**EmPuAssistant: A semantic pipeline for EmPULIA documents using LLaMAntino-ANITA**

Martina Capone*1,\**

1 University of Bari Aldo Moro Bari, Apulia, Italy

Abstract

This paper introduces *EmpuAssistant*, a modular and reproducible pipeline for the semantic extraction and organization of procedural knowledge from EmPULIA, a public administration platform for procurement in Apulia, Italy. The system integrates PDF parsing, text reformulation via fine-tuned language models, and RDF triple extraction to construct a domain-specific knowledge graph. Through a Retrieval-Augmented Generation (RAG) framework, users can query procedural content using natural language. Evaluation on 52 manually curated questions demonstrates the effectiveness of the LLaMAntino-3-ANITA model in producing accurate and context-aware responses.

Keywords

LLaMAntino-3-ANITA, Knowledge Graph construction, RDF triple extraction, RAG

# Introduction and Motivations

Digital platforms in the public administration domain, such as EmPULIA, provide a wide range of procedural documents addressed to institutions and economic operators. However, these contents are typically published in PDF format, whose structure is often unsuitable for automated processing. In this context, a system has been developed for the semantic organization of procedural knowledge through the automated generation of a knowledge graph.

The proposed pipeline performs PDF scraping from the EmPULIA platform [1], followed by text conversion and reformulation using large language models (LLMs). This step aims to clean the extracted text by removing redundancies and structural artifacts introduced during PDF conversion. A controlled mechanism is then applied to extract RDF triples using the LLaMAntino-3-ANITA model, a fine-tuned LLM developed by the research group at the University of Bari [2]. The extracted triples are validated using regular expressions to ensure syntactic consistency and structural correctness.

In the second stage, the generated knowledge graph is integrated into a Retrieval-Augmented Generation (RAG) framework, enabling natural language queries to retrieve and rank the most relevant triples. To support user interaction, a simple **chat-style web** interface has been developed using Gradio, making the system easily accessible to end users.

# Related Work

The integration of Large Language Models (LLMs) with Knowledge Graphs (KGs) has recently emerged as a promising paradigm for structuring and accessing information extracted from unstructured documents. Several works have explored the automatic generation of KGs from heterogeneous sources, the use of RDF triples in Retrieval-Augmented Generation (RAG) architectures, and the application of domain-specific ontologies for semantic enrichment.

Docs2KG presents a unified pipeline for constructing knowledge graphs from various document types, including PDFs, by leveraging LLMs for content extraction and structuring. This approach is particularly relevant to the problem of transforming semi-structured procedural content into reusable, semantic knowledge bases [3]. Similarly, the DO-RAG framework demonstrates how domain-specific knowledge graphs can be combined with RAG models to support question answering from technical documentation, integrating both symbolic and neural components [4].

Several works focus on enriching RAG architectures with graph-based representations. For instance, OG-RAG proposes the use of ontologies to ground the retrieval process, improving the relevance and consistency of the context provided to the language model [5]. GraphRAG, in particular, explores how graphs built from document content can improve the grounding of generative responses and mitigate hallucination in LLM outputs [6].

Surveys such as *"Unifying LLMs and Knowledge Graphs: A Roadmap"* highlight current efforts to bridge parametric knowledge stored in LLMs with explicit symbolic representations in KGs [7]. These studies support the idea that hybrid systems—where LLMs are used both to construct and query semantic graphs—can provide more explainable and controllable outputs.

The approach proposed in this work is aligned with these research directions, but it is tailored to a specific and underexplored context: the extraction of procedural knowledge from public administration documents published on the EmPULIA platform. Unlike general-purpose methods, the proposed system combines PDF scraping, fine-tuned LLM processing using LLaMAntino-3-ANITA, controlled RDF triple extraction, and a custom scoring mechanism for semantic retrieval. Moreover, the integration of this pipeline into a lightweight chatbot interface using Gradio makes it suitable for direct interaction with end users.

