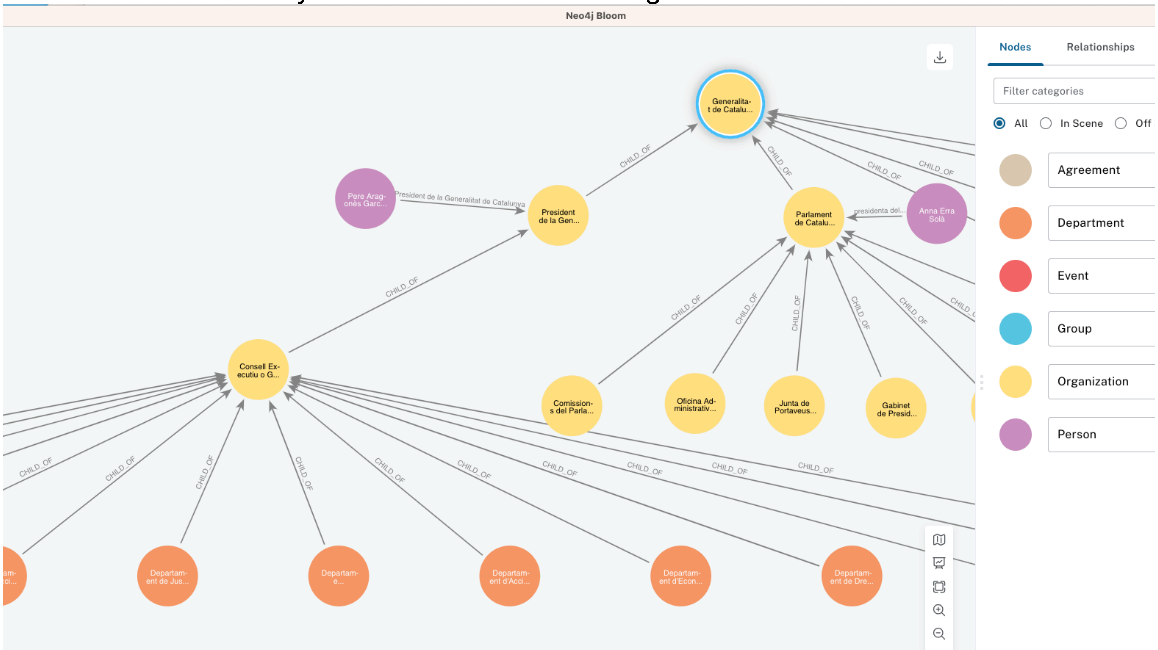
Graph relationships:

- CHILD_OF: Indicates the organization's hierarchy.
- RESPONSIBLE_OF: Indicates who oversees a given organization (role) on a given period.
- PARTICIPATE: Indicates who participated in each event. In the case of the person the "role" property tells the role that the person had at that time.
- LINKED_TO: Connects the different agreements which are related to a master agreement.

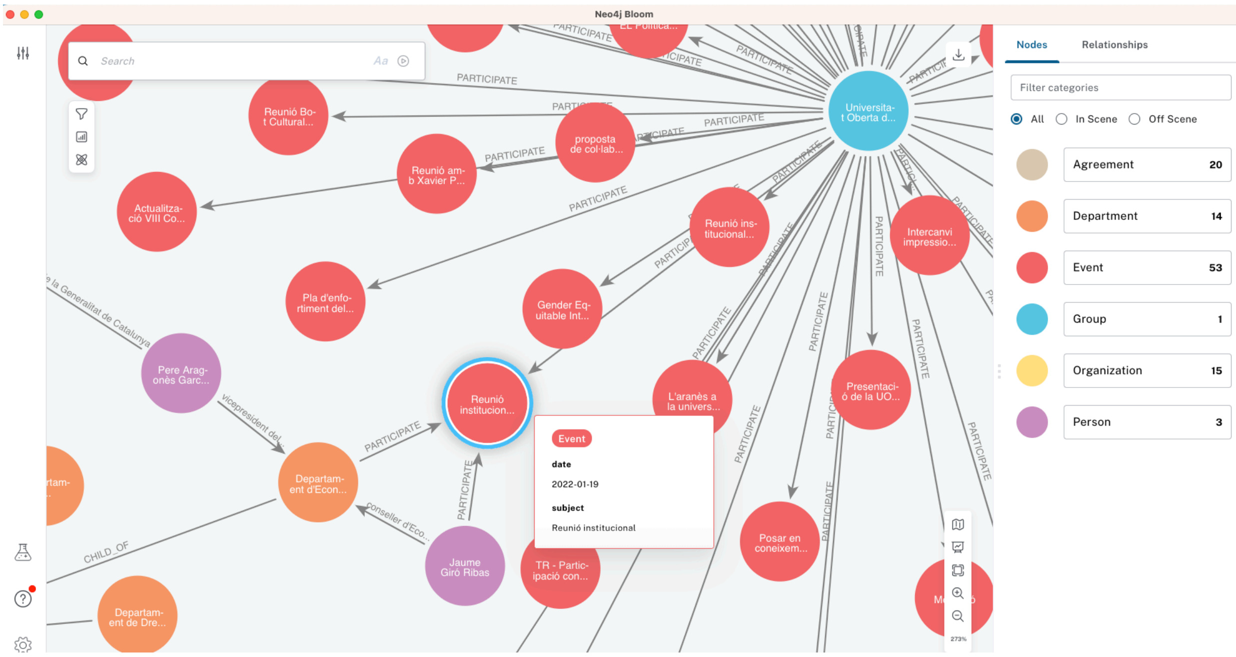
Use Neo4j Browser to execute Cypher queries on the graph, or Neo4j Bloom to navigate interactively.

Some examples of the graph content (Bloom screenshots):

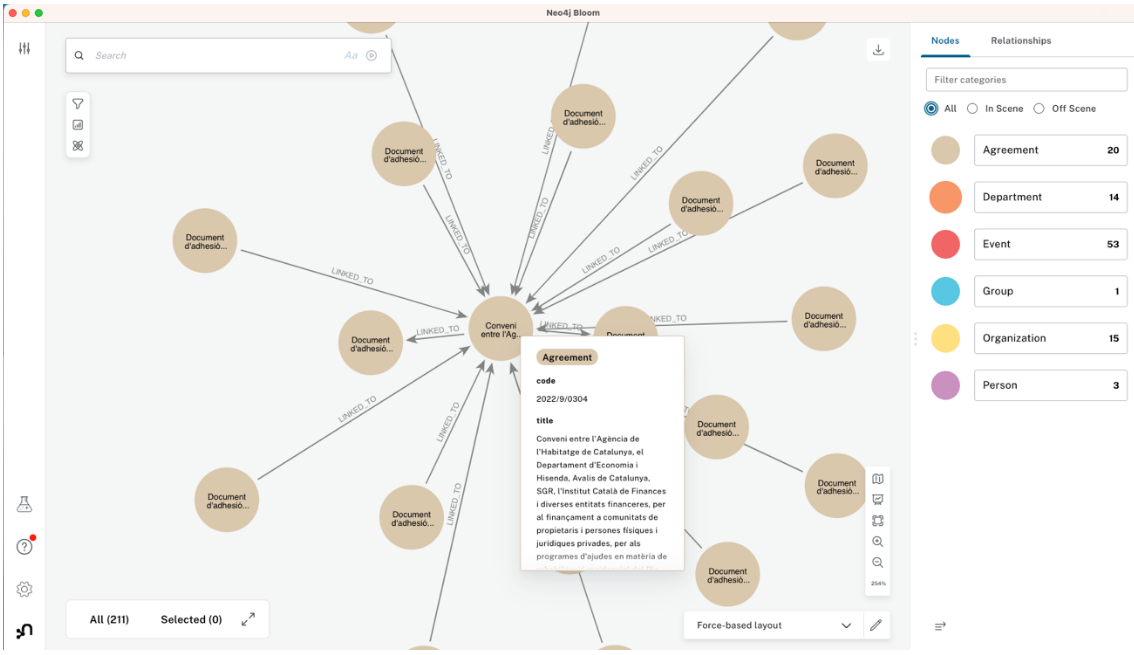
Generalitat de Catalunya and first levels of the Organization:



Meeting between Universitat Oberta de Catalunya and Departament d'Economia:

