MARTIALIS: An Open Framework for Knowledge Graphs-based Retrieval Augmented Generation

Edoardo Bianchini[1,2], Filippo Bianchini[1], Marco Calamo[1], Francesca De Luzi[1], Mattia Macrì[1,*] and Massimo Mecella[1]

[1]Sapienza Università di Roma, NESMOS and DIAG, Italy
[2]Université Grenoble Alpes, Autonomie, Gérontologie, E-santé, Imagerie et Société - AGEIS, France

Abstract

We present Martialis[1], a framework designed to enhance the performance of Large Language Models (LLMs) in domains-specific tasks (e.g. medical, legal) by implementing an ontology-based data representation approach. We designed the framework to be almost transparent to the end user: once set up with domain-relevant documents and an ontology, it enables both complex reasoning and domain-specific text generation. That is made possible by our novel information extraction pipeline that improves existing Retrieval Augmented Generation (RAG) techniques with a Domain Specific Knowlegde Graph – inferred from the documents – and a sanity check on the output – inferred from the ontology. This dual-layered approach ensures accuracy and relevance, addressing common limitations in existing solutions. The Martialis framework has been rigorously evaluated through collaboration with domain experts, comparing its performance against similar state-of-the-art systems. Results indicate improvements across key metrics, and speed-up in the efficiency of executing user tasks.

Keywords

Large Language Models, Knowledge Graphs, Ontology-base Data Representation, Retrieval Augemented Generation

1. Introduction

The development of Natural Language Processing (NLP) systems has rapidly advanced in recent years with the rise of Large Language Models (LLMs), specifically generative ones, which have demonstrated great capabilities in understanding and generating human-like text. These advancements have opened new possibilities for addressing a variety of linguistic tasks. However, despite their potential, generative LLMs face significant challenges, including aligning outputs with user intent and mitigating hallucinations [1, 2]. Such limitations can be attributed to structural aspects of LLMs: *(i)* the absence of deep reasoning capabilities and *(ii)* the lack of inherent domain-specific knowledge. As the task complexity grows, these limitations become more impacting. Hence, LLMs necessitate the introduction of complementary methodologies to address them. A promising approach to addressing these structural limitations lies in the adoption of hybrid AI methodologies, a paradigm emphasizing the integration of sub-symbolic AI techniques [3], such as LLMs, with symbolic AI ones [4], including Knowledge Graphs (KGs). The two approaches are somewhat complementary: sub-symbolic methods are characterized by their capacity, adaptability, and reliance on large datasets, while symbolic methods are transparent, precise, and data-efficient. Furthermore, while symbolic approaches often require the manual encoding of knowledge by human experts, sub-symbolic techniques rely on automatic learning from data, thereby reducing the need for manual intervention. By combining the adaptive and data-driven nature of sub-symbolic AI with the interpretability and logical rigor of symbolic AI, hybrid AI offers a synergistic framework that addresses the deficiencies of each approach when used in isolation [3, 5].

[1]Martialis, the well-known ancient roman poet appreciated for his writing versatility.

SEBD 2025: 33rd Symposium on Advanced Database Systems, June 16-19, 2025, Ischia, Italy

*Corresponding author.
edoardo.bianchini@univ-grenoble-alpes.fr (E. Bianchini); lastname@diag.uniroma1.it (F. Bianchini); lastname@diag.uniroma1.it (M. Calamo); lastname@diag.uniroma1.it (F. D. Luzi); lastname@diag.uniroma1.it (M. Macrì); lastname@diag.uniroma1.it (M. Mecella)
0000-0002-9730-8882 (E. Bianchini); 0009-0006-3278-9853 (F. Bianchini); 0009-0006-2602-9604 (M. Calamo); 0000-0002-9896-2528 (F. D. Luzi); 0009-0004-7679-6153 (M. Macrì); 0000-0002-9730-8882 (M. Mecella)

The introduction of *ontology-based knowledge representation* could pave the way for the development of Intelligent Information Systems (IIS) [6].