# Proposed Approach

## Description of the solution and dataset

The proposed solution addresses the need for structured access to procedural documentation published on the **EmPULIA** platform, a public administration portal used for managing procurement procedures in the Apulia region (Italy). The target content consists of PDF manuals available under the following sections:

* *Guide pratiche nuova piattaforma*
* *Nuove Guide dedicate agli utenti SA*
* *Nuove Guide dedicate agli Operatori Economici*

These documents are characterized by formal procedural language, structured lists, and domain-specific terminology. They are primarily aimed at guiding users—both institutional actors and economic operators—through the functionalities and obligations of the EmPULIA platform.

As of today, **a total of 38 documents** have been collected and processed. These consist of publicly available PDFs published on the platform, which form the basis of the dataset used in this work.

To transform this semi-structured material into a format suitable for **semantic querying**, an automated pipeline was developed for the generation of a **domain-specific knowledge graph (KG)**. The pipeline is designed to be **modular and reusable**, allowing for minimal adjustments when switching to other sources of administrative documentation.

The process begins with the **scraping of PDFs** directly from the EmPULIA website. Each document is converted into plain text using PDF parsing techniques. However, raw conversion introduces a variety of issues such as duplicated content, broken formatting, and artifacts from index structures. To address this, each text block is passed through a **Large Language Model (LLM)**—specifically, **LLaMAntino-3-ANITA**, a fine-tuned model developed by the University of Bari—for **semantic reformulation**. This step ensures that the resulting text is concise, readable, and suitable for knowledge extraction.

After reformulation, the same LLM is prompted to **extract RDF triples** that capture factual or procedural knowledge present in the text. Since LLM-generated triples may vary in structure, a post-processing step using **regular expressions** is applied to validate the format and ensure compatibility with standard KG structures. This validation is critical, as the **accuracy of the triples directly impacts the quality and navigability of the resulting knowledge graph**.

The second phase of the solution is focused on **interactive querying**. When a user submits a natural language query—typically referring to specific use cases or platform functionalities—a retrieval mechanism processes the query by removing stopwords and identifying **triples whose content overlaps with the query terms**. To prioritize the most relevant information, a **custom scoring function** ranks the candidate triples, and the top-*k* entries are selected.

These selected triples, along with the user's query, are passed to a separate **Large Language Model**, specifically **Mistral**, which is used for **final answer generation**. This separation of concerns—using **LLaMAntino-3-ANITA** for document understanding and triple extraction, and **Mistral** for response synthesis—ensures optimal performance for both extraction accuracy and fluency in natural language interaction.

Finally, to make the system accessible to end-users, a lightweight **chatbot-style interface** was developed using **Gradio**. This interface allows users—particularly non-technical ones—to interact with the knowledge graph through natural language, receive contextualized responses, and navigate complex administrative documentation more intuitively.

## Main technical details

**Query Answering with Knowledge Graph Augmentation**

Once the knowledge graph has been constructed, it is employed as an external source of structured knowledge to augment responses generated by a local language model. This architecture enables a controlled, verifiable interaction framework, in which generated answers remain grounded in factual information extracted from institutional documents.

**Triple-Based Context Retrieval**

To handle user queries in natural language, the system first identifies the most semantically relevant triples from the knowledge graph. Each user query is tokenized and filtered using an Italian stopword list. Relevance scoring is performed by matching keywords against the subject, predicate, and object components of each triple. A weighted heuristic assigns higher scores to matches on subject and object nodes than on the relation label, reflecting their greater importance in semantic alignment.

The top-ranked triples are selected and used to build a contextual prompt that informs the language model of relevant facts. This lightweight retrieval mechanism provides a balance between precision and computational efficiency and ensures that the model is exposed only to contextually meaningful graph fragments.