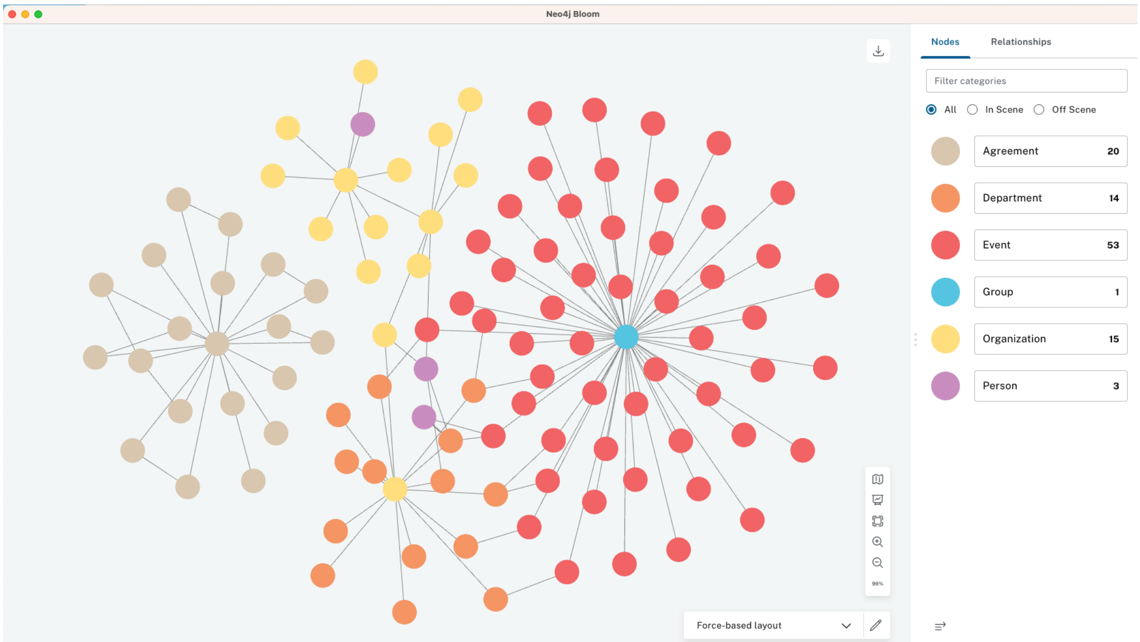


Conveni 2022/9/0304 Agència de l'Habitatge:



A combined view of the previous examples.

Notice that Agreement nodes are isolated from the rest of the graph. This will change because of the POC2.



POC 1: How to query a KG using Natural Language.

In this POC we showcase how prompt engineering techniques using state of the art Large Language Models (LLM) can provide a Knowledge Graph with Natural Language query capabilities.

The Natural Language query chain implements this flow:

1. **Generate Cypher query**: First the question (expressed in natural language) and the Graph schema are sent to the LLM, which returns a Cypher Query as per the prompt instructions.

   We use 3 different prompt templates to try different prompt engineering techniques:

   a. **Default template (template 0)**: Basic instructions.
   b. **Template1**: Template 0 + Hints on how to produce Cypher and 2 few shot samples.
   c. **Template2:** Template 1 + Additional metadata or descriptions of the graph model (meaning of the node labels and relationships) + Additional examples.

   The actual text of the prompt templates is available in the source code.

2. **Execute the generated Cypher code** and retrieve the results in a python dict. The expectation is that the generated code is a query or a MATCH expression. **WARNING**: it is possible to use prompt injection attacks by replacing the question by orders to produce data manipulation (DML) statements. By default the query only retrieves the TOP 10 results. We keep this default to avoid unnecessary tokens consumption.

3. **Produce the answer in Natural language**: The results obtained in the previous step together with the initial question are sent to the LLM with the ask to produce a response to the question using natural language. In this case we use the default `CYPHER_QA_PROMPT` template provided by the `langchain.chains.graph_qa.prompts` module:

```
You are an assistant that helps to form nice and human
understandable answers.
The information part contains the provided information that
you must use to construct an answer.
The provided information is authoritative, you must never
doubt it or try to use your internal knowledge to correct
it.
Make the answer sound as a response to the question. Do not
mention that you based the result on the given information.
Here is an example:

Question: Which managers own Neo4j stocks?
Context:[manager:CTL LLC, manager:JANE STREET GROUP LLC]
```

```
Helpful Answer: CTL LLC, JANE STREET GROUP LLC owns Neo4j
stocks.

Follow this example when generating answers.
If the provided information is empty, say that you don't
know the answer.
Information:
{context}

Question: {question}
Helpful                                          Answer:
```

It is not in the scope of the POC to build an application with chat or Q&A capabilities but to use the necessary building blocks and to measure the performance of these tools on a set of predefined questions.

Instead of an interactive UI, the code in this Notebook performs tests on questions datasets (questions.json) using the indicated configuration (configurations.json). Questions, answers and related metadata are stored in JSON files (poc1_answers folder) for proper analysis.

For a detailed analysis of the interactions/calls with the LLMs, it is recommended to configure LangSmith.

These are the LLMs in scope of the POC:

Vendor	Model	Description
OpenAI	GPT-3.5 Turbo[17]	The latest GPT-3.5 Turbo model with higher accuracy at responding in requested formats and a fix for a bug which caused a text encoding issue for non-English language function calls. Returns a maximum of 4,096 output tokens.
OpenAI	GPT-4o[18]	Our most advanced, multimodal flagship model that's cheaper and faster than GPT-4 Turbo. Currently points to gpt-4o-2024-05-13.
Anthropic	Claude 3 Haiku[19]	Our most powerful model, delivering state-of-the-art performance on highly complex tasks and demonstrating fluency and human-like understanding
Anthropic	Claude 3 Opus[19]	Our fastest and most compact model, designed for near-instant responsiveness and seamless AI experiences that mimic human interactions

[17] https://platform.openai.com/docs/models/gpt-3-5-turbo
[18] https://platform.openai.com/docs/models/gpt-4o
[19] https://docs.anthropic.com/en/docs/models-overview

IMPORTANT

Carefully read the data privacy and any other usage conditions of the used models.

It is NOT in the SCOPE of this POC to manage scenarios where data privacy is a concern when using third party large language models.

The 3 prompt templates by the chosen 4 LLMs gives a total of 12 combinations. To save some time and with the understanding that the result won't be impacted, we test 8 out the 12 possible combinations with the 20 test questions (160 tests):

Model/ template	gpt-3.5-turbo	gpt-4o	claude-3-haiku	claude-3-opus
Default	GPT35_P0	GPT4O_P0		
Template 1	GPT35_P1	GPT4O_P1	HAIKU_P1	OPUS_P1
Template 2		GPT4O_P2		OPUS_P2

The source code of the POC1 is available in the "**poc1 query graph.ipynb**" Jupiter Notebook.[15]

Please examine the documentation and code cells in the Notebook for a detailed understanding of the implementation.

The results of the POC 1 execution are described in the section 4.1 POC1 Performance (Query).

POC 2: How to enrich a KG extracting content from unstructured documents.

Once the initial graph is modeled and the nodes and relationships are created from the structured files, we propose enriching it by extracting additional information from unstructured documents.

This task is known as Named Entity Recognition or NER.

In this POC 2 we showcase how Large Language Models (LLM) can be used to perform NER.

The process consists of sending the extraction instructions (prompt template) including the document content (text only) and the schema of interest. Following the prompt instructions, the model will return the extracted content in the form of a python dict.
The structured output from the LLM is enriched by adding context metadata and then saved into a JSON file. This file is used for model performance analysis and for the actual enrichment of the graph.

It is NOT in the scope of the POC 2:

- Process images or sounds as input for the model.
- Implement a generic Langchain Agent that would categorize the document and proxy it to the corresponding extractor chain / schema. Instead, we will focus on a specific use case or document class.

Use case: Agreements.

