Article

An Agent-Based RAG Architecture for Intelligent Tourism Assistance: The Valencia Case Study

Andrea Bonetti †, Adrián Salcedo-Puche †, Joan Vila-Francés *,†, Xaro Benavent-Garcia †, Emilio Fernández-Vargas †, Rafael Magdalena-Benedito †, Emilio Soria-Olivas †

Intelligent Data Analysis Laboratory (IDAL) – Department of Electronic Engineering, ETSE (Engineering School), Universitat de València (UV), Av. Universitat, sn, 46100 Burjassot, Valencia, Spains

* Correspondence: joan.vila@uv.es

† These authors contributed equally to this work.

Abstract

The contemporary digital landscape overwhelms visitors with fragmented and dynamic information, complicating travel planning and often leading to decision paralysis. This paper presents a real-world case study on the design and deployment of an intelligent tourism assistant for Valencia, Spain, built upon a Retrieval-Augmented Generation (RAG) architecture. To address the complexity of integrating static attraction data, live events, and geospatial context, we implemented a multi-agent system comprising specialized Retrieval, Events, and Geospatial Agents. Powered by a large language model, the system unifies heterogeneous data sources — including official tourism repositories and OpenStreetMap — within a single conversational interface. Our contribution centers on practical insights and engineering lessons from developing RAG in an operational urban tourism environment. We outline data preprocessing strategies such as coreference resolution to improve contextual consistency and reduce hallucinations. System performance is evaluated using Retrieval Augmented Generation Assessment (RAGAS) metrics, yielding quantitative results that assess both retrieval efficiency and generation quality, with the Mistral Small 3.1 model achieving an Answer Relevancy score of 0.897. Overall, this work highlights both the challenges and advantages of using agent-based RAG to manage urban-scale information complexity, providing guidance for developers aiming to build trustworthy, context-aware AI systems for smart destination management.

Keywords: retrieval-augmented generation (RAG); agent-based conversational AI; geospatial information; evaluation workflow

1. Introduction

Travelers increasingly rely on the Internet to plan their trips, yet face an overwhelming volume of fragmented and dynamic information. Online sources—ranging from destination guides and accommodation reviews to travel blogs and social media posts—offer unprecedented access to recommendations but also create cognitive overload. Navigating these options can be daunting, often resulting in decision paralysis and dissatisfaction with the planning process [1]. This complexity motivates the need for intelligent systems that provide context-aware, trustworthy, and easily navigable tourism information.

Tourist-oriented AI systems have been explored extensively in both academic and commercial contexts. Foundational research investigated the use of recommender systems and intelligent agents to support trip planning and destination exploration [2,3]. In parallel, commercial platforms such as TripAdvisor and Google Travel offer context-aware recommendations. However, these systems generally rely on static retrieval and ranking methods, lacking advanced multi-step reasoning and the ability to dynamically combine contextual data sources. These limitations highlight the need for intelligent assistants that can integrate reasoning with external information tools to deliver more personalized and cognitively transparent support.

Traditional search and recommendation systems struggle to reconcile heterogeneous and temporally dynamic data—e.g., static attraction descriptions, live event schedules, and geospatial queries—particularly in urban tourism contexts where timely, location-aware suggestions are crucial. Recent efforts in geospatial question-answering systems have sought to address similar challenges through retrieval and structured query generation. For instance, the MapQA framework [4] proposes a retrieval-based approach combined with SQL generation to answer map-related queries, offering a relevant foundation for spatial reasoning in tourism contexts. Retrieval-Augmented Generation (RAG) is a pragmatic approach for such applications because it grounds language model outputs in curated, domain-specific knowledge sources, thereby improving factual consistency and reducing hallucinations [5]. Recent research in domains such as manufacturing and pharmaceuticals has shown that agentic RAG systems—which combine retrieval with reasoning architectures like ReAct to chain multiple retrieval and action steps [6]—significantly improve accuracy and multi-step query resolution. Despite these advances, the systematic application of ReAct-augmented RAG in tourism remains largely unexplored.

Motivated by this gap, we developed an agentic RAG assistant for Valencia (Spain) that orchestrates three specialized agents: (1) a Retrieval Agent for static and semi-structured documents using hybrid search; (2) an Events Agent for near-real-time event ingestion; and (3) a Geospatial Agent that answers location-based queries using OpenStreetMap. The orchestration follows the LangChain [7] ReAct paradigm to interleave reasoning and tool use, enabling the system to handle temporal, spatial, and semantic aspects of travel queries within a single conversational interface—akin to a knowledgeable local guide offering personalized advice.