**Response Generation**

The selected triples are passed to a locally hosted instruction-tuned model (Mistral-7B in GGUF format) through a structured prompt. The prompt includes the original user query and a flattened representation of the retrieved triples. The model is instructed to answer in Italian using clear and accessible language, suitable for non-technical users. This is particularly important given the public-facing nature of the documentation and the assistant's intended audience.

By grounding the model’s response on curated graph information, the system significantly reduces the risk of hallucinated content. Furthermore, since the retrieved triples originate from a manually verified knowledge graph, the assistant provides answers with higher factual reliability and traceability.

**Deployment and User Interface**

The final system is deployed as a web-based chat interface using Gradio. A custom dark-themed UI presents the assistant as a conversational agent capable of handling open-ended queries about procedures, access methods, and participation criteria on the EmPULIA platform. The interface supports rapid experimentation, direct user interaction, and qualitative evaluation by domain experts.

1. **Language model**:

* swap-uniba/LLaMAntino-3-ANITA-8B-Inst-DPO-ITA\_GGUF
* Loaded with llama-cpp
* Local, quantized (Q4\_K\_M), context window: 2048 tokens

1.  **Rephrasing**:

* Each .txt file is split into chunks of ~2000 tokens
* A knowledge base (compiled from EmPULIA documentation) is injected into the prompt for contextual rephrasing
* Prompt format includes instructions + KB + chunk

1.  **Tokenizer**: tiktoken with the cl100k\_base encoding for compatibility with chunking limits
2.  **Graph structure**:

* Triples are stored in a MultiDiGraph
* Edges are labeled with relations
* Final output: knowledge\_graph/kg.graphml

1.  **Triple extraction**:

* Early tests with ReBEL and mReBEL were discarded
* The rephrased text proved more suitable for simpler extraction rules

## Other information useful to replicate the approach

The full pipeline is implemented in Python using the following libraries: llama-cpp-python, tiktoken, networkx, PyMuPDF, nltk, Gradio, and langchain. A centralized Config class manages all parameters and paths across the pipeline. Each stage of the pipeline is modular and checks for pre-existing outputs to avoid redundant computation.

Directory structure:

* data/pdf: downloaded PDF manuals
* data/text: raw text extracted from PDFs
* data/rephrased\_text: LLM-rewritten text chunks
* data/triples: extracted RDF triples
* knowledge\_graph/kg.graphml: final knowledge graph

Tokenization is performed using tiktoken with the cl100k\_base encoding to ensure chunk compatibility with the model's context window. Each .txt file is split into ~2000-token chunks and rephrased individually. The model used is swap-uniba/LLaMAntino-3-ANITA-8B, loaded via llama-cpp in quantized GGUF format (Q4\_K\_M), with a maximum context window of 2048 tokens.

Graph construction leverages networkx.MultiDiGraph, where nodes represent entities or concepts and labeled directed edges encode semantic relations. Each edge is assigned a unique ID. The resulting graph is exported in GraphML format for interoperability.

# Evaluation

To assess the effectiveness of the proposed pipeline in generating accurate responses grounded in EmPULIA’s official documentation, a benchmark set of **52 domain-specific questions** was manually created. These questions were derived from PDF manuals published on the EmPULIA platform and were designed to reflect realistic scenarios involving procurement procedures, platform functionalities, document submission, and evaluation processes.

## Experimental Setup

Each question was processed using a retrieval-augmented generation (RAG) approach. Relevant triples were extracted from the knowledge graph and embedded within a structured prompt, which was then passed to the language model. Two models were compared under identical inference conditions:

* **LLaMAntino-ANITA**: an 8B parameter instruction-tuned model for the Italian language, optimized for clarity, politeness, and factual responses.
* **Mistral**: a general-purpose multilingual instruction-following model, not specialized for public administration content or the Italian domain.

Both models were evaluated with the same context length, temperature, and formatting constraints to ensure consistency.