In this context, although several preliminary studies [7, 8, 9] explore the potential benefits and challenges of hybrid AI methodologies, few offer practical, end-to-end, and repeatable solutions for developing Intelligent Information Systems in real-world scenarios. To address this gap, we focus on the challenge of building end-to-end and repeatable IIS capable of performing linguistic tasks that require domain-specific knowledge and reasoning abilities, which we define as complex linguistic tasks. In our previous works [10, 11], we presented the concept of an enhanced Retrieval Augmented Generation (RAG) pipeline. This pipeline leverages KGs to supply LLMs with relevant information and rules, effectively compensating for their lack of domain-specific knowledge and reasoning capabilities in complex linguistic tasks. In this article, we present MARTIALIS[1], our open-source and easily repeatable framework to realize such adaptable pipeline. MARTIALIS is an open framework designed to provide users with accurate, precise answers or comprehensive, detailed text generation within domain-specific contexts (e.g., legal, healthcare).

To evaluate our framework, we collaborated with a team of medical experts from a hospital in Rome to benchmark the performance in the healthcare domain. Specifically, we assessed its ability to answer medical questions and generate various domain-specific documents derived from Electronic Clinical Records (ECRs).

The paper is structured as follows: in Section 2, we review the existing literature on the automatic or semi-automatic construction of KGs from unstructured documents, as well as frameworks implementing RAG pipelines and ontology-based compliance processes. In Section 3, we present the architecture of MARTIALIS, along with a case study in the healthcare domain. Section 4 details the dataset composition and the evaluation process used to assess our framework. Lastly, in Section 5, we discuss the limitations and challenges of our framework and propose future directions for improvement.

2. Related Work

In this section, we review existing works that directly relate to the core features and theoretical foundation of Martialis: *(i)* Text2KG, for extracting structured knowledge from unstructured text to create KGs in an automatic or semiautomatic way; *(ii)* KG RAG, namely methods that combine LLMs with KGs for context-aware text generation; *(iii)* and Ontology-Augmented Text Generation approaches that enhance generation by integrating ontologies for validation and consistency. Our objective is to analyze existing frameworks that claim practical, implementable outcomes, emphasizing contributions with concrete applications or open-source solutions. These works have been assessed by evaluating six key criteria: code availability (open source), innovative methodology, adaptability (to multiple domains), correctness, repeatability, and support the generation (of complex and detailed responses from KG).

Text to KG. The task of transforming unstructured text into a Knowledge Graph has always been done by experts in the domain due to the high level of abstraction required to comprehend a text and structure it into nodes and relationships. To automate this task several attempts have been made by using various machine learning techniques [12]. According to the latest survey [13] the results are encouraging, since many frameworks have been created for extracting KG from unstructured text using LLMs with little or –in some special cases– no human expertise, [14, 15, 16, 17].

RAG with Knowledge Graphs. Hallucination is the most common and impactful limitation of LLMs [1, 2]. RAG addresses these shortcomings by combining the strengths of LLMs with external knowledge bases [18]. RAG enables language models to access and process information from an external source (e.g. documents, database) in real time, allowing them to generate more accurate, informative, and up-to-date responses [19]. While several techniques for traditional RAG have been proposed and tested, there is still room for innovative approaches, such as incorporating KGs as external sources. Among the

[1] MARTIALIS' code is available at: DIAG-Sapienza-BPM-Smart-Spaces/Martialis

most relevant, GLens [20] from Apple researchers and KRAGEN [21] are frameworks that integrate KGs into RAG to efficiently retrieve and structure relevant information for precise query resolution. Some other works examine RAG and KG primarily from a theoretical and visionary standpoint, lacking the introduction of innovative approaches: in [22] a framework that combines web search and Knowledge Graphs that is called WeKnow-RAG is proposed, introducing a domain-specific RAG system with a multi-stage information retrieval logic and an LLM self-assessment system.

Ontology Augmented Text Generation. Ontologies have always been the pillar for structuring data and enabling reasoning. The advent of LLM has drawn increasing interest in combining language generation capabilities with the structured knowledge represented in ontologies [16, 23]. In particular, we are interested in using LLMs for validating the structure of natural text against an ontology. We envisioned a scoring mechanism on a Resource Description Framework KG automatically extracted by the LLM from natural text checked against the ontology, in a way inspired by Shapes Constraint Language (SHACL) [2]. To the best of our knowledge, it does not exist in literature a similar mechanism that is fully implemented in a complete text generation framework. The work by [24] proposes a similar way of validating semantic artifacts using ontology and LLMs, however they did not run tests on natural text. Finally, the authors of [25] present a vision of a roadmap to improve the accuracy of LLMs with ontologies for sanity checks.