To illustrate the end-to-end process from entity identification to graph enrichment, we focus on a specific scenario which is relevant to our graph.
The documents used for this POC 2 are the agreements published in PDF format in the open Registre de Convenis of the Generalitat de Catalunya

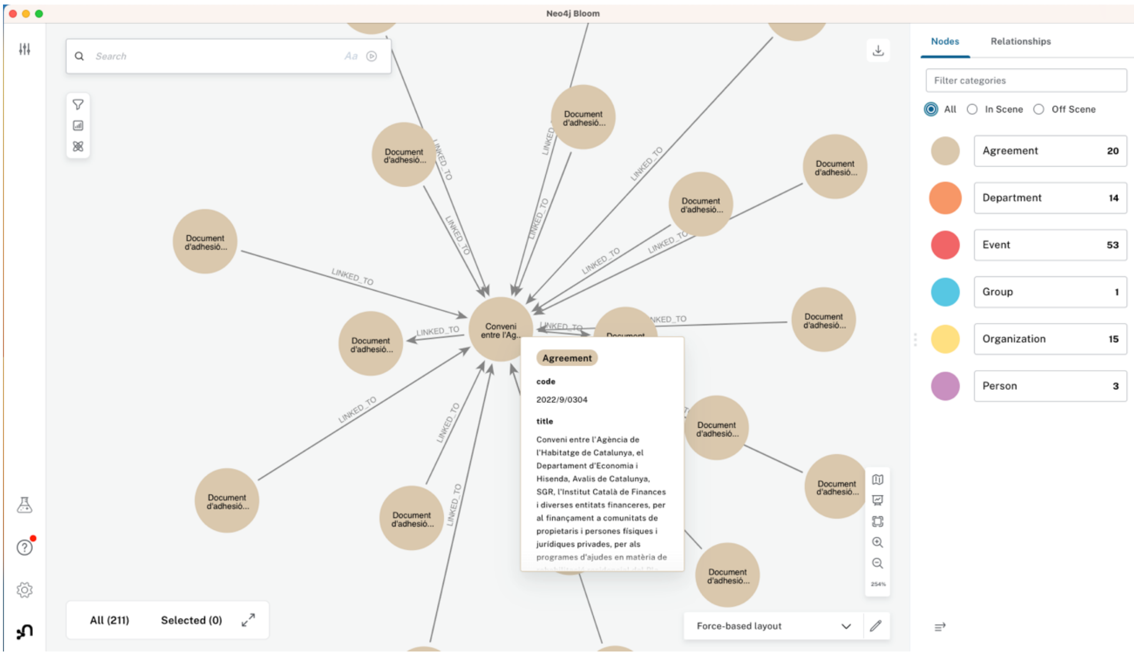
The information about the agreements is already part of the graph in the form of nodes labeled "Agreement". But these nodes are isolated from the other entities as illustrated in the Graph Creation section before.

The goal of this POC 2 is to connect the isolated "Agreement" nodes with the rest of the Graph. More specifically the connection will happen by:

1. Creating necessary new "Person" nodes.
2. Creating the new SIGNED relationship between the "Person" nodes and the agreement.
3. Adding the role, organization and document properties to the SIGNED relationship.
4. Creating new Groups if the organization does not exist already.
5. Creating the new REPRESENT relationship between the Person and the Organization or Group.
6. Creating the SIGNED relationship between the Organization or Group and the Agreement.

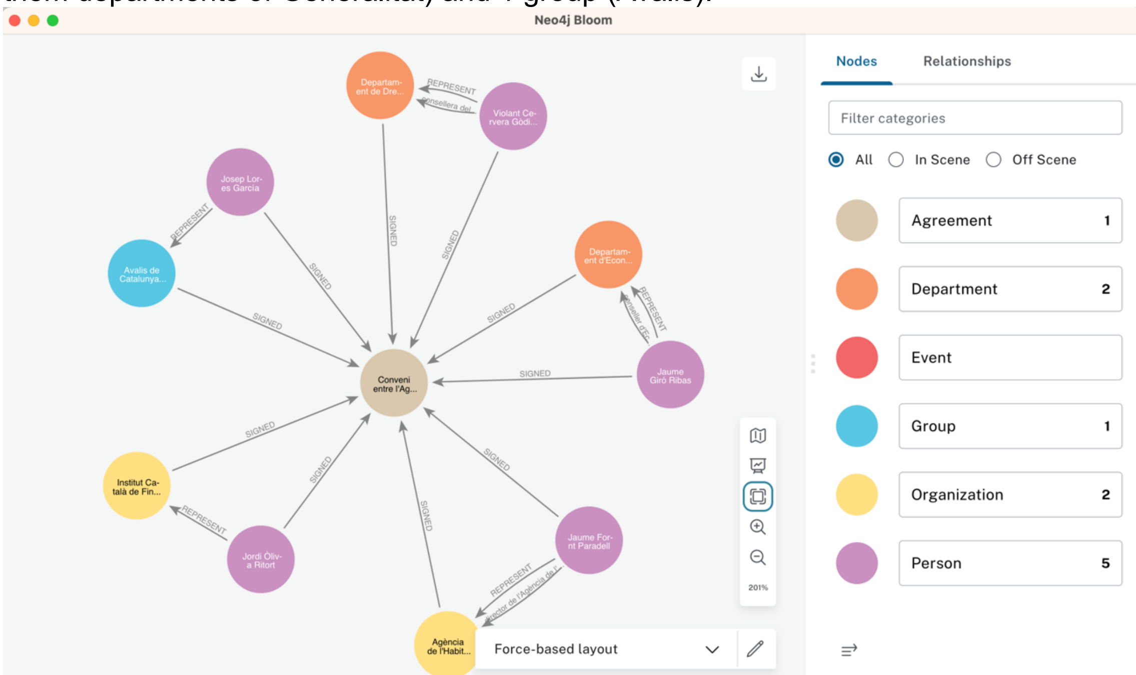
After executing the code for the chosen sample agreement 2022/9/0304 we observe how the graph has effectively enriched:

BEFORE POC 2: The “Agreement” nodes are isolated from the rest of the graph.

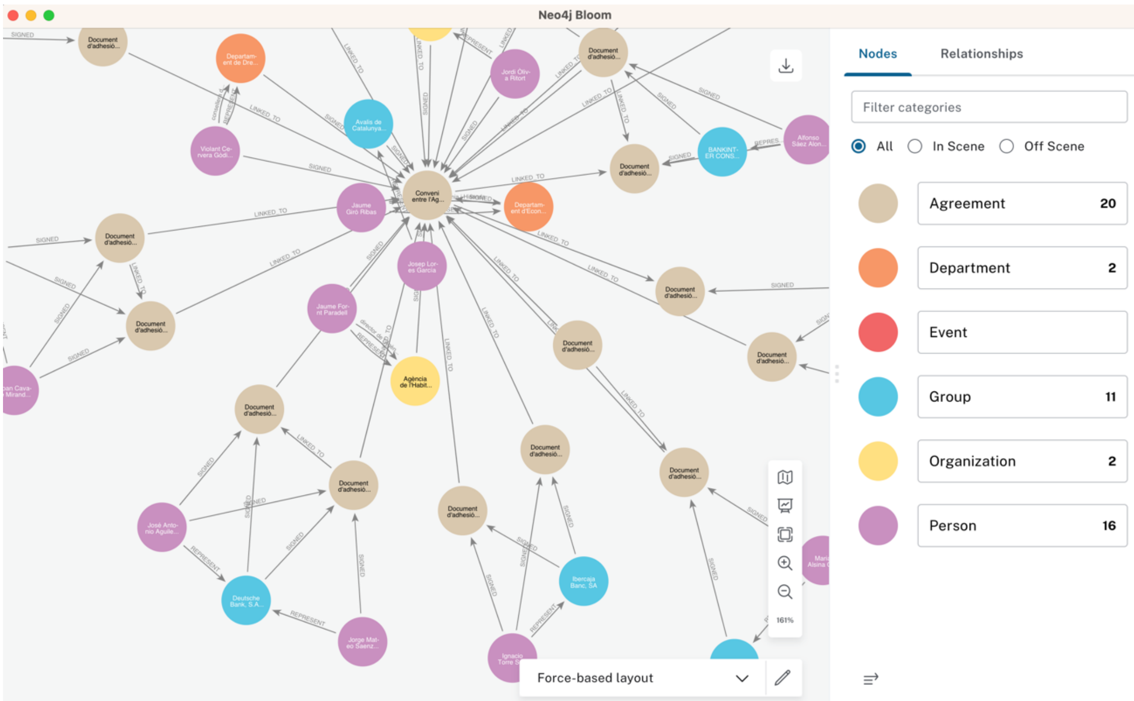


AFTER POC2:

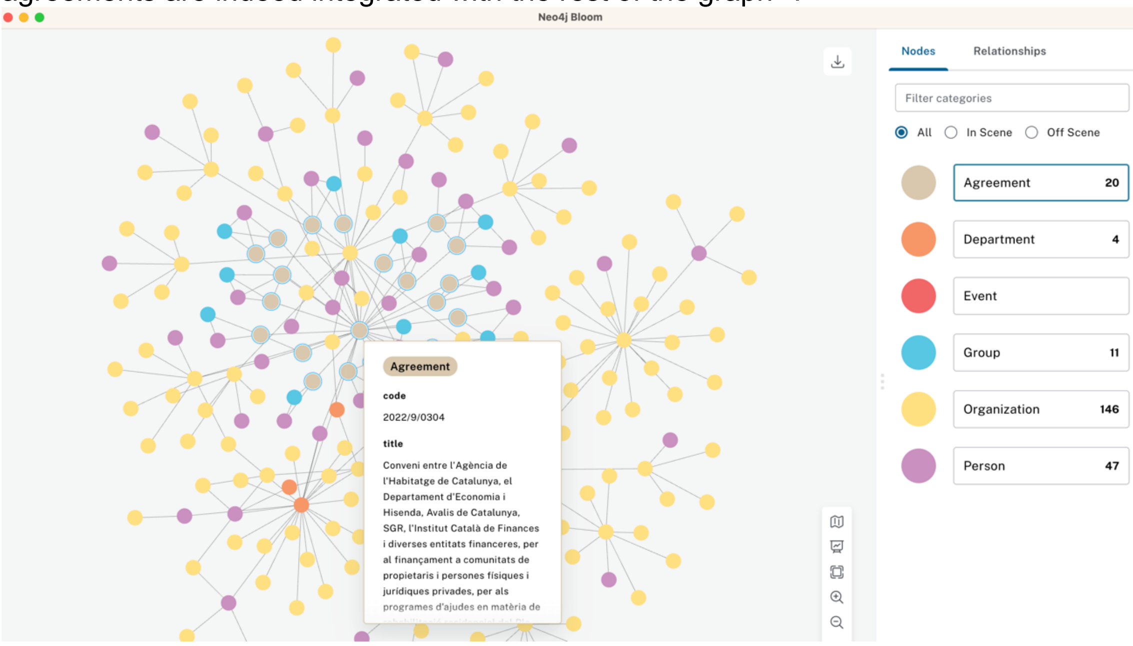
The main agreement is now connected to 5 people and 5 entities: 4 organizations (2 of them departments of Generalitat) and 1 group (Avalis).



The same applies to the agreements linked to the main one. The financial institutions who adhered to the agreement appear as new Group nodes connect to the signees and the agreement:



Extending the visualization to the directly connected nodes gives evidence that the agreements are indeed integrated with the rest of the graph[20].



[20] For the sake of clarity, the "Event" nodes are omitted from the visualization.

Apparently, the proposed LLM based Named Entity Recognition (NER) technique on unstructured documents, can effectively enrich the initially loaded Wirearchy.

The detailed results of the POC2 are presented in the section 4.2 below.

The POC 2 is available in the "**poc2 enrich graph (agreements).ipynb**" Jupiter Notebook.

Please examine the documentation and code cells in the Notebook available in GitHub for a detailed explanation of the implementation. [15]

4. Results

4.1 POC1 Performance (Query)

To measure the performance or capacity of the system to provide useful Natural Language query capabilities we perform a series of manual tests by asking a set of questions using different configurations.

The test consists of the analysis of the results obtained after the execution of the 3 steps described in the previous section:

- Cypher query generation: Is the query syntax correct and align with the question?
- Query execution: Are the results relevant and sufficient?
- Answer in Natural Language: Is the final response answering the question?

Regarding the questions dataset, the possible combinations are quite large and vary based on the pattern complexity. Plus, the fact that we use natural language to perform the question makes the actual number difficult if not impossible to calculate.

We took a heuristic approach here by defining a set of 20 sample questions. These questions are created manually ensuring that all the node labels and relationship types in the graph are involved.

4.1.1 Cypher query evaluation

After manually reviewing the generated Cypher statements in a preliminary round of tests, we determine the criteria to evaluate the correctness or performance of the query by reviewing the main clauses of a cypher query.

Extracts from the Neo4j documentation[21]:

MATCH: The MATCH clause allows you to specify the patterns Neo4j will search for in the database. This is the primary way of getting data into the current set of bindings.

WHERE: MATCH is often coupled to a WHERE part which adds restrictions, or predicates, to the MATCH patterns, making them more specific. The predicates are part of the pattern description and should not be considered a filter applied only after the matching is done. This means that WHERE should always be put together with the MATCH clause it belongs to.

RETURN: The RETURN clause defines the parts of a pattern (nodes, relationships, and/or properties) to be included in the query result.

LIMIT: LIMIT constrains the number of returned rows.

[21] https://neo4j.com/docs/cypher-manual/current/clauses/

To measure the query performance, we determine the following evaluation criteria and a score that can be positive, negative or 0 if the criteria do not apply:

Metric	Cypher Clause	Criteria	Score +/- (*)
M1	MATCH	Identification of nodes and relationships.	1.0
M2		Relationship direction.	0.5
M3		Quantified path patterns.	0.25
M4		Shortest path (few shot sampling).	0.25
W1	WHERE	Filters the proper values (entity location).	0.5
W2		Case insensitive / regexp search. Date.year.	0.75
R1	RETURN	ALL necessary attributes are returned (including those in where clause that give the full query context).	0.5
R2		Descriptive aliases.	0.25
R3		Usage of collect function.	0.25
L1	LIMIT	Usage of LIMIT when asking for ranking.	0.25
E	--	Query is syntactically wrong (will trigger error, timeout or invalid results).	0.0

(*) The score will be (+) POSITIVE when the criteria are met or (-) NEGATIVE when the criteria are NOT met.

In the cases where the generated statement is completely nonsense, the clauses are not scored separately. Instead, the whole query is tagged with an “E” (error with score 0).

The result of the analysis of every question is shown in Table: Evaluation of the POC1 performance by configuration (Metrics). (page 36).

4.1.2 Query results and final answer evaluation

After the query is generated, it is executed against the graph and the results are sent again to the LLM to obtain the final answer. The produced results (JSON data) and the final response (text) returned by the model are also analyzed. All these results and responses are available in the code repository.

The evaluation criteria defined for this step are:

Metric	Criteria	Score
0	No data is provided to the model, or the data is irrelevant to the question.	0.0
1	“Don’t know the answer” because the question context is missing in the data.	-0.5
2	The answer is consistent with the provided data, but the initial question is NOT answered	+0.5
3 (*)	The question is properly answered	+1.0
4	The model has apparently used internal knowledge to resolve ambiguities.	-0.25
5	The model provides a detailed and well-structured answer.	+0.5
6	The model has incorporated hallucinations.	-0.5
7	The answer is correct but partially answered because context is missing in the provided data.	-0.25
8	The model answers assuming that the data is relevant to the question (as per the one-shot sample in the default prompt).	+0.5

(*) The metric 3 plays a special role because it is an indication that the full chain, covering all the steps, was successful. This is used to present the results in the Table: Final assessment of the performance per question and model x template (hits)

Table: Evaluation of the POC1 performance by configuration (Metrics).

The cypher metrics are shown in upper case when the criteria is met (POSITIVE score) or in lower case when it is NOT met (NEGATIVE score). This table presents the evaluation of each question against the 8 scenarios.

	GPT35_P0	GPT4o_P0	GPT35_P1	GPT4o_P1	HAIKU_P1	OPUS_P1	GPT4o_P2	OPUS_P2
Q001	m1 0	m1 0	m1R1R2 0	m1R1R2 0	m1R1R2 2	m1R1R2 2	M1W1R1R2 3	M1W1R1R2 3
Q002	m1M1 34	M1 3	m1M1 34	M1R1R2 3	M1R1 3	M1W1R1R2 3	M1W1R1R2 3	M1W1R1R2 3
Q003	M1R2 3	E 0	M1R2 3	M1R2 3	M1R2 3	M1R2 6	M1W1R1R2 3	M1W1R1R2 35
Q004	E 2	M1W1R1 3	E 2	E 0	R1R2m2M3 2	M1W1R1R2M3 35	M1W1R1R2 3	M1W1R1R2M3 35
Q005	w1 0	M1W1 7	w1 0	M1W1R2 7	M1W1 7	M1W1R2 7	M1W1R1R2 3	M1W1R2 7
Q006	M1W1 3	M1W1 3	m2 0	M1w1R2 7	M1W1R1 35	M1W1R2r1 57	M1W1R2r1 57	M1W1R2r1 57
Q007	E 2	E 2	M1W1R1 3	M1W1R1R2 3	M1W1R1 3	M1W1R1R2 3	M1W1R1R2 35	M1W1R1R2 35
Q008	M1r1 1	M1r1 1	M1W1R2r1 8	M1W1R2r1 1	E 2	M1W1R2r1 8	M1W1R2r1 1	M1W1R2r1 8
Q009	E 0	M1W1r1 1	M1W1R2r1 1	M1W1R2r1 1	E 2	M1W1R2r1 1	M1W1R2r1 8	M1W1R1R2 35
Q010	M1R1R2 3	M1R1R2 3	M1R1R2 3	M1R1R2 3	M1R1R2 35	M1R1R2+R3 35	M1R1R2 3	M1R1R2 35
Q011	M1W1R1 3	E 0	E 0	E 0	m1M1W1R1 36	M1W1R1R2 3	M1W1R1R1R2 3	M1W1R1R1R2 35
Q012	EE 0	E 0	M1W1M4 37	M1W1M4 37	E 0	M1W1M4 3	M1W1M4 37	M1W1M4 3
Q013	M1w1 0	M1W1R1R2 37	M1W1R1R2 36	M1W1R1R2 37	M1W1R1 37	M1W1R1R2 37	M1W1R1R2 3	M1W1R1R2 37
Q014	M1w1 0	M1W1R2 3	M1w1 0	M1W1R2 3	M1W1R2L1 35	M1W1R1L1 35	M1W1R2 3	M1W1W1R2 355
Q015	M1w1 0	E 0	M1M2W1R2 1	M1M2W1R2L1 3	E 0	E 0	M1M2W2R2L1 3	M1M2W2R2L1 3
Q016	M1w1 0	M1W1R2 3	M1W1L1 3	M1W1R2 3	M1W1L1 3	M1W1R2L1 3	M1W2R2 3	M1W2R2 35
Q017	M1w1 0	M1W1R2 3	M1m2W1 0	M1W1R1R2 3	M1m2W1 0	M1W1R1R2 35	M1W1R1R2 3	M1W2R1R2 35
Q018	E 0	M1 W1	M1W1 1	M1W1R2 1	M1W1R2 8	M1W1R1R2 3	M1W2R2 1	M1W1R1R2 3
Q019	E 0	E 0	E 0	E 0	M1M2W1R1R2 3	E 0	E 0	E 0
Q020	E 0	E 2	E 0	E 0	E 0	E 0	E 0	E 0

