Challenges in the Convergence of LLMs and Knowledge Graphs for Air Traffic Information Systems: An Empirical Study on Granularity and the Impact of Advanced RAG

Douglas Silva Teixeira[1], José Maria Parente de Oliveira[1] and Nauane Linhares do Nascimento[1]

[1]*Technological Institute of Aeronautics (ITA), Praça Marechal Eduardo Gomes, 50 - Vila das Acácias, São José dos Campos - SP, 12228-900, Brazil*

Abstract

The increasing complexity of critical domains like air traffic management necessitates robust information systems capable of transforming vast datasets into actionable insights. Large Language Models (LLMs) hold significant promise for this, but their effective deployment requires anchoring them to specific, reliable knowledge bases to mitigate hallucination risks. This study presents a system integrating LLMs, Retrieval-Augmented Generation (RAG), and Knowledge Graphs (KGs) to answer queries related to Brazilian air traffic, drawing from operational flight data, regulatory documents, and incident reports. Our empirical findings reveal a critical dependency on knowledge granularity and the judicious application of advanced RAG techniques. Paradoxically, sophisticated RAG methods, such as re-ranking and summarization, were observed to compromise response accuracy and even induce "systemic hallucinations" when applied to broad, general-purpose KGs. In contrast, smaller, dynamically constructed, and highly focused KGs consistently yielded more precise and reliable answers. This work underscores the practical challenges in aligning ontological knowledge with LLMs, emphasizing the need for careful design considerations regarding knowledge granularity and RAG strategy in critical domain applications.

Keywords

LLM, Knowledge Graphs, RAG, Ontology, Air Traffic, Granularity, Hallucination

CEUR-WS.org/Vol-4176/ontollm-4.pdf

1. Introduction

1.1. Motivation

Large Language Models (LLMs) have rapidly emerged as transformative tools for information processing, significantly boosting productivity across various domains since the launch of ChatGPT in November 2022. Their advanced capabilities in understanding and generating human-like text make them highly promising for complex, data-rich environments. In the context of air traffic management, a sector characterized by vast operational data, intricate regulations, and critical safety requirements[1], LLMs offer a powerful avenue to enhance information accessibility and operational intelligence, crucial for safety and real-time decision-making. Specifically, considering the Brazilian air traffic system, which is frequently impacted by avoidable aeronautical occurrences and persistent delays, the application of LLMs can become an invaluable tool for monitoring and real-time decision support. These models possess the potential to analyze diverse data points such as meteorological conditions, holding patterns, and landing/takeoff schedules, thereby generating actionable insights. Consequently, they can address a wide range of inquiries, from passenger questions about airline reliability to providing critical support for reducing delays and optimizing airport operational efficiency.

1.2. Problem Context

Despite the continuous collection of extensive data by Brazilian aviation authorities, including DECEA, ANAC, and CENIPA, the challenge of converting this raw information into practical and actionable in-

Proceedings of the Joint Ontology Workshops (JOWO) - Episode XI: The Sicilian Summer under the Etna, co-located with the 15th International Conference on Formal Ontology in Information Systems (FOIS 2025), September 8-9, 2025, Catania, Italy

douglasst102@gmail.com (D. S. Teixeira); jose.parente@gp.ita.br (J. M. P. d. Oliveira); nauane.linhares@gmail.com (N. L. d. Nascimento)