3. Martialis' Architecture

MARTIALIS is a next-generation, open-source IIS designed to perform complex linguistic tasks in a specific domain of knowledge, addressing a significant gap in the current literature: directly supports advanced text generation MARTIALIS *(i)* answers complex questions regarding the selected domain even if they require reasoning steps and *(ii)* generates structured text for that domain.

The complete architecture of the framework is presented in Figure 1. The design is modular, comprising the following key components, each of which can be individually replaced with customized implementations if necessary: *Automatic KG-Extractor*, *Advanced Retrieval-Augmented-Generation*, and *Ontology-Based Validator*. The implementation leverages the Python libraries Llamaindex[3] and Langchain[4], with GPT-4o [26] serving as the primary LLM.

In the remainder of this section, we will detail the required inputs, elucidate the operational logic of MARTIALIS using the healthcare domain as an illustrative example (also employed for validation), and provide an in-depth analysis of MARTIALIS' modules.

3.1. Martialis' Input

After identifying the target knowledge domain, it is necessary to indicate to MARTIALIS, through a configuration file, the source folder of the available documents. If the user wants to generate some advanced domain-specific artifact, they have also to provide an ontology that describes that artifact.

Domain Documents The documents inherent to the specified domain are the main pillar for MARTIALIS. We have no constraints on the data format, as long as it contains meaningful text. In the healthcare domain, we used several plain text and .PDF files that describe patient hospitalizations. The documents are parsed and processed into a Vector Store [27], ready to be retrieved by our system.

Ontology To support advanced text generation, MARTIALIS needs a blueprint to follow. This blueprint is provided as an ontology for the type of text that the user wants to create. In our example domain, we produced –guided by domain experts– one ontology for both clinical history and discharge letter

[2] https://www.w3.org/TR/shacl/
[3] https://www.llamaindex.ai/
[4] https://www.langchain.com/

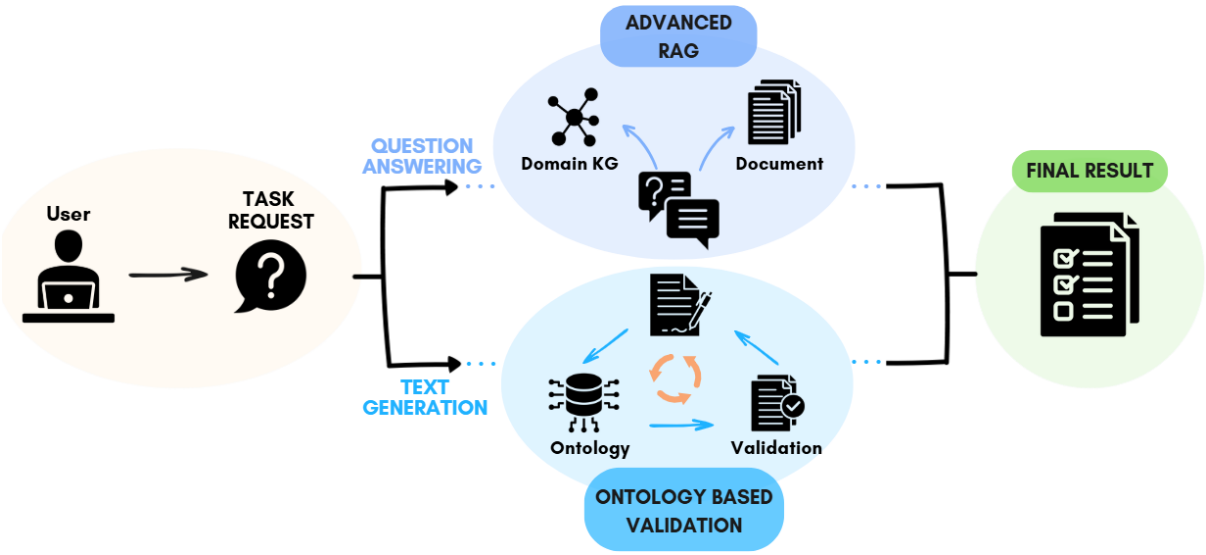

Figure 1: MARTIALIS ' architecture

Figure 2: MARTIALIS preliminary Step

generation. We support both OWL and SHACL ontologies, as long as they model all pieces of information about the abstractions that have to be present in the text (e.g. patient class, healthcare provider class). There is also the option to automatically generate the ontology using tools like [16], but we have obtained the best result with a human-crafted one.