Table: Evaluation of the POC1 performance by configuration (Scores).

Same a previous table after assigning the score given to each metric.

	GPT35_P0	GPT4o_P0	GPT35_P1	GPT4o_P1	HAIKU_P1	OPUS_P1	GPT4o_P2	OPUS_P2	average
Q001	0.50	0.50	1.25	1.25	1.75	1.75	3.25	3.25	**1.688**
Q002	1.25	2.00	1.25	2.75	2.50	3.25	3.25	3.25	**2.438**
Q003	2.25	0.00	2.25	2.00	2.25	0.75	3.25	3.75	**2.062**
Q004	0.50	3.00	0.50	0.00	1.00	4.00	3.25	4.00	**2.031**
Q005	-0.50	1.25	-0.50	1.50	1.25	1.50	3.25	1.50	**1.156**
Q006	2.50	2.50	-0.50	0.50	3.50	1.50	1.50	1.50	**1.625**
Q007	0.50	0.50	3.00	3.25	3.00	3.25	3.75	3.75	**2.625**
Q008	0.00	0.00	1.75	0.75	0.50	1.75	0.75	1.75	**0.906**
Q009	0.00	0.50	0.75	0.75	0.50	0.75	1.75	3.75	**1.094**
Q010	2.75	2.75	2.75	2.75	3.25	3.50	2.75	3.25	**2.969**
Q011	3.00	0.00	0.00	0.00	3.00	3.25	3.75	4.25	**2.156**
Q012	0.00	0.00	2.50	2.50	0.00	2.75	2.50	2.75	**1.625**
Q013	0.50	3.00	2.75	3.00	2.75	3.00	3.25	3.00	**2.656**
Q014	0.50	2.75	0.50	2.75	3.50	3.75	2.75	4.25	**2.594**
Q015	0.50	0.00	1.75	3.50	0.00	0.00	3.75	3.75	**1.656**
Q016	0.50	2.75	2.75	2.75	2.75	3.00	3.00	3.50	**2.625**
Q017	0.50	2.75	1.00	3.25	1.00	3.75	3.25	4.00	**2.438**
Q018	0.00	3.50	1.00	1.25	2.25	3.25	1.50	3.25	**2.000**
Q019	0.00	0.00	0.00	0.00	3.75	0.00	0.00	0.00	**0.469**
Q020	0.00	0.50	0.00	0.00	0.00	0.00	0.00	0.00	**0.062**
Total	**15.25**	**28.25**	**24.75**	**34.50**	**38.50**	**44.75**	**50.50**	**58.50**	

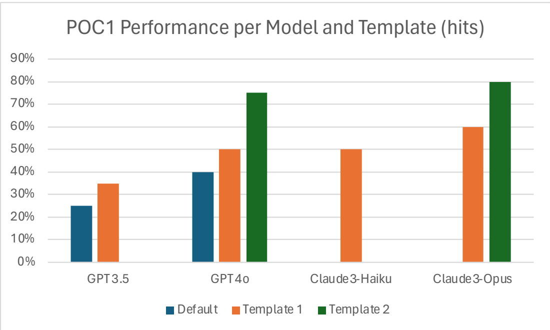
Table: Final assessment of the performance per question and model x template (hits)

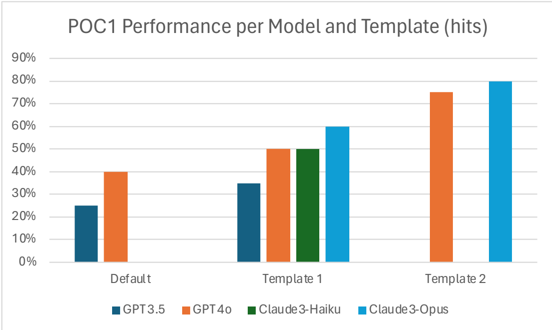
	GPT35_P0	GPT4o_P0	GPT35_P1	GPT4o_P1	HAIKU_P1	OPUS_P1	GPT4o_P2	OPUS_P2
Q001							✓	✓
Q002	✓	✓	✓	✓	✓	✓	✓	✓
Q003	✓		✓	✓	✓		✓	✓
Q004		✓				✓	✓	✓
Q005							✓	
Q006	✓	✓			✓	✓	✓	✓
Q007			✓	✓	✓	✓	✓	✓
Q008								
Q009								✓
Q010	✓	✓	✓	✓	✓	✓	✓	✓
Q011	✓				✓	✓	✓	✓
Q012			✓	✓		✓	✓	✓
Q013		✓	✓	✓	✓	✓	✓	✓
Q014		✓		✓	✓	✓	✓	✓
Q015				✓			✓	✓
Q016		✓	✓	✓	✓	✓	✓	✓
Q017		✓		✓		✓	✓	✓
Q018						✓		✓
Q019					✓			
Q020								
Total	**5**	**8**	**7**	**10**	**10**	**12**	**15**	**16**

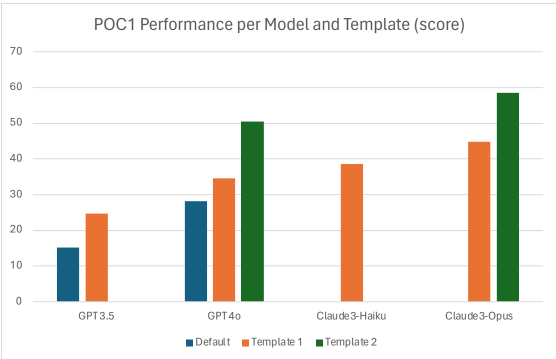
Results visualization and conclusions

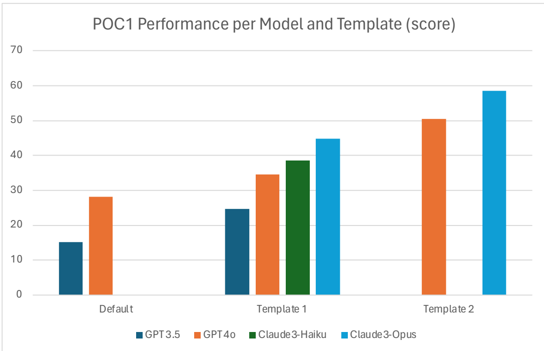
Results of POC1 are presented in aggregated format:

- HITS: Measurement of the effectiveness of the process. Is it capable to produce the expected output?
- Score: How is the quality of the results? This metric tries to add a quality dimension to the HITS. This is because the same question could be interpreted and executed in slightly different manners yielding equivalent results.
  Higher scores suggest higher model performance and eventually better results in the future.









The analysis of the previous data throws two clear conclusions:

- Prompt engineering relevance: This conclusion is consistent with the LLM paradigm. It is paramount to design adequate prompts for the model to produce the desired results. This is relevant regardless of the chosen model. It can be observed that the more context if provided to the model, the better the performance.
- LLMs singularity: As expected, different models show different behaviors. Larger and more expensive models show better performance. We observed a slightly higher performance of the Claude offering versus OpenAI.
  Still, the stochastic nature of the Large Language Model makes difficult to assert that this will be always the case for any given query. In fact, in questions Q005 and Q015 Claude is not the best performer.