© 2025 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073

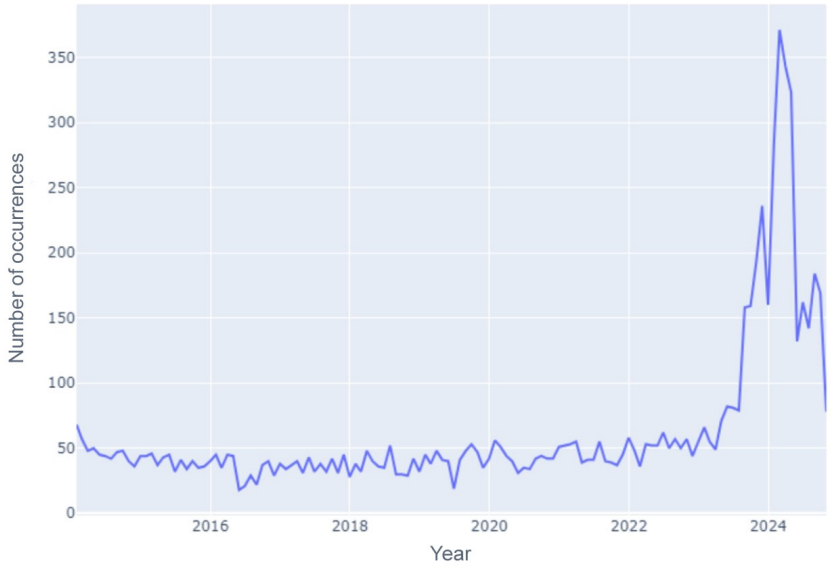

Figure 1: Time series of the number of aeronautical occurrences in Brazil

sights remains significant. While LLMs are powerful, they are inherently prone to ”hallucinations”—generating plausible but factually incorrect information. This necessitates the development of robust mechanisms to ground their responses in verified and authoritative knowledge. Furthermore, despite advancements in air traffic management in Brazil, significant challenges persist that compromise the efficiency and safety of operations. Many airports frequently operate near or above their maximum capacity, leading to recurrent delays and congestion, and sometimes necessitating flight diversions. Adverse meteorological conditions, such as intense rainfall, fog, and storms, also directly impact operations, causing further delays and cancellations. Beyond operational difficulties, aeronautical occurrences represent a relevant and recurring problem. Data from the System for Investigation and Prevention of Aeronautical Accidents (SIPAER) and investigations conducted by the Center for Investigation and Prevention of Aeronautical Accidents (CENIPA) indicate that many incidents and accidents are caused by known factors that could be avoided with more rigorous application of existing standardization and regulatory documents. The repetition of failures across different scenarios suggests that issues such as adherence to safety procedures and personnel training still require attention. Notably, in recent years, the number of aeronautical occurrences has significantly increased[2], as shown in Figure 1, highlighting a concerning situation that demands preventive measures and greater care in aviation. This underscores the critical need for effective LLM integration with external, authoritative data sources to overcome these challenges.

1.3. Objective

This study develops and evaluates a system combining LLMs, Retrieval-Augmented Generation (RAG), and Knowledge Graphs (KGs) to answer complex queries related to Brazilian air traffic. Our aim is to transform raw data into precise, practical answers. Specifically, the work seeks to develop LLM-based models capable of responding to inquiries across three main areas:

- Information about flights, addressing questions such as the possibility of delays due to meteorological conditions.
- Risk assessment based on previously recorded aeronautical incidents and occurrences.
- Consultations on Brazilian regulations to clarify doubts regarding aviation rules.

1.4. Contribution

This paper offers empirical insights into the practical challenges of integrating LLMs with ontological knowledge. Through an air traffic case study, we demonstrate the nuanced impact of knowledge granularity and advanced RAG techniques. Our findings show that while the LLM-RAG-KG paradigm is powerful, advanced RAG (e.g., re-ranking, summarization) can paradoxically reduce accuracy and induce hallucinations with broad KGs. Conversely, smaller, dynamically focused KGs consistently yield superior results, providing valuable lessons for reliable, ontology-driven LLM applications in critical real-world scenarios.

2. Related Works

The proposed system leverages the synergistic capabilities of Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), and Knowledge Graphs (KGs). Understanding these components and their interplay is crucial for appreciating the challenges and successes observed in our study.

2.1. Large Language Models (LLMs)

LLMs are advanced neural networks, typically based on the transformer architecture [3], trained on vast corpora of text data. Their primary function is to predict the next word or sequence of words, enabling them to generate coherent and contextually relevant text. LLMs can be categorized into decoder-only (e.g., GPT-3), encoder-only (e.g., BERT), and encoder-decoder models (e.g., T5), each excelling in different natural language processing (NLP) tasks. Despite their remarkable capabilities, LLMs inherently suffer from limitations, notably the propensity for "hallucinations"—generating factually incorrect or nonsensical information—and biases inherited from their training data [4]. These limitations underscore the necessity of external mechanisms to ground their responses in verifiable facts.

2.2. Retrieval-Augmented Generation (RAG)

RAG is a paradigm designed to enhance LLM outputs by enabling them to reference specific, external data sources. A basic RAG pipeline involves three stages: indexing, retrieval, and generation. During indexing, raw data (e.g., documents, databases) is processed, segmented into manageable chunks, embedded into vector representations, and stored in a vector database. Upon receiving a user query, the system retrieves the most semantically similar chunks from this database. The hallucinations generated by LLMs pose significant problems, especially in specific applications where incoherent responses arise from insufficient context. This is precisely where RAG intervenes: by indexing and retrieving data, the text generation by the LLM is significantly improved, reducing hallucinations and ensuring coherent responses. These retrieved chunks, along with the original query, are then fed to the LLM as augmented context, guiding its generation process and significantly reducing hallucinations. Advanced RAG techniques [5] further refine this process through pre-retrieval and post-retrieval optimizations. Pre-retrieval strategies focus on optimizing the indexing structure (e.g., metadata insertion, mixed retrieval) and query clarity (e.g., query rewriting, text transformation). Post-retrieval steps involve re-ranking the retrieved chunks to prioritize relevance and context compression to select only the most critical sections, preventing information overload for the LLM. While these advanced methods aim to improve precision and efficiency, their impact on response quality, particularly concerning knowledge granularity, is a central focus of our investigation.