3.2. Preliminary Step pipeline

As illustrated in Figure 2, when the user adds domain-specific documents or an ontology to the MARTIALIS folder—either for the first time or incrementally—the framework automatically initiates a preliminary processing step consisting of the following actions: *(i)* the documents are parsed and stored in a Vector Database; *(ii)* a domain-specific Knowledge Graph is automatically generated from the documents using the Automatic KG-Extractor module. This KG can then be queried to retrieve precise and relevant information required for task completion; *(iii)* the ontology, which serves both as a foundation for extracting the entities necessary for text generation tasks and as a reference for correctness validation in the final output, is parsed and prepared for use within MARTIALIS.

KG-Automatic Extractor Module The Domain KG must be an instance-level Knowledge Graph. Consequently, unlike ontologies or schema-level knowledge, it is typically impractical to construct it manually, even with the assistance of domain experts. Given that the objective of this work is to make MARTIALIS' real-world application feasible, we deemed it essential to address the need for automatically building its own Domain KG. Since the majority of knowledge and information about organizations remains stored in unstructured text documents, MARTIALIS automatically constructs the Domain KG from the unstructured documents provided. To implement the domain KG extraction based on [14] prompts, we decided to use the LLamaIndex Python library. Below, we provide real-world examples of Knowledge Graphs generated using our KG-Automatic Extractor module. Figure 3 showcases the

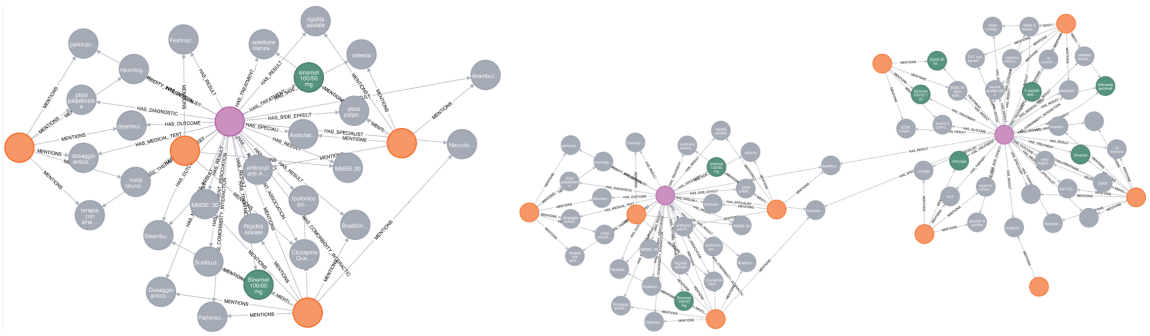

Figure 3: MARTIALIS result from a single document of Domain KG (left); result from a multiple documents of Domain KG (right)

extraction results from a single clinical record, visualized in Neo4j. On the left, the Knowledge Graph represents the extraction from a single document, while on the right, the updated graph incorporates an additional document. This incremental approach enables seamless integration of new documents while retaining all previously extracted information, ensuring no data is lost during the process.

3.3. Question answering and Text Generation pipeline

MARTIALIS can understand if the user is asking a question about the domain or wants to generate some advanced text. It routes the user prompt to the *Advanced-Retrieval-Augmented-Generation* module or the *Ontology-Based-Validator* module that we will present below in detail.

Advanced-Retrieval-Augmented-Generation Module. It is the module that support advanced question answering. It is capable of producing the answer relies on a chain of thoughts [28] that includes information extracted from the documents, structured with the entities and relationships of the Domain KG, using text2cypher prompt-engineering.