4.2 POC2 Performance (Graph Enrichment):

To evaluate the performance of the graph enrichment task, we identified a relevant agreement within the “convenis.csv” file.

The master agreement with code “2022/9/0304”[22] consists of one core agreement linked to 19 agreements which totals 20 documents.

In this POC 2 we want to evaluate the Named Entity Recognition (NER) capability of the models. The information we are interested on are the following attributes from the signees of each document:

Name: Full name of the signee.
Organization: The name of the organization that the person represents.
Role: The role of the person within the organization.
Document: The document proving the person's role within the organization.

Notice that these are precisely the attributes defined in the Signee class:

```
class Signee(BaseModel):
    """Information about the person who signs the agreement on behalf of an organization."""

   name: description="The name of the person"
   organization: description="The name of the organization that the person represents if known"
   role: description="The role of the person within the organization if known"
   document: description="The document proving the person's role within the organization if known"
```

To perform the evaluation, we manually extract the “ground truth” from the documents and then compare it with the information extracted by the LLMs. The extracted information is available in the JSON files within the “poc2_extracts” folder in Github.

When checking the extracted attribute value, we perform an exact comparation with the ground truth. For instance, in the case of the name attribute we expect the person’s full name to be extracted.

The below table lists the ground truth and the evaluation results of each document per model.

22 https://registredeconvenis.gencat.cat/drep_rccc/public/Convenis.do?accion=DownloadDocumentsConveni&numFila=0&numConveni=2022/P/0126

Table: Evaluation of the POC2 performance by LLM (Metrics).

Legend for the models' column:

	Name (people)	Organization	Role	Document
Extracted	✓---	-✓--	--✓-	---✓
NON-Extracted	✗---	-✗--	--✗-	---✗

Agreement	Ground Truth	GPT 35 turbo	GPT 4o	claude 3 - haiku	claude 3 - opus
2022/8/0004	Arquia Bank, SA Raimon Royo Uño Director General Adjunt escriptura atorgada davant el notari, en data 28 de juliol de 2017, número del seu protocol 1.631, i inscrita al Registre Mercantil de Barcelona, en data 25 d'agost de 2017, en el volum 46.011, foli 220, full B-2.363, inscripció 308a.	✓✓✓✗	✓✓✓✓	✓✓✓✗	✓✓✓✓
2022/8/0005	Banc de Sabadell, SA David Massana Gracia apoderat segons escriptura atorgada davant el notari, en data 25 de juliol de 2019, número del seu protocol 2.774, i inscrita al Registre Mercantil d'Alacant, en data 16 d'agost de 2019, en el volum 4.070, foli 85, full A-156.980, inscripció 164a.	✓✗✗✗	✓✓✓✓	✓✓✓✗	✓✓✓✗
2022/8/0006	BANC SANTANDER, SA 1) David Baños Baeza 2) Garcia (broken name showing in the spanish version) apoderats mancomunats escriptura atorgada davant el notari , en data 21 d'abril de 1998, número del seu protocol 1.316, i inscrita al Registre Mercantil de Cantabria, en data 25 de maig de 1998, en el volum 611, foli 68, full S-1.960, inscripció 402a, i segons escriptura atorgada davant el notari , en data 10 d'abril de 2013, número del seu protocol 2.287, i inscrita al Registre Mercantil de Cantabria, en el volum 1.006, foli 220, full S-1.960, inscripció 2.312a, respectivament	1) ✓✓✓✓ 2) ✗✓✓✗	1) ✓✓✓✗ 2) ✗✗✗✗	1) ✓✓✓✗ 2) ✗✓✓✗	1) ✓✓✓✗ 2) ✓✓✓✗

Agreement	Ground Truth	GPT 35 turbo	GPT 4o	claude 3 - haiku	claude 3 - opus
2022/8/0007	BANKINTER CONSUMER FINANCE EFC, SA Alfonso Sáez Alonso-Muñumer Conseller-Director General escriptura atorgada davant el notari , en data 23 de juliol de 2014, número del seu protocol 2.370, i inscrita al Registre Mercantil de Madrid, en data 29 de juliol de 2014, en el volum 27.322, foli 176, full M-259.543, inscripció 82a.	✓✓✓✗	✓✓✓✓	✓✓✓✗	✓✓✗✗
2022/8/0008	BANC BILBAO VIZCAYA ARGENTARIA, SA José Raúl Pérez González de Uriarte apoderat escriptura atorgada davant el notari, en data 3 de desembre de 2020, número del seu protocol 2216, i inscrita al Registre Mercantil de Bizkaia, en data 9 de desembre de 2020, en el volum 5.928, foli 153, full BI-17-A, inscripció 4.164a	✓✓✗✓	✓✓✓✓	✓✓✓✗	✓✓✓✗
2022/8/0009	CAIXA DE CRÈDIT DELS ENGINYERS-CAJA DE CRÉDITO DE LOS INGENIEROS, S.COOP. DE CRÈDIT Joan Cavallé Miranda Director General escriptura atorgada davant el notari , en data 20 de gener de 2006, número del seu protocol 96, i inscrita al Registre Mercantil de Barcelona, en data 7 de març de 2006, en el volum 21.606, foli 161, full B-25.121, inscripció 177a.	✗✓✓✗	✓✓✓✗	✓✓✓✗	✓✓✓✗ (several noise)
2022/8/0010	DEUTSCHE BANK, SAE 1) José Antonio Aguilera Núñez 2) blank -> JORGE MATEO SAENZ DE MIERA (page signature) apoderats mancomunats segons escriptura atorgada davant el notari, en data 20 de desembre de 1996, número del seu protocol 4281, i inscrita al Registre Mercantil de Barcelona, en data 23 de gener de 1997, en el volum 26.516, foli 210, full B-2.861, inscripció 2.782a, i segons escriptura atorgada davant el notari, en data 13 de març de 2007, número del seu protocol 914, i inscrita al Registre Mercantil de Barcelona, en data 5 d'abril de 2007, en el volum 39.491, foli 14, full B-2.861, inscripció 4.387a, respectivament.	1) ✓✓✓✗ 2) ✓✗✗✗	1) ✓✓✓✗ 2) ✓✗✗✗	1) ✓✓✓✗ 2) ✗✓✓✗	1) ✓✓✓✗ 2) ✗✓✓✗
2022/8/0011	Ibercaja Banc, SA Ignacio Torre Sola Director de Màrqueting i Estratègia Digital segons escriptura atorgada davant el notari, en data 15 de març de 2017, número del seu protocol 589, i inscrita al Registre Mercantil de Saragossa, en data 28 de març de 2017, en el volum 4.176, foli 222, full Z-52.186, inscripció 1.225a.	✓✓✓✓	✓✓✓✓	✓✓✓✗	✓✓✓✗
2022/8/0012	UNIÓ DE CRÈDITS IMMOBILIARIS, SA, ESTABLIMENT FINANCER DE CRÈDIT Roberto Colomer Blasco Director General segons escriptura atorgada davant el notari, en data 28 de març de 1990, número del seu protocol 1.076, i inscrita al Registre Mercantil de Madrid, en data 1 de juny de 1990, en el volum 9876, foli 146,full M-67.739, inscripció 35a.	✓✓✓✗	✓✓✓✗	✓✓✓✗	✓✓✓✗