2.3. Knowledge Graphs (KGs) and Ontologies

Knowledge Graphs are structured representations of knowledge, where nodes represent entities (e.g., people, places, concepts) and edges represent the relationships between these entities. KGs provide a formal, explicit, and machine-readable way to organize information, making them highly valuable for

semantic search and reasoning. Ontologies, in this context, can be seen as the schema or conceptualization underlying a KG, defining the types of entities, attributes, and relationships that exist within a specific domain. They provide a shared vocabulary and a hierarchical structure that enables consistent interpretation and integration of data. KGs are particularly powerful in anchoring LLMs by providing a structured, factual backbone. Instead of relying solely on the LLM's internal, parametric knowledge (which can be prone to hallucinations), KGs offer a verifiable source of truth. This synergy allows LLMs to generate responses that are not only fluent but also factually accurate and contextually relevant, by navigating the structured relationships within the KG. For instance, KGs have been successfully applied in customer service question answering to retrieve relevant information based on structured relationships [6].

This synergy allows LLMs to generate responses that are not only fluent but also factually accurate and contextually relevant, by navigating the structured relationships within the KG. Existing research highlights several successful implementations of this integration. Knowledge graphs effectively restrict the context of analysis for LLMs, mitigating information overload and improving performance in tasks like relation extraction by dynamically filtering irrelevant data [7] [8]. Beyond simple factual retrieval, contextualized KGs further enhance LLM reasoning capabilities by incorporating richer contextual information like temporal validity or provenance [9]. Hybrid models leveraging KGs demonstrate improved factual accuracy, address knowledge gaps inherent in LLM training data, and significantly enhance contextual awareness by providing specific domain knowledge [10] [11]. Furthermore, integrating topological information from KGs can refine context restriction and improve tasks such as knowledge graph completion [12]. These combined approaches have shown promising results in reducing hallucinations and producing more reliable outputs, leading to improved computational efficiency and scalability in real-world applications across various domains, such as customer service question answering [6]. The effectiveness of this integration, however, remains heavily influenced by the design and granularity of the KG itself, a central focus of our study.

3. Methodology

Our system's design integrates diverse data sources to provide a comprehensive and accessible information retrieval platform for Brazilian air traffic. The methodology is structured around three distinct analytical lines: historical flight data, aeronautical regulations, and aviation occurrence reports. All implementations were developed in Python, leveraging its extensive ecosystem for LLM applications, ensuring flexibility and broad accessibility.

3.1. Overall Architecture

The overarching architecture of the developed system, as shown in Figure 2, is designed to offer users three primary modes of inquiry, each tailored to specific data types and information needs. To empirically validate the benefits of KG and RAG integration, a comparative baseline evaluation was conducted against a pure LLM (without KG or RAG assistance) to highlight improvements in factual accuracy and domain-specific reasoning. In essence, the system allows users to access contextualized and precise answers, supported by a broad base of data and regulatory documents, adapting to the user's specific query requirements. The first mode of consultation is based on historical flight data from ICEA and ANAC's VRA. The second leverages knowledge graphs representing Brazilian civil aviation regulations and norms. The third dynamically generates knowledge graphs from specific user searches concerning aeronautical occurrences. This multi-faceted approach employs LLMs, Retrieval-Augmented Generation (RAG), and Knowledge Graphs (KGs) in configurations optimized for each data type, aiming to maximize response accuracy and relevance while minimizing hallucinations.

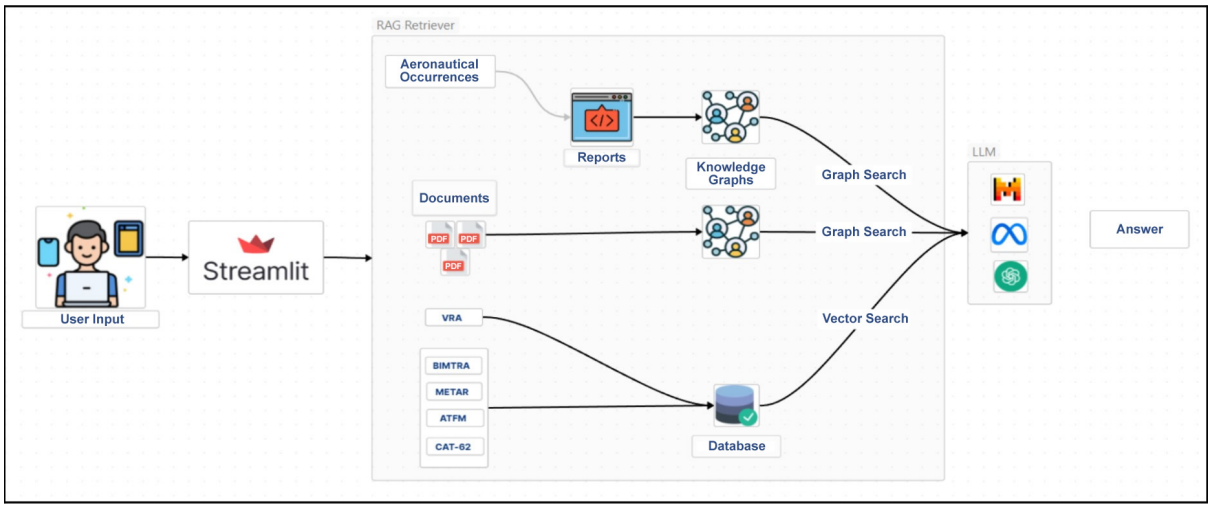

Figure 2: Architecture of the developed system

3.2. Data Sources and Pre-processing

3.2.1. Historical Flight Data

For analyzing past flight operations, we integrated data from two primary sources: the Active Regular Flight (VRA) database provided by the National Civil Aviation Agency (ANAC)[2] and the Institute of Airspace Control (ICEA) API. The VRA database offers detailed flight information, including scheduled and actual times, cancellations, and operational status. The ICEA API, crucial for meteorological and traffic insights, provides access to several key datasets:

- BIMTRA (Air Traffic Movement Information Bank): Contains information on air traffic movements, including basic flight data like departure times and estimated arrival.
- CAT-62 (Radar Synthesis Data): Provides historical radar synthesis data.
- Holds (Flight Hold Quantity Data per Hour): Contains historical information on the quantity of flight holds per hour at aerodromes.
- METAF (Aerodrome Terminal Forecast): Offers meteorological forecasts for aerodromes.
- METAR (Aerodrome Meteorological Report): Contains meteorological reports for aerodromes.
- Meteorological Satellite: Provides meteorological data from satellites in images.
- Aerodrome Runway Change Forecast: Offers historical forecasts on runway changes at aerodromes.
- Aerodrome Runway Change History: Contains information on historical runway changes at aerodromes.

Data from these sources, covering 12 major Brazilian airports from March 2022 to May 2023, were meticulously integrated into a Pandas DataFrame. This DataFrame included comprehensive flight details, meteorological conditions at departure/arrival, and various operational statuses (e.g., delays, cancellations). The explicit purpose of this analytical line was to evaluate how the LLM could effectively answer questions about historical flight data, offering useful information to the user on practical issues such as airline reliability or the probability of delays and cancellations under unfavorable meteorological conditions. To enable semantic search on this structured data, the LLM was designed to interpret user queries and dynamically generate executable Python code. This code would then query the DataFrame, extract relevant information, and the LLM would synthesize the final response. This innovative approach effectively transforms structured data into a natural language interface, allowing the LLM to act as a sophisticated data analyst, providing accessible insights into historical performance and operations.

3.2.2. Aeronautical Regulations

This analytical line focused on building a comprehensive knowledge graph representing Brazilian civil aviation regulations. The construction of this knowledge graph followed established ontology engineering principles, beginning with a detailed domain analysis. This process benefited from subject matter expertise in Brazilian air traffic regulations, guiding the formal definition of the ontology schema. Entities (e.g., *Regulation*, *Aircraft Type*, *Procedure*, *Requirement*) and their relationships (e.g., *governs*, *applies_to*, *specifies*, *requires*) were carefully identified and prioritized to represent the regulatory framework. While not based on a pre-existing top-level ontology, the schema was designed to capture the semantic nuances critical for rule-based queries. Subsequently, LlamaIndex facilitated document ingestion and chunking, while the LLM (GPT-4o-mini via API) was employed to extract instances of these predefined entities and relationships from the textual content of the PDF regulations, populating the Neo4j graph database. This semi-automated approach ensured adherence to the expert-defined schema, forming a broad and extensive regulatory knowledge graph.

We processed various regulatory documents in PDF format, including selected Brazilian Civil Aviation Regulations (RBACs 39, 43, 91)[13], Air Command Instructions (ICAs 100-37, 100-12, 81-1), and the Operational Norm for Flow Management (NOGEF)[14]. The construction of this knowledge graph utilized LlamaIndex for document ingestion and chunking, and Neo4j as the graph database to store entities and their relationships. The LLM (GPT-4o-mini via API) was employed to extract entities and relationships from the textual content of these regulations, forming a broad and extensive knowledge graph. This graph aimed to cover a wide range of regulatory information, making it inherently less granular for highly specific queries, as the relevant information might be embedded within a vast network of interconnected but distantly related concepts. For information retrieval, an advanced RAG pipeline was implemented, incorporating re-ranking and summarization techniques (specifically using Cohere Rerank). Cohere Rerank was configured with a minimum relevance score threshold of 0.7 and selected the top 5 most relevant documents. Summarization, utilizing a ‘TreeSummarize‘ method, was constrained to a maximum of 512 tokens per response. The intention was for these advanced RAG methods to refine the retrieved context before feeding it to the LLM, thereby improving response quality. However, as detailed in the Results section, the interaction between this broad KG and advanced RAG presented unexpected challenges regarding response accuracy, particularly when the specific information was deeply nested or required precise extraction from a large context.

3.2.3. Aeronautical Occurrences

The third analytical line focused on leveraging historical aviation incidents and accidents to provide insights for prevention and understanding. Data was sourced from the System for Investigation and Prevention of Aeronautical Accidents (SIPAER) and investigation reports from the Center for Investigation and Prevention of Aeronautical Accidents (CENIPA). Unlike the regulatory knowledge base, this approach adopted a dynamic and focused knowledge graph strategy. Upon a user query about an occurrence (e.g., ”engine malfunction incidents”), the system first performed a semantic search within a Pandas DataFrame containing a summary of occurrences to identify relevant incidents. For each identified incident, the system then retrieved its specific, detailed report from CENIPA/SIPAER. A new, highly focused knowledge graph was then constructed ad-hoc from the content of these specific reports.