The Domain KG enables the LLM to follow a more strict path in generating the final output, effectively simulating an improvement in the LLM's reasoning capabilities. Its function is carried out in the augmented generation phase of the RAG pipeline and is iterative. In the standard RAG procedure, once the enriched prompt is created, it is given to the LLM as input for generating a response. However, the accuracy or relevance of the response is not verified using external tools. For complex language tasks, producing the desired response may not be straightforward, which is why the Domain KG is used to add a reasoning and abstraction layer. During the augmented generation phase, the specialized LLM generates a preliminary output based on the enriched prompt. This output is bounded by the logic of the Domain KG. An example from the healthcare domain would be:

QA Task

Q: *"Which medications were prescribed to the patient PATIENT_008? Provide a comprehensive list"*

A: Sinemet 100/25 (1 tablet 3 times daily: 7:30 AM, 12:30 PM, 5:30 PM), Zoloft 50 mg (1 tablet in the evening).

Ontology-Based-Validator Module. This module is capable of generating text artifacts that require deep comprehension of a specific knowledge domain. The final text output is validated against the provided ontology. When this kind of request is detected, MARTIALIS uses prompt engineering to retrieve from the ontology the relevant class and properties for the specific task (e.g. MEDICAL_CONDITION, TREATMENT_OUTCOME, PATIENT_INFORMATION). Those extracted objects are integrated into a chain of prompts to generate the requested text. The text generated in this way is then passed to the validation step: it is converted into a Resource Description Framework KG using the KG-Automatic Extractor

module and checked against the ontology. A score from 0 to 1 based on compliance with the ontology is computed by checking the percentage of constraints on the class and the attributes that are present or missing. In this way, MARTIALIS can show the user a numeric value for confidence in the generated output.

$$score = \frac{\sum_{i=1}^{n} p_i(D) + \sum_{j=1}^{m} c_j(D)}{n + m} \tag{1}$$

Where: D is the generated text, $p_i(D)$ is 1 if property p_i is present in D, and 0 otherwise. $c_j(D)$ is 1 if class c_j is present in D, and 0 otherwise. n is the total number of properties. m is the total number of classes. An example we generated with score 0.63 is:

Text Generation Task

Q: *"I need a detailed summary of the clinical history for the patient.* PATIENT_008*"*

Class Extracted: PATIENT_INFORMATION, MEDICAL_CONDITION, THERAPY, TREATMENT_OUTCOME [...]

A: The patient, PATIENT_008, female, born on May 26, 1952, is being treated at the Neurology Unit of Hospital. She initially presented with tremors and was prescribed Artane, which she later reduced due to internal tingling and nausea. Subsequently, she was switched to Sinemet, which led to a significant improvement in her tremors [...].

4. Evaluation

To validate our framework, we employed two types of evaluations, automatic and human, tailored to the structure and objectives of our approach. For tasks focused on answering specific questions, we utilized a single automatic evaluation approach. This involved assessing both the reliability of the extracted context and the consistency of the generated responses about the posed questions. For the text generation tasks, automatic validation was conducted by comparing the knowledge graph derived from the final answer against the ontology provided as the initial context for the task. Additionally, for the text generation task, we performed a human evaluation. This involved developing and administering detailed questionnaires to medical experts to assess the quality and domain-specific relevance of the generated outputs.

Dataset. The dataset used for our case study and testing consists of 32 clinical records[5], which were collected and selected by an expert medical team from a hospital in Rome. This selection was carried out following approval from the hospital's ethical committee to process the data strictly for research purposes. The clinical records have been fully anonymized through PII masking with Named Entity Recognition (NER) implemented through Llama Index. To maintain the integrity of the information, the clinical records were left unprocessed in their original language, Italian. As a result, any information produced by the model is also in Italian.

Automatic Evaluation for QA task. We designed a test involving eight questions of increasing complexity. These questions required the extraction and re-elaboration of various types of medical information to produce accurate answers. Each of the eight questions was applied to all test documents available in our dataset. In parallel, we collaborated with the team of domain experts to extract the correct answers for each of the eight questions from the clinical records. These expert-verified answers

[5]Datset for QA, clinical record, and all evaluation results are open and accessible at the following link: Healthcare_Dataset_Martialis

served as the ground truth, allowing us to directly compare the outputs of our framework with the expected responses. Finally, we selected and calculated four automatic metrics using the DeepEval framework[6]. Selected metrics are the following ones: *(i) Contextual Precision.* It evaluates the quality of a ranked list of retrieved nodes by considering both their relevance and their position in the ranking. The metric is computed using the total number of relevant nodes in the retrieval context and the relevance of each node; *(ii) Contextual Recall.* This metric assesses the quality of the RAG pipeline's retriever by measuring the extent to which the retrieved context aligns with the expected output; *(iii) Faithfulness.* It evaluates whether the model generates factually correct information by comparing the actual output of the LLM to the provided context; *(iv) Answer Relevancy.* This metric measures the quality of the RAG pipeline's generator by evaluating how relevant the actual output of the LLM is compared to the provided input.