Agreement	Ground Truth	GPT 35 turbo	GPT 4o	claude 3 - haiku	claude 3 - opus
2022/8/0013	THIS DOCUMENT DOES NOT CONTAIN TEXT BUT AN IMAGE OF AN SCANNED DOCUMENT (**OUT OF SCOPE**) CaixaBank, SA Maria Alsina Call apoderada segons escriptura atorgada davant el notari…	✗✗✗✗	✗✗✗✗	✗✗✗✗	✗✗✗✗
2022/9/0304	1) Jaume Giró i Ribas, conseller del Departament d'Economia i Hisenda, nomenat pel Decret 22/2021, de 26 de maig (DOGC 8418A, 26.5.2021). 2) Violant Cervera i Gòdia, consellera del Departament de Drets Socials, nomenada pel Decret 22/2021, de 26 de maig (DOGC 8418A, 26.5.2021). 3) Jaume Fornt i Paradell, director de l'Agència de l'Habitatge de Catalunya, designat per l'Acord de Govern 93/2020, de 14 de juliol, i de conformitat amb les funcions atribuïdes a l'article 7.3 e) de la Llei 13/2009 de 22 de juliol, de l'Agència de l'Habitatge de Catalunya, entitat adscrita al Departament de Drets Socials. 4) Jordi Òliva i Ritort, en representació de l'Institut Català de Finances (en endavant, ICF), en qualitat de conseller delegat, nomenat per l'Acord de Govern GOV/175/2021, de 2 de novembre (DOGC núm. 8536, de 4 de novembre de 12021) 5) Josep Lores i García, en representació d'Avalis de Catalunya, SGR (en endavant, Avalis), en qualitat de conseller delegat, en virtut dels poders atorgats a favor seu mitjançant escriptura pública atorgada davant el notari de Barcelona, , en data 10 de juny de 2021, amb número de protocol 2.454, inscrita al Registre Mercantil de Barcelona. 6) els representants de les entitats financeres que s'adhereixin al present Conveni, mitjançant la signatura del document d'adhesió que consta com a annex del present Conveni.	1) ✓✓✗✗ 2) ✓✓✗✗ 3) ✓✓✗✗ 4) ✓✓✗✗ 5) ✓✓✗✗	1) ✓✓✓✓ 2) ✓✓✓✓ 3) ✓✓✓✓ 4) ✓✓✓✓ 5) ✓✓✓✓	1) ✓✓✓✗ 2) ✓✓✓✗ 3) ✓✓✓✗ 4) ✓✓✓✗ 5) ✓✓✓✗	1) ✓✓✓✗ 2) ✓✓✓✗ 3) ✓✓✓✗ 4) ✓✓✓✗ 5) ✓✓✓✗
2024/8/0047	Arquia Bank, SA Raimon Royo Uño Director General Adjunt segons escriptura atorgada davant el notari, en data 28 de juliol de 2017, número del seu protocol1.631,i inscrita al Registre Mercantil de Barcelona, en data 25 d'agost de 2017, en el volum 46.011, foli 220, full B-2.363, inscripció 308a. Aquest apoderament va ser ratificat segons escriptura atorgada davant el notari , en data 6 d'octubre de 2017 i inscrita al Registre Mercantil de Madrid, en el volum 36.871, foli 136, full M-659.820, inscripció 36a.	✗✓✓✗	✓✓✓✓	✓✓✓✗	✓✓✓✗

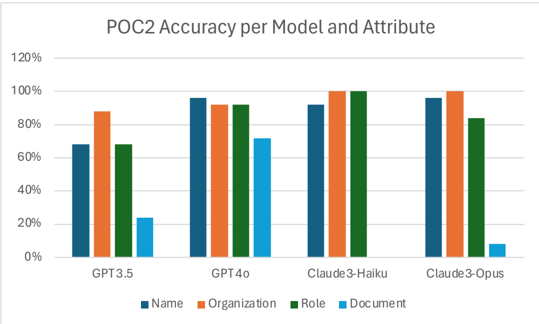
Agreement	Ground Truth	GPT 35 turbo	GPT 4o	claude 3 - haiku	claude 3 - opus
2024/8/0048	Banc de Sabadell, SA David Massana Gracia apoderat segons escriptura atorgada davant el notari , en data 25 de juliol de 2019, número del seu protocol2.774,iinscritaalRegistreMercantil d'Alacant, en data 16 d'agost de 2019, en el volum 4.070, foli 85, full A-156.980, inscripció 164a.	✗✓✓✓	✓✓✓✓	✓✓✓✗	✓✓✓✓
2024/8/0049	BANC BILBAO VIZCAYA ARGENTARIA, SA José Raúl Pérez González de Uriarte apoderat segons escriptura atorgada davant el notari en data 3 de desembre de 2020, número del seu protocol 2216, i inscrita al Registre Mercantil de Bizkaia, en data 9 de desembre de 2020, en el volum 5.928, foli 153, full BI-17-A, inscripció 4.164a.	✗✓✓✗	✓✓✓✓	✓✓✓✗	✓✓✗✗
2024/8/0050	CAIXA DE CRÈDIT DELS ENGINYERS-CAJA DE CRÉDITO DE LOS INGENIEROS, S.COOP. DE CRÈDIT Joan Cavallé Miranda Director General segons escriptura atorgada davant el notari , en data 20 de gener de 2006, número del seu protocol96,iinscritaalRegistreMercantilde Barcelona, en data 7 de març de 2006, en el volum 21.606, foli 161, full B-25.121, inscripció 177a.	✓✓✓✗	✓✓✓✓	✓✓✓✗	✓✓✓✗
2024/8/0051	DEUTSCHE BANK, SAE José Antonio Aguilera Núñez Jorge Mateo Saenz de Miera Alonso apoderats mancomunats segons escriptura atorgada davant el notari, en data 20 de desembre de 1996, número del seu protocol 4281, i inscrita al Registre Mercantil de Barcelona, en data 23 de gener de 1997, en el volum 26.516, foli 210, full B-2.861, inscripció 2.782a, i segons escriptura atorgada davant el notari en data 13 de març de 2007, número del seu protocol 914, i inscrita al Registre Mercantil de Barcelona, en data 5 d'abril de 2007, en el volum 39.491, foli 14, full B-2.861, inscripció 4.387a, respectivament.	✗✗✓✗	✓✓✓✗	✓✓✓✗	✓✓✗✗
2024/8/0052	Ibercaja Banc, SA Ignacio Torre Sola Director de Màrqueting i Estratègia Digital segons escriptura atorgada davant el notari, en data 15 de març de 2017, número del seu protocol 589, i inscrita al Registre Mercantil de Saragossa, en data 28 de març de 2017, en el volum 4.176, foli 222, full Z-52.186, inscripció 1.225a.	✗✓✓✓	✓✓✓✓	✓✓✓✗	✓✓✓✗

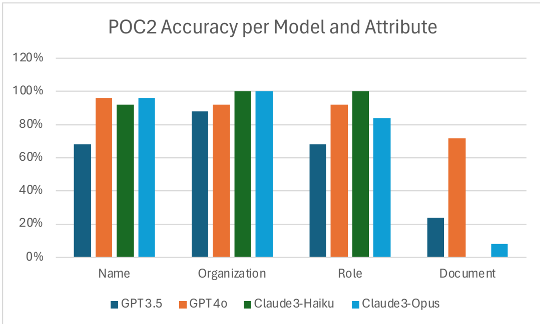
Agreement	Ground Truth	GPT 35 turbo	GPT 4o	claude 3 - haiku	claude 3 - opus
2024/8/0053	UNIÓ DE CRÈDITS IMMOBILIARIS, SA, ESTABLIMENT FINANCER DE CRÈDIT Philippe Jacques Laporte Director d'operacions (COO) según escritura otorgada ante el notario en fecha 19 de junio de 1997, número de su protocolo 2.859.	✓✗✓✗	✓✓✓✓	✓✓✓✗	✓✓✓✗
2024/8/0055	CaixaBank, SA Maria Alsina Call apoderada segons escriptura atorgada davant el notari , en data 16 d'octubre de 2017, número del seu protocol2.722,i inscrita al Registre Mercantil de València, en data 6 de novembre de 2017, en el volum 10.370, foli1, full V- 178.351, inscripció 3a.	✗✓✓✓	✓✓✓✓	✓✓✓✗	✓✓✗✗
2024/8/0056	BANKINTER CONSUMER FINANCE EFC, SA Alfonso Saez Alonso-Muñumer Conseller-Director General segons escriptura atorgada davant el notari, en data 23 de juliol de 2014, número del seu protocol 2.370, i inscrita al Registre Mercantil de Madrid, en data 29 de juliol de 2014, en el volum 27.322, foli 176, full M-259.543, inscripció 82a.	✓✓✓✗	✓✓✓✓	✓✓✓✗	✓✓✓✗

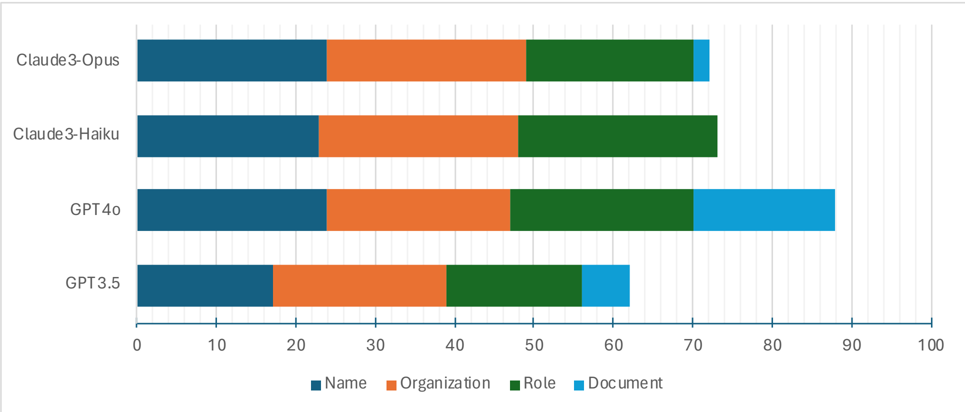
Summary of the evaluation results presented by Model and Attribute. Hits or correct extractions in absolute numbers and percentage:

Attribute	**GPT 3.5**	**GPT 4o**	**Haiku**	**Opus**
Name	17	24	23	24
Organization	22	23	25	25
Role	17	23	25	21
Document	6	18	0	2
ALL	**62**	**88**	**73**	**72**

GPT 3.5	**GPT 4o**	**Haiku**	**Opus**
68%	96%	92%	96%
88%	92%	100%	100%
68%	92%	100%	84%
24%	72%	0%	8%
62%	**88%**	**73%**	**72%**







We observe a great performance in all attributes but “Document” which is a complicated one given the nature of the documents. Still GPT-4o does a great job here extracting it correctly in 72% of the cases, way higher than the 8% exhibit by Opus.