This dynamic knowledge graph construction for aeronautical occurrences similarly leveraged domain insights to define a flexible schema for incident reporting. Unlike the static regulatory KG, these KGs are built *ad-hoc* for each relevant occurrence report. Entities (e.g., *Incident*, *Aircraft*, *Cause*, *Location*, *Consequence*) and relationships (e.g., *involved_in*, *caused_by*, *occurred_at*, *resulted_in*) are extracted on-the-fly by the LLM from the specific, detailed reports retrieved from CENIPA/SIPAER. This approach ensures maximal granularity and contextual relevance for each queried incident, adapting the KG structure to the specific narrative of the report.

This approach ensured that the knowledge graph used for generating the final response was highly

granular and contextually relevant to the user's specific query, minimizing extraneous information. The RAG pipeline then operated on this focused KG, allowing the LLM to provide precise and detailed answers grounded in specific incident reports. This methodology aligns with the LinkedIn customer service model [6], adapting it to retrieve accident reports instead of customer tickets, thereby providing a powerful tool for analyzing and preventing future aviation incidents.

3.3. Tools and Technologies

The project utilized several key frameworks and tools to implement the described methodologies. LlamaIndex facilitated efficient data ingestion and indexing, crucial for preparing diverse data sources for LLM applications. Neo4j served as the robust graph database, enabling the storage, management, and querying of complex knowledge graphs. Cohere Rerank was integrated into the advanced RAG pipeline, specifically for the aeronautical regulations domain, to re-rank retrieved documents and enhance relevance. An interactive web interface was developed using Streamlit, making the system accessible and user-friendly for various types of queries. For the primary LLM operations across all three analytical lines, OpenAI's GPT-4 was employed via its API, chosen for its advanced capabilities and computational efficiency in a cloud-based environment.

LLM interactions were managed through dynamically constructed prompts, tailored for specific tasks such as structured data extraction from DataFrames or populating knowledge graphs. The core RAG pipeline architecture integrates the LLM with a retriever, a re-ranker, and a summarizer. Specifically, OpenAI's GPT-4o was chosen as the LLM, configured with a temperature of 0.25 to balance response creativity and factual consistency. Data indexing for the knowledge graphs leveraged Neo4j with 'PropertyGraphIndex' for efficient storage and retrieval. While local models (e.g., LLaMA family via Ollama) were explored for comparative analysis, as detailed in Table 1, the computational demands of the comprehensive system necessitated the use of a more powerful, cloud-based LLM for the main study, ensuring optimal performance and accuracy.

Table 1
Comparative LLM Tests

Model	Response Time (s)	Response Coherence/Accuracy (Example: % of Azul cancelled flights)	Parameters (Billions)
GPT-4 (OpenAI)	~2.68	Correct and Fast (2.47%)	~175B[15]
GPT-4o Mini	~2.23	Incoherent (0%)	~8B
Llama 3.1 8b	~54.01	Close, with hallucination (2.63%)	8B
Llama 3.2 1b	~62.05	Incoherent and Slow	1B
Llama 3.2 3b	~20.07	Incoherent	3B

4. Results and Discussion

This section presents the empirical results obtained from evaluating the developed system across its three distinct analytical approaches: historical flight data, aeronautical regulations, and aeronautical occurrences. The evaluation primarily focused on the quality and accuracy of the LLM's responses, particularly in relation to the effectiveness of RAG and knowledge graph integration.

4.1. Baseline Comparison: Pure LLM vs. LLM-KG-RAG

To quantify the value added by our proposed integration, we conducted a comparative analysis of a pure LLM (GPT-4 without RAG or KG integration) against our system. For domain-specific inquiries, such as "What are the requirements for major alterations to an airframe per RBAC 43?", the pure LLM frequently

generated generic, imprecise, or entirely hallucinated responses, demonstrating its inherent limitations in accessing and verifying specialized, external knowledge. For instance, it might provide general advice on regulatory compliance without citing specific RBAC sections or detailing the precise items. In contrast, our LLM-KG-RAG system consistently delivered factually accurate and detailed answers, directly referencing relevant sections of the regulatory documents (e.g., RBAC 43, ICA 100-37), and effectively leveraging the structured knowledge from the KG. This comparison empirically underscores the critical role of grounding LLMs in external, verifiable knowledge bases to mitigate hallucinations and ensure reliability in critical domains like air traffic management, particularly when dealing with information not implicitly contained within the LLM's pre-training data.