The results of this automatic evaluation were aggregated using two complementary approaches: *(i) Overall Performance Analysis*; *(ii) Performance by Question Complexity Analysis.* The *overall performance analysis* shows strong results across the metrics. The Contextual Precision (80.36%) and Contextual Recall (83.07%) demonstrate the model's ability to effectively understand and retrieve relevant contextual information. The Faithfulness score (80.64%) indicates that the model's responses are fairly faithful to the original content. The Answer Relevancy score (88.84%) suggests that the answers are highly relevant to the questions asked.

As for *question complexity analysis*, Contextual Precision shows the highest degree of variability with relevant low values for question 2 and question 5. Notably, on most complex questions (e.g. questions 6, 7, and 8) it seems to be very precise in exacting context reaching values above 90%. *Contextual Recall* shows a similar situation: variability is high, we can note that as for Contextual Precision, we have the lowest values for questions 2 and 5. Again, greater values are distributed in the last three questions exceeding the threshold of 90%. Furthermore, *Faithfulness* shows a different distribution: variability is lower, and the smallest values are concentrated in questions 1 and 7. Question 8 shows still a great result. Finally, *Answer Relevancy* shows the best results and the lowest variability as it is the only metric having all values above the threshold of 75%. In particular, for question 1 we reached 100% of answer relevancy.

Automatic Evaluation for Text Generation. We evaluated the model's outputs in two distinct scenarios: *(i)* generating the patient's clinical history, and *(ii)* drafting the patient's discharge letter. In both cases, the initial stage of response generation involves extracting key entities and relationships from the ontology preliminarily extracted. Once the text-based response is generated, a knowledge graph is constructed from the output using the KG-Automatic Extractor Module. This module identifies the entities and relationships present in the generated text and maps them into a structured knowledge graph format. The resulting graph is then compared against the existing ontology to assess its alignment with the ground truth.

The evaluation metric used was the score of matching elements (both entities and relationships) between the extracted knowledge graph and the reference ontology presented in Section 3. The results, reflecting the model's ability to preserve the semantic integrity of the data during text generation, shows low values but this outcome align with expectations. This evaluation is not absolute; rather, it serves as a trigger to initiate the response revision process

Human Evaluation for Text Generation. To carry out and report the human evaluation of the text generation by MARTIALIS, we followed the guidelines in [29]. First, we determined the objective: to collect human feedback on the quality of the generated output. We intended this evaluation as an intrinsically quantitative evaluation. Hence, we made subjective opinions about a document quantifiable. We selected questions emphasizing the quality of the documents and transformed them into evaluation criteria. As a rating scale, we opted for a 4-point Likert scale, widely used in NLG assessments, and provided an interpretation of the scale before the questions. We recruited 18 medical domain experts,

[6] DeepEval is an open-source evaluation framework for LLMs

including doctors, medical trainees, medical PhD students, and nurses.

Generation Approaches Used. We compared the quality of draft documents written by a domain expert with those generated by an LLM using different approaches. In particular, we used the MARTIALIS' generation approach, a prompt guided by a domain ontology, and standard zero/few-shot prompts. For each approach, the LLM used is GPT-4o [26]. Using these four different document origins allowed us *(i)* to evaluate the potential future benefits of LLMs as draft document writers by comparing the quality of human-written and LLM-written documents, and *(ii)* to analyze the efficacy and efficiency of MARTIALIS' approach by comparing it with standard approaches *Questionnaire Structure.* We generated 24 documents, 6 for each of the four generation approaches, divided into 12 clinical histories and 12 discharge letters. This allowed us to evaluate a more heterogeneous set of sentences while requiring each participant to evaluate only a subset of the total sentences. Subsequently, the results from all forms were combined and analyzed as a single dataset. For any document, participants rated seven text features (Grammar, Medical Lexicon, Structural coherence, Logical Coherence, Ambiguity's Absence, Reliability, Human-like Writing). Overall, the results of the Human Evaluation of text generation, shown that the highest-rated documents are the human-written ones (2.88), followed by those generated with few-shot prompts, MARTIALIS-generated documents, and those generated with zero-shot (2.72). The results fall in a restricted range but show a clear separation between the performance of human-written and few-shot-generated documents, the MARTIALIS-generated documents, and zero-shot-generated ones. Similar observations emerge when looking into single text features; human-written documents outperform in four distinct features, few-shot approach excelled in two features, MARTIALIS' ones excel in one, while zero-shot approach never led in any features. Furthermore, focusing only on the LLMs-generated documents, few-shot and MARTIALIS approaches outperformances increase up to four and three while zero-shot approach stays at zero.