The above results plus the cost element, with Opus being between 3 to 5 times more expensive, makes GPT-4o the best option for this task.

5. Conclusion and future work

5.1 Conclusion

The first conclusion is that knowledge graphs are a very convenient option to model a "Wirearchy". They prove very versatile, so it is easy to incorporate additional content to enrich an existing model. Complex queries can be expressed using their rich query language which make capable to get new insights from the graph. On top of that, graph visualization tools help to understand and navigate the data and their relationships in a way that is not possible in traditional relational / SQL based systems.

Regarding the capability of LLMs for the two proposed activities, we conclude that:

5.1.1 LLMs for Natural Language query:

The first conclusion is that LLMs show a huge potential to implement a Q&A system using Natural Language, but this still comes with several caveats.

On the one side we observe that LLMs are increasing their capabilities to manage larger and more complex prompts. This allows the application of advanced Retrieval Augmented Generation (RAG) techniques such as providing few shot examples or enriching the graph model with additional metadata to improve the LLM performance.

The performance of the models is correlated with:

- Prompt template design: It is paramount to provide a well-crafted prompt that provides the relevant instructions to guide the LLM models.
- Graph design: The graph design should be simple and unambiguous. The node labels, relationship types and directions and all the properties should accurately reflect the actual content.
- Time dimension: It is important to use consistent data types and names to refer to event dates or validity periods.

Another important aspect to consider is that prompt engineering techniques should be reflected in the language used when posting questions. In other words, the more information and clear instructions are provided to the model the more likelihood that the answer will be accurate.

In any case the system will **require human supervision**, meaning that it could be useful in the context of a copilot tool for users who understand the underlying graph model and can identify situations where the LLM is producing wrong results. Opening such solution to a general audience is risky because the results might look plausible and still being completely wrong.

5.1.2 LLMs for KG augmentation:

The results in this task are quite promising. The models show a great performance even in confusing scenarios like the bilingual documents where the document's content is intertwined.

There is one relevant caveat when it comes to the implementation though. The model is presented with both the document content and the schema of the required structured output simultaneously which should be consistent.
This means that certain understanding of the document's structure is required prior to the NER step. Same is applicable when integrating the new knowledge into the graph, the cypher code used in the chosen use case is tailor made.

This can be challenging if we aim to automate a generic task able to process different document types.

Another challenge is about managing duplicates. This is to identify whether an entity extracted from a document refers to a new or an existing entity the graph when there are minor differences in the identifiers such as with people names. This is in fact a generic data quality / ETL related task where LLMs (embeddings) could be of assistance. This was not in the scope of the POC.

A final note of warning when it comes to the cost of the LLMs, notice that there are huge differences in the cost of the different models used in the POCs. We can expect costs lowering in the future, as it was already the case with GPT4, together with smaller models being more capable again reducing the cost element.

5.2 Future work

With this thesis we have just briefly touched the knowledge graph capabilities and the potential of the emerging LLM technologies in the scope of the initial problem statements.

It is amazing seeing how vibrant the emerging world of LLMs is. The big players are delivering new generic models rapidly while thousands of fine-tuned models are already available.

The major graph vendors are already incorporating LLMs in their products with the promising slogan of “talk to your data”. The AI “copilots” are the current must-have capability.

Here a proposal of future lines of work:

- Include the descriptions of the new relationships created by POC2 in the prompt template.
- Test other LLMs.
- Fine tune or train models for custom metadata.
- Add lang chains from other graph vendors. E.g. Amazon Neptune.
- Refine the graph Model to eliminate some ambiguities.
- Create a more detailed template with additional few shot examples.
- Send questions in English or any other language.
- Develop a more capable QA agent rather than one step question/chain. E.g. extract a subgraph schema to be presented to the model before the cypher generation.
- Develop a more capable NER agent able to deal with multiple document types.
- Integrate other RAG techniques such as embeddings or preprocessing tasks.
- Incorporate entity deduplication artifacts.
- Incorporate security artifacts such as RBAC and fine-grained privileges to secure the data privacy.

Hope you can take some of the above actions and enjoy as much as I did.

6. References

[1] What is wirearchy? - Wirearchy. (2023, December 9). https://web.archive.org/web/20231209183821/http://wirearchy.com/what-is-wirearchy/

[2] Britt. (2014, December 22). Wirearchy as leadership concept. Learning In a Flat World. https://bwatwood.edublogs.org/2014/12/22/wirearchy-as-leadership-concept-2/

[3] Pan, Shirui, Luo, Linhao, Wang, Yufei, Chen, Chen, Wang, Jiapu, & Wu, Xindong. (2023). Unifying large language models and knowledge graphs: A roadmap. https://doi.org/10.48550/ARXIV.2306.08302

[4] Taffa, Tilahun Abedissa, & Usbeck, Ricardo. (2023). Leveraging llms in scholarly knowledge graph question answering(arXiv:2311.09841). arXiv. http://arxiv.org/abs/2311.09841

[5] Abu-Salih, Bilal. (2021). Domain-specific knowledge graphs: A survey. Journal of Network and Computer Applications, 185, 103076. https://doi.org/10.1016/j.jnca.2021.103076

[6] Making sense of data with rdf* vs. Lpg. (s.d.). OpenCredo. Retrieved March 15, 2024, from https://opencredo.com/blogs/making-sense-of-data-with-rdf-vs-lpg/

[7] «ISO/IEC 39075:2024». ISO, https://www.iso.org/es/contents/data/standard/07/61/76120.html. Query June 9, 2024.

[8] «Download Neo4j Desktop». Graph Database & Analytics, https://neo4j.com/download/. Consulta 31 maig 2024.

[9] Vaswani, Ashish, Shazeer, Noam, Parmar, Niki, Uszkoreit, Jakob, Jones, Llion, Gomez, Aidan N., Kaiser, Łukasz, & Polosukhin, Illia. (2017). Attention is All you Need. Advances in Neural Information Processing Systems, 30. https://proceedings.neurips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[10] Beeching, Edward, Fourrier, Clémentine, Habib, Nathan, Han, Sheon, Lambert, Nathan, Rajani, Nazneen, Sanseviero, Omar, Tunstall, Lewis, & Wolf, Thomas. (2023). Open llm leaderboard. Hugging Face. https://huggingface.co/spaces/HuggingFaceH4/open_llm_leaderboard

[11] Li, Shiyang, Gao, Yifan, Jiang, Haoming, Yin, Qingyu, Li, Zheng, Yan, Xifeng, Zhang, Chao, & Yin, Bing. (2023). Graph reasoning for question answering with triplet retrieval. https://doi.org/10.48550/ARXIV.2305.18742

[12] Han, Jiuzhou, Collier, Nigel, Buntine, Wray, & Shareghi, Ehsan. (2024). Pive: Prompting with iterative verification improving graph-based generative capability of llms (arXiv:2305.12392). arXiv. http://arxiv.org/abs/2305.12392

[13] Wei, Jason, Wang, Xuezhi, Schuurmans, Dale, Bosma, Maarten, Ichter, Brian, Xia, Fei, Chi, Ed, Le, Quoc V., & Zhou, Denny. (2022). Chain-of-thought prompting elicits reasoning in large language models. Advances in Neural Information Processing Systems, 35, 24824-24837. https://proceedings.neurips.cc/paper_files/paper/2022/hash/9d5609613524ecf4f15af0f7b31abca4-Abstract-Conference.html

[14] Chen, Xiaojun, Jia, Shengbin, & Xiang, Yang. (2020). A review: Knowledge reasoning over knowledge graph. Expert Systems with Applications, 141, 112948. https://doi.org/10.1016/j.eswa.2019.112948

[15] Ventura (2024). Application of LLM-Augmented Knowledge Graphs for Wirearchy Management [Computer software]. Retrieved from https://github.com/xventura70/tfm-uoc-wirearchy