4.2. Historical Flight Data

The system demonstrated robust performance when answering objective questions directly queryable from the integrated Pandas DataFrame of VRA and ICEA data. For instance, queries regarding the total number of cancelled flights, the details of the last cancelled flight, or the percentage of flights cancelled by a specific airline like Azul were consistently answered accurately and promptly. The LLM effectively translated natural language queries into executable Python code to extract and process the relevant data, showcasing its capability as a sophisticated data analysis interface. However, challenges emerged with more subjective or inferential questions. For example, when asked about meteorological conditions most likely to cause flight delays, the LLM initially provided a generic response, indicating a lack of direct, pre-computed insights in the raw data. This limitation was mitigated by providing additional context to the LLM, such as specifying which DataFrame columns to analyze. With this guidance, the LLM was able to process the data and provide a more precise answer regarding average delay times under specific conditions. This highlights that while the LLM can perform data analysis, its ability to infer complex patterns from raw data without explicit guidance is limited, suggesting a need for more sophisticated pre-analysis or fine-tuning for such queries. Furthermore, the LLM could not directly render charts but successfully provided the necessary Python code for their generation, demonstrating its utility in supporting data visualization tasks.

4.3. Aeronautical Regulations

This approach involved constructing a broad and extensive knowledge graph from various regulatory documents (RBACs, ICAs, NOGEF) and employing an advanced RAG pipeline with re-ranking and summarization. While the system provided coherent, albeit sometimes generic, answers for certain queries, a critical issue arose with specific, detailed questions. The most striking example occurred with a query related to RBAC 43, asking about items considered major repairs or alterations to an airframe. When the advanced RAG pipeline (with re-ranking and summarization) was active, the LLM generated a completely random and irrelevant response, indicating a significant failure in retrieving or processing the correct information. To diagnose this, the same query was re-executed with the re-ranking and summarization methodologies of the advanced RAG pipeline disabled. In this scenario, the LLM successfully returned the correct and comprehensive list of items directly from the document. This paradoxical outcome suggests that, in the context of a broad and complex knowledge graph, the advanced RAG techniques, intended to refine and prioritize information, inadvertently hindered accuracy. It appears that the re-ranking and summarization processes either filtered out the precise details required, over-summarized the relevant passages, or mis-ranked them due to the sheer volume and interconnectedness of the information in the extensive regulatory KG. This finding is crucial, as it challenges the assumption that more sophisticated RAG always leads to better results, particularly when dealing with large, broadly structured knowledge bases. It implies that the "optimization" introduced by advanced RAG can, in certain scenarios, induce a form of "systemic hallucination" by distorting or omitting critical information. The extent to which this "systemic hallucination" generalizes across different advanced RAG configurations and broader knowledge domains remains an intriguing area for future research. While this observation in the regulatory context is compelling, it is important

to note that the full generality of this phenomenon—where advanced RAG potentially compromises accuracy on broad KGs—requires further investigation across diverse knowledge domains and with various re-ranking and summarization algorithms.

4.4. Aeronautical Occurrences

In contrast to the regulatory domain, the approach for aeronautical occurrences adopted a dynamic and focused knowledge graph strategy, which yielded significantly more precise and consistent results. When queried about incidents involving engine malfunction or bird strikes, the system successfully identified relevant cases, even correcting a misspelled keyword. It then generated focused knowledge graphs and provided detailed, critical analyses of the incidents, including their severity. A particularly illustrative case was the query about the recent Vinhedo-SP accident. The system accurately retrieved the incident details, provided a link to the official report, and, upon further request, offered comprehensive information about the accident's specifics, including the number of affected individuals and structural details. Crucially, a complex and detailed knowledge graph was dynamically generated specifically for this accident report. This success highlights the effectiveness of using highly granular and contextually focused knowledge graphs. By first identifying the specific occurrence and then building a dedicated KG from its detailed report, the RAG pipeline operated on a much cleaner and more relevant information set, minimizing noise and maximizing the LLM's ability to extract and synthesize accurate information. This multi-step approach, where initial broad search leads to the creation of a highly specific KG, proved to be robust and less prone to the issues observed with the broad regulatory KG.

4.5. Comparative Analysis and Lessons Learned

The empirical results from the three analytical lines provide critical insights into the effective integration of LLMs, RAG, and KGs.

A summary of the observed performance across different system configurations and query types is presented in Table 2, visually illustrating the critical dependencies discussed.

Table 2
System Configuration Performance Analysis

System Configuration / Query Type	Observed Accuracy / Coherence	Key Finding Illustrated
Pure LLM (No KG/RAG)	Generic / Often Inaccurate / Hallucinated for domain-specific queries	Lack of external knowledge
LLM + Dataframe Query (Historical Flight Data)	High for objective queries; requires guidance for subjective ones	Effective data analysis interface
Broad KG (Regulatory) + Basic RAG	Accurate for specific factual queries (e.g., RBAC 43 details without summarization)	KG structure helps retrieval
Broad KG (Regulatory) + Advanced RAG	Inaccurate / Hallucinated for specific factual queries (e.g., RBAC 43 details with summarization)	Advanced RAG detrimental to precision on broad KGs
Focused KG (Occurrence) + Advanced RAG	High for detailed and contextual queries	Effective with granular context

The most significant finding is the paramount importance of knowledge granularity. For structured data (historical flight data), the LLM's ability to generate and execute code on a DataFrame proved effective for objective queries, but required more explicit guidance for subjective analyses. This suggests that while LLMs can act as data analysts, their inferential capabilities on raw structured data might need further enhancement or pre-processing. For knowledge graphs, a clear dichotomy emerged.