5. Discussion & concluding remarks

As far as the automatic evaluation, results are solid; the insights suggest that LLMs, if correctly guided, might be truly useful in drafting semi-structured documents like those in our case study. This assumption is reinforced considering the human evaluation, where there is a clear gap in documents quality between the ones generated without guidance (zero-shot prompt) and the ones generated with guidance (few-shot prompt and MARTIALIS). Furthermore, since the performances of MARTIALIS and few-shot are complementary, it seems interesting to explore MARTIALIS' RAG on KGs approach in synergy with classical few-shot prompting techniques.

We argue that MARTIALIS has the potential to stand out for completing complex linguistic tasks. We intend to continue working in this direction by *(i)* releasing evolutionary updates working on each module and *(ii)* investigating, and eventually fine-tuning, open LLMs to replace GPT-4o, ensuring accessibility and repeatability.

References

[1] J.-Y. Yao, K.-P. Ning, Z.-H. Liu, M.-N. Ning, L. Yuan, LLM Lies: Hallucinations are not Bugs, but Features as Adversarial Examples, 2023. arXiv:2310.01469.

[2] Z. Xu, S. Jain, M. Kankanhalli, Hallucination is inevitable: An innate limitation of large language models, arXiv preprint arXiv:2401.11817 (2024).

[3] E. Ilkou, M. Koutraki, Symbolic vs sub-symbolic ai methods: Friends or enemies?, in: CIKM (Workshops), volume 2699, 2020.

[4] M. Flasiński, M. Flasiński, Symbolic artificial intelligence, Introduction to Artificial Intelligence (2016) 15–22.

[5] R. Calegari, G. Ciatto, A. Omicini, On the integration of symbolic and sub-symbolic techniques for xai: A survey, Intelligenza Artificiale 14 (2020) 7–32.

[6] W. Liu, Y. Li, B. Huang, Evaluation of intelligent information system, in: 2022 IEEE 22nd International Conference on Software Quality, Reliability, and Security Companion (QRS-C), IEEE, 2022, pp. 183–188.

[7] F. Shi, F. Zhou, H. Liu, L. Chen, H. Ning, Survey and tutorial on hybrid human-artificial intelligence, Tsinghua Science and Technology 28 (2022) 486–499.

[8] A. Correia, A. Grover, D. Schneider, A. P. Pimentel, R. Chaves, M. A. De Almeida, B. Fonseca, Designing for hybrid intelligence: A taxonomy and survey of crowd-machine interaction, Applied Sciences 13 (2023) 2198.

[9] J. Wang, T. Lu, L. Li, D. Huang, Enhancing personalized search with ai: A hybrid approach integrating deep learning and cloud computing, International Journal of Innovative Research in Computer Science & Technology 12 (2024) 127–138.

[10] F. Bianchini, M. Calamo, F. De Luzi, M. Macrì, M. Mecella, Enhancing Complex Linguistic Tasks Resolution through Fine-tuning LLMs, RAG and Knowledge Graphs, in: International Conference on Advanced Information Systems Engineering, Springer, 2024.

[11] F. Bianchini, M. Calamo, F. De Luzi, M. Macrì, M. Mecella, A service-based pipeline for complex linguistic tasks adopting llms and knowledge graphs, in: M. Aiello, J. Barzen, S. Dustdar, F. Leymann (Eds.), Service-Oriented Computing, Springer Nature Switzerland, Cham, 2025, pp. 145–161.

[12] Z. Zhao, X. Luo, M. Chen, L. Ma, A survey of knowledge graph construction using machine learning., CMES-Computer Modeling in Engineering & Sciences 139 (2024).