## Evaluation Methodology

Given the domain-specific nature of the content, a fully authoritative evaluation of factual accuracy would require input from subject-matter experts (SMEs). In the absence of such an annotation layer, a **qualitative analysis** of the generated responses was conducted, using the following criteria:

* **Factual consistency** with the content in the extracted triples and source documents.
* **Completeness** in addressing the key aspects of the input question.
* **Fluency and formality** of the generated text, particularly in an institutional context.
* **Presence of hallucinations**, such as fabricated acronyms, commands, or roles.
* **Language appropriateness**, focusing on adherence to Italian.

## Observed Behaviors

The LLaMAntino-ANITA model consistently delivered coherent and well-structured responses. Notable characteristics include:

* Clear and often **summarized answers**, providing both overview and detail.
* A **polite and institutionally appropriate tone**, aligning with the expected user-facing role.
* Inclusion of a **disclaimer** in highly specific cases, stating that the answer is generated based on the knowledge graph and may not reflect official or complete information—an effective measure for encouraging user verification.
* Absence of hallucinations or fabricated terminology.

Conversely, the Mistral model exhibited several shortcomings:

* Frequent **repetition of query keywords** at the beginning of responses, resulting in unnatural phrasing.
* Tendency to **invent acronyms or procedural elements** not grounded in the documentation.
* In some instances, answers were **partially or fully in English**, despite the Italian context.
* Overall, the content was less structured and more prone to verbosity without informational gain.

## Qualitative Summary

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Model | Factual Accuracy | Completeness | Fluency & Tone | Hallucinations | Language Adherence |
| LLaMAntino-ANITA | High | High | Clear, Formal | None observed | Fully in Italian |
| Mistral | Moderate | Partial | Inconsistent | Frequent | Occasionally English |

# Conclusion and limitations

This work demonstrates the effectiveness of integrating a locally hosted, instruction-tuned LLM into a structured pipeline for knowledge extraction from procedural documents. While ReBEL, mReBEL, and spaCy were considered, **LLaMAntino-3-ANITA-8B** proved superior in this use case due to its Italian language specialization and context-aware generation.

**Limitations**:

* Incomplete coverage of edge cases with fuzzy terminology
* Manual curation still required for knowledge base construction
* No evaluation of fact consistency across the graph and model generations

**Future directions**:

* Using semantic retrievers (e.g., FAISS) for graph search
* Integrating automatic KB updating
* Building a front-end interface for virtual assistant interaction

References

[1] “EmPULIA - Guide pratiche.” Accessed: Jun. 30, 2025. [Online]. Available: http://www.empulia.it/tno-a/empulia/Empulia/SitePages/Guide%20pratiche.aspx

[2] M. Polignano, A. Moro Bari, P. Basile, and G. Semeraro, “Advanced Natural-based interaction for the ITAlian language: LLaMAntino-3-ANITA,” May 2024, Accessed: Jun. 30, 2025. [Online]. Available: https://arxiv.org/pdf/2405.07101

[3] Q. Sun *et al.*, “Docs2KG: Unified Knowledge Graph Construction from Heterogeneous Documents Assisted by Large Language Models,” *Proceedings of  (Under Review)*, vol. 1, Jun. 2024, doi: XXXXXXX.XXXXXXX.

[4] D. O. Opoku, M. Sheng, and Y. Zhang, “DO-RAG: A Domain-Specific QA Framework Using Knowledge Graph-Enhanced Retrieval-Augmented Generation,” May 2025, Accessed: Jun. 30, 2025. [Online]. Available: https://arxiv.org/pdf/2505.17058

[5] K. Sharma, P. Kumar, and Y. Li, “OG-RAG: Ontology-Grounded Retrieval-Augmented Generation For Large Language Models,” Dec. 2024, Accessed: Jun. 30, 2025. [Online]. Available: https://arxiv.org/pdf/2412.15235

[6] H. Han *et al.*, “Retrieval-Augmented Generation with Graphs (GraphRAG),” Dec. 2024, Accessed: Jun. 30, 2025. [Online]. Available: https://arxiv.org/pdf/2501.00309

[7] J. Z. Pan *et al.*, “Large Language Models and Knowledge Graphs: Opportunities and Challenges,” vol. 000, no. 42, p. 30, Aug. 2023, doi: 10.1234/0000000.00000000.