The broad, general-purpose KG for aeronautical regulations, when combined with advanced RAG techniques (re-ranking and summarization), led to a paradoxical decrease in accuracy for specific queries. This suggests that these RAG optimizations, while beneficial for general information retrieval, can become detrimental when the target information is highly specific and embedded within a vast, interconnected knowledge base. They might inadvertently prune or dilute the precise context required by the LLM, leading to ”systemic hallucinations” where the LLM generates plausible but incorrect answers. Conversely, the dynamic, highly focused KGs generated for specific aeronautical occurrences consistently demonstrated superior performance in our empirical study. By narrowing the scope of the KG to a single, relevant document or event, the RAG pipeline operated on a much higher level of contextual granularity. This allowed the LLM to extract and synthesize information with remarkable precision and detail, effectively grounding its responses in verifiable facts. Therefore, a key lesson derived from this study is that the success of LLM-KG integration is not merely about the presence of a knowledge graph or the sophistication of the RAG pipeline, but critically depends on the alignment of knowledge granularity with the query’s specificity. For complex domains, a ”divide and conquer” strategy, where initial broad searches lead to the creation or selection of highly specific and granular KGs for subsequent detailed querying, appears to be a more robust and reliable approach. This adaptive granularity ensures that the LLM receives the most relevant and least noisy context, mitigating the risk of hallucinations and maximizing factual accuracy.

5. Conclusion and Future Work

5.1. Conclusion

This study explored the application of Large Language Models (LLMs) combined with Knowledge Graphs (KGs) and Retrieval-Augmented Generation (RAG) to answer questions related to Brazilian air traffic. We developed and evaluated a system across three distinct approaches: historical flight data, aeronautical regulations, and aviation occurrence reports. For historical flight data, the LLM demonstrated strong performance in answering objective queries by dynamically generating and executing Python code on structured data. However, its ability to infer complex patterns from raw data for subjective questions required more explicit guidance, suggesting areas for further pre-processing or fine-tuning. The approach involving aeronautical regulations, which utilized a broad and extensive knowledge graph with an advanced RAG pipeline (including re-ranking and summarization), yielded a critical insight. Paradoxically, these advanced RAG techniques, intended to enhance precision, were observed to contribute to a compromise in response accuracy and, in some cases, potentially induce “systemic hallucinations” when applied to the vast and complex regulatory KG. The case of RBAC 43 vividly illustrated this, where disabling the advanced RAG components led to accurate retrieval, while their activation resulted in an irrelevant response. This suggests that the optimization processes within advanced RAG can inadvertently filter out or distort precise information when the knowledge base is too broad and the target information is highly specific. In stark contrast, the approach for aeronautical occurrences, which employed a dynamic strategy of creating highly focused and granular knowledge graphs for specific incidents, consistently demonstrated superior and more reliable results in our evaluation. By narrowing the scope of the KG to the context of a single, relevant report, the RAG pipeline operated on a cleaner, more precise information set, enabling the LLM to provide remarkably accurate and detailed answers. In summary, our findings underscore that the effectiveness of integrating LLMs with KGs and RAG is not solely dependent on the sophistication of the components but critically hinges on the granularity of the knowledge and the judicious application of RAG techniques. Broad, general-purpose KGs, even with advanced RAG, can lead to diminished accuracy for specific queries, while highly granular and contextually focused KGs were found to be significantly more effective in the contexts studied. This implies that for complex domains, a “divide and conquer” strategy—where initial broad searches lead to the creation or selection of highly specific and granular KGs for subsequent detailed querying—appears to be a more robust and reliable approach based on our findings.

5.2. Limitations

This study's evaluation was primarily qualitative, relying on manual assessment of response accuracy, which limited the ability to rigorously quantify consistency or compare configurations in a statistically significant way. While insightful, this limits the generalizability and scalability of the findings. Furthermore, the computational resources available constrained the extensive testing of local LLMs. As detailed in Table 1, the performance of smaller, open-source models on local hardware was significantly lower compared to cloud-based solutions like GPT-4, influencing our choice for the main system implementation.

5.3. Future Work

Building upon these insights, future work will focus on several key areas:

- Quantitative Evaluation Framework: Develop robust quantitative metrics and automated testing frameworks to systematically evaluate response accuracy, relevance, and factual consistency across different LLMs, RAG configurations, and KG granularities. This will enable more rigorous comparisons and validation.
- Adaptive RAG Strategies: Investigate and develop adaptive RAG strategies that dynamically adjust their behavior (e.g., re-ranking thresholds, summarization aggressiveness) based on the detected granularity of the query and the underlying knowledge graph.
- Dynamic KG Optimization: Explore more efficient and intelligent methods for dynamic KG creation and fusion from diverse, unstructured, and semi-structured data sources, ensuring optimal granularity for specific queries.
- Integration of Real-time Data: Incorporate real-time data streams (e.g., from FlightRadar24[16]) to enhance the system's predictive capabilities and provide up-to-the-minute information, further enriching the knowledge base.
- LLM Architecture Exploration: Conduct further research into the performance of smaller, fine-tuned LLMs and alternative open-source models, potentially leveraging more powerful hardware or distributed computing, to assess their viability for critical domain applications.
- Mitigating Systemic Hallucinations: Deepen the understanding of how advanced RAG techniques can induce hallucinations on broad KGs and develop specific countermeasures or alternative architectures to prevent such occurrences.
- User Experience and Visualization: Enhance the user interface to provide more intuitive interactions and advanced visualization tools for complex information, including interactive knowledge graph exploration.