[13] Y. Zhu, X. Wang, J. Chen, S. Qiao, Y. Ou, Y. Yao, S. Deng, H. Chen, N. Zhang, Llms for knowledge graph construction and reasoning: Recent capabilities and future opportunities, World Wide Web 27 (2024) 58.

[14] Y. Lairgi, L. Moncla, R. Cazabet, K. Benabdeslem, P. Cléau, itext2kg: Incremental knowledge graphs construction using large language models, arXiv preprint arXiv:2409.03284 (2024).

[15] B. Zhang, H. Soh, Extract, define, canonicalize: An llm-based framework for knowledge graph construction, arXiv preprint arXiv:2404.03868 (2024).

[16] V. K. Kommineni, B. König-Ries, S. Samuel, From human experts to machines: An llm supported approach to ontology and knowledge graph construction, arXiv preprint arXiv:2403.08345 (2024).

[17] Q. Sun, Y. Luo, W. Zhang, S. Li, J. Li, K. Niu, X. Kong, W. Liu, Docs2kg: Unified knowledge graph construction from heterogeneous documents assisted by large language models, arXiv preprint arXiv:2406.02962 (2024).

[18] P. Lewis, E. Perez, Retrieval-augmented generation for knowledge-intensive nlp tasks, Advances in Neural Information Processing Systems 33 (2020) 9459–9474.

[19] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, Q. G. ., Retrieval-augmented generation for large language models: A survey, 2024. arXiv:2312.10997.

[20] S. Zheng, H. Bai, Y. Zhang, Y. Su, X. Niu, N. Jaitly, Kglens: A parameterized knowledge graph solution to assess what an llm does and doesn't know, arXiv preprint arXiv:2312.11539 (2023).

[21] N. Matsumoto, J. Moran, H. Choi, M. E. Hernandez, M. Venkatesan, P. Wang, J. H. Moore, KRAGEN: a knowledge graph-enhanced RAG framework for biomedical problem solving using large language models, Bioinformatics 40 (2024) btae353. URL:

https://doi.org/10.1093/bioinformatics/btae353. doi:10.1093/bioinformatics/btae353. arXiv:https://academic.oup.com/bioinformatics/article-pdf/40/6/btae353/58186419/btae
[22] W. Xie, X. Liang, Y. Liu, K. Ni, H. Cheng, Z. Hu, Weknow-rag: An adaptive approach for retrieval-augmented generation integrating web search and knowledge graphs, arXiv preprint arXiv:2408.07611 (2024).
[23] D. Allemang, J. Sequeda, Increasing the llm accuracy for question answering: Ontologies to the rescue!, arXiv preprint arXiv:2405.11706 (2024).
[24] N. Tufek, A. Saissre, A. Hanbury, Validating semantic artifacts with large language models, in: Proceedings of the 21th European Semantic Web Conference (ESWC), Krete, Greece, 2024, pp. 24–30.
[25] M. Monti, O. Kutz, G. Righetti, N. Troquard, Improving the accuracy of black-box language models with ontologies: a preliminary roadmap, in: Proceedings of the Joint Ontology Workshops, 2024.
[26] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford, et al., Gpt-4o system card, arXiv preprint arXiv:2410.21276 (2024).
[27] J. J. Pan, J. Wang, G. Li, Survey of vector database management systems, The VLDB Journal 33 (2024) 1591–1615.
[28] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al., Chain-of-thought prompting elicits reasoning in large language models, Advances in neural information processing systems 35 (2022) 24824–24837.
[29] C. van der Lee, A. Gatt, E. van Miltenburg, E. Krahmer, Human evaluation of automatically generated text: Current trends and best practice guidelines, Comput. Speech Lang. 67 (2021) 101151.

Acknowledgments

The work of Mattia Macrì has been supported by the PhD fellowship Pubblica Amministrazione DM118 - CUP B83C22003460006. The work of Marco Calamo and Filippo Bianchini has been supported by the Next-Generation EU (Italian PNRR - M4 C2, Invest 1.3 - D.D. 1551.11-10-2022), named PE4 - MICS (Made in Italy - Circular and Sustainable).

Declaration on Generative AI

During the preparation of this work, the authors used GPT-4o-2024-08-06 in order to: Grammar and spelling check.