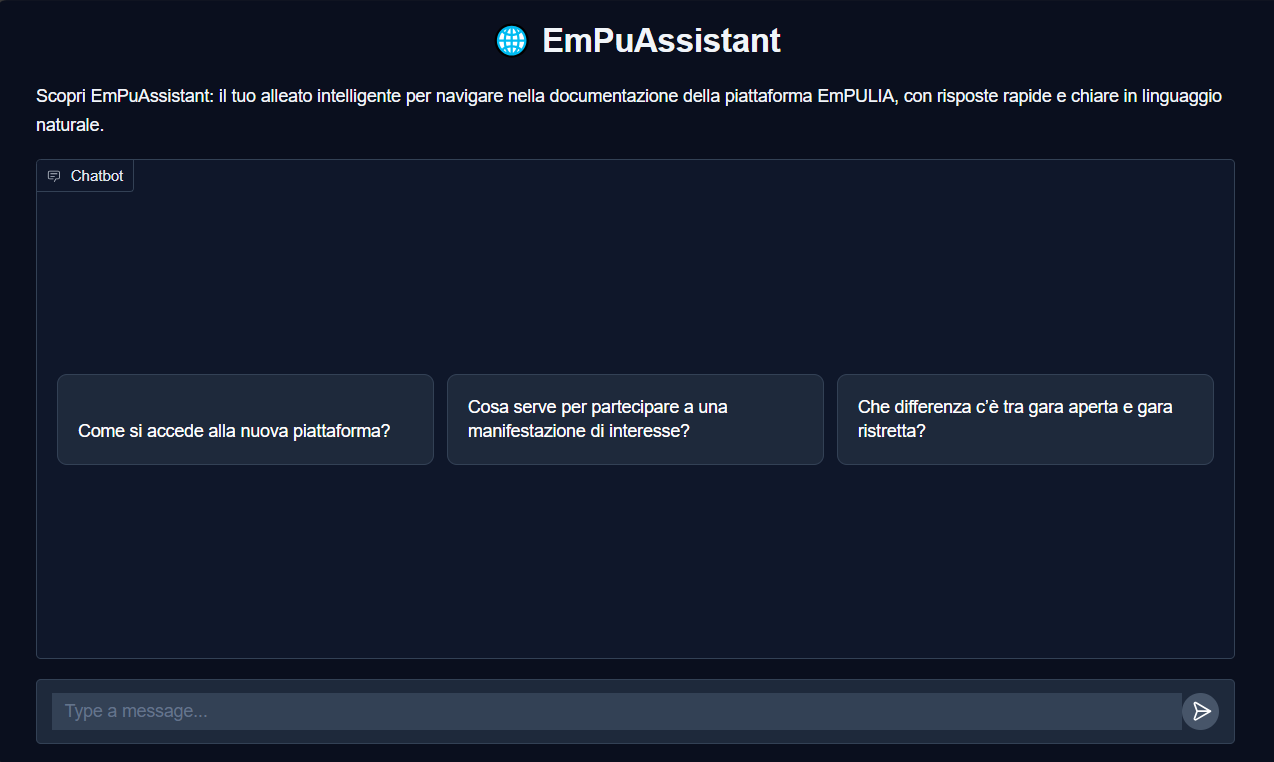


Immagine che contiene testo, schermata, software, Sito Web

Il contenuto generato dall'IA potrebbe non essere corretto.

Immagine che contiene testo, schermata, software, Sistema operativo

Il contenuto generato dall'IA potrebbe non essere corretto.