For reproducibility and further exploration, the source code and implementation details of this project are publicly available at: https://github.com/nauanelinhares/graphrag-with-aircraft-accident-reports

Declaration on Generative AI

During the preparation of this work, the authors used Gemini-2.5 to: Grammar and spelling check. After using these services, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

[1] DECEA, Statistical Yearbook 2023, Technical Report, Infraero, Brasília, Brazil, 2023. URL: https://transparencia.infraero.gov.br/estatisticas/.

[2] National Civil Aviation Agency (ANAC), Open data - anac, 2024. URL: https://www.gov.br/anac/pt-br/acesso-a-informacao/dados-abertos, accessed: 2024-11-20.

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Łukasz Kaiser, I. Polosukhin, Attention is all you need, in: Advances in Neural Information Processing Systems 30 (NIPS 2017), 2017, pp. 5998–6008.

[4] L. Weidinger, I. Gabriel, C. Griffin, M. Rauh, J. Uesato, J. Mellor, W. Isaac, P.-S. Huang, L. A. Hendricks, M. W. Cheng, M. Glaese, B. Balle, A. Kasirzadeh, Z. Kenton, S. Brown, W. Hawkins, T. Stepleton, C. Biles, A. Birhane, J. Haas, L. Rimell, S. Legassick, G. Irving, Ethical and social risks of harm from language models, 2021. URL: https://arxiv.org/abs/2112.04359. arXiv:2112.04359.

[5] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, H. Wang, Retrieval-augmented generation for large language models: A survey, 2024. URL: https://arxiv.org/pdf/2312.10997. arXiv:2312.10997.

[6] Z. Xu, M. J. Cruz, M. Guevara, T. Wang, M. Deshpande, X. Wang, Z. Li, Retrieval-augmented generation with knowledge graphs for customer service question answering, 2024. URL: https://arxiv.org/abs/2404.17723v2. arXiv:2404.17723.

[7] A. Nadgeri, A. Bastos, K. Singh, I. O. Mulang', J. Hoffart, S. Shekarpour, V. Saraswat, KGPool: Dynamic knowledge graph context selection for relation extraction, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, Association for Computational Linguistics, Online, 2021, pp. 535–548. URL: https://aclanthology.org/2021.findings-acl.48/. doi:10.18653/v1/2021.findings-acl.48.

[8] Z. Zheng, J. Wang, Y. Li, S. Li, Y. Xu, KG-CF: Knowledge Graph Completion with Context Filtering under the Guidance of Large Language Models, 2025. arXiv:2501.02711.

[9] C. Xu, A. Singh, P. Kumar, S. Goel, S. Samsi, M. Galkin, Context graph, 2024. arXiv:2406.11160.

[10] V. AI, Building accountable llms with knowledge graphs, 2024. URL: https://valkyrie.ai/post/building-accountable-llms-with-knowledge-graphs/, accessed: 2024-04-01.

[11] B. Liu, R. Hao, J. Yang, W. Zhou, H. Chen, X. Zhao, W. Zhang, Large language models for knowledge graph embedding: A survey, 2025. arXiv:2501.07766.

[12] U. M. Sehwag, S. Y. Kumar, S. Maharana, R. Mukherjee, S. Bhowmick, In-context learning with topological information for knowledge graph completion, 2024. arXiv:2412.08742.

[13] National Civil Aviation Agency (ANAC), RBAC 91 - General Operating Requirements for Civil Aircraft, Technical Report, National Civil Aviation Agency, 2024. URL: https://www.anac.gov.br/assuntos/legislacao/legislacao-1/boletim-de-pessoal/2024/bps-v-19-no-12-18-a-22-03-2024/rbac-91-emd-04/visualizar_ato_normativo.

[14] DECEA, Operational Standard for Air Traffic Flow Management (NOGEF), Technical Report, Department of Airspace Control, 2018. URL: http://portal.cgna.decea.mil.br/files/uploads/legislacao/NOGEF-2018.pdf.

[15] SEMAFOR, Gpt-4 has a trillion parameters, 2024. URL: https://the-decoder.com/gpt-4-has-a-trillion-parameters/, accessed: 2024-11-09.

[16] FLIGHTRADAR24, Flightradar24, 2024. URL: https://www.flightradar24.com/52.61,-3.48/9, accessed: 2024-11-08.