This study is framed as a real-world case study within the CitCom.ai [8] Testing and Experimentation Facility (TEF), an European project that emphasizes the use of trustworthy AI in urban environments. Accordingly, our contribution focuses on practical engineering lessons and empirical observations from the development and prototyping of a RAG-based tourims assistant, rather than exhaustive benchmarking. Concretely, the core contributions are:

- Preprocessing Impact: A focused analysis of coreference resolution as a data-preprocessing step, quantifying its observed effect on contextual coherence and hallucination reduction in downstream RAG outputs.

- Evaluation of the system: A pragmatic evaluation using RAGAS metrics [9] reports the generation performance of the mixtral-small-24b model (Answer Relevance = 0.837), demonstrating the prototype's ability to produce accurate and contextually relevant responses.

- Agentic Integration: An examination of the multi-agent design as an effective mechanism for orchestrating diverse data modalities (text, temporal, and spatial) within a coherent AI-driven workflow in the prototype setting.

The remainder of the paper is organized as follows. Sections 1.1–1.3 provide a concise background on language models, RAG systems, and agent orchestration. Section 2 describes the methodology, including data collection, preprocessing, model and tool choices, and implementation details. Section 3 presents the RAGAS-based evaluation and qualitative examples from the prototyping. Section 4 discusses engineering lessons, limitations, ethical considerations, and operational trade-offs. Section 5 concludes and outlines concrete next steps for expanding evaluation and reproducibility.

1.1. Brief history and development of language models

Language models initially relied on statistical methods, estimating word and phrase probabilities from large corpora [10–12], but were limited by context length. Neural Language Models later employed neural networks to capture more complex representations and semantic relationships in language [13–15]. The *Transformer* architecture [16] enabled Pretrained Language Models (PLMs), which first learn general linguistic structures from large untagged text corpora and are subsequently fine-tuned for specific NLP tasks [17,18]. Modern Large Language Models (LLMs), such as GPT-3 [19] and GPT-4 [20], leverage billions of parameters and extensive training data, combining large-scale pretraining with alignment to human instructions to achieve strong adaptability across diverse tasks

[17]. Recent advances explore sparse Mixture of Experts (MoE) architectures, such as Mixtral 8x7B [21], which activate only a subset of feedforward experts per token, improving efficiency while maintaining high performance. Collectively, these models represent major milestones in language technology, supporting advanced reasoning and generation capabilities, as utilized in the proposed RAG-based agent.

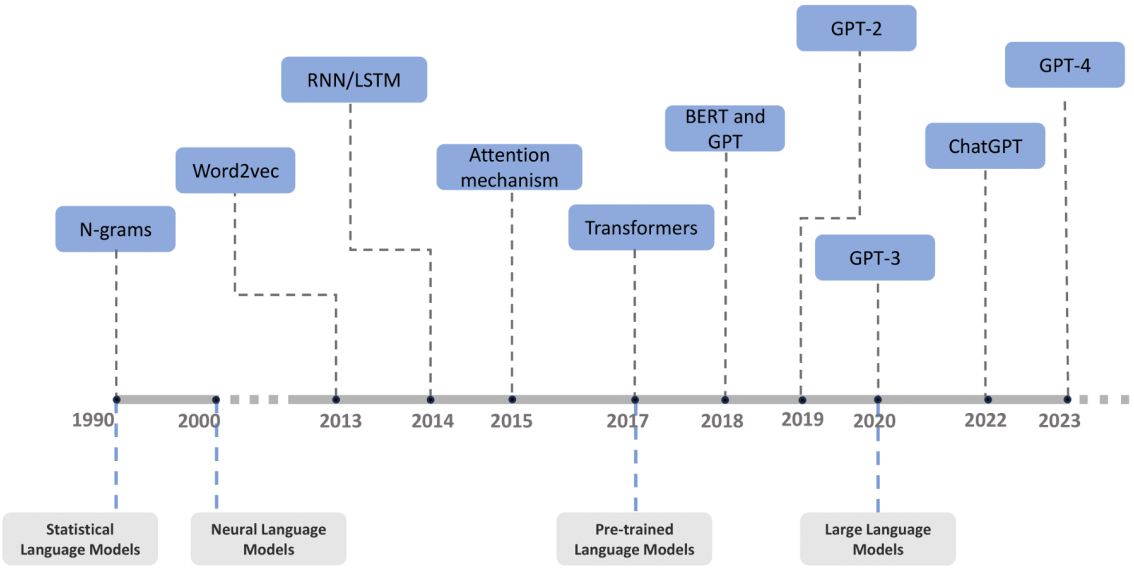

Figure 1. History and development of large language models [**?**].

1.2. RAG systems: principles and practical choices

The Retrieval-Augmented Generation (RAG) methodology enhances LLMs by integrating external knowledge, overcoming their limitations in accessing current information. RAG retrieves relevant document fragments from external sources, which are combined with the original query to formulate enriched questions, enabling the model to generate informed responses (Figure 2). This approach synergistically merges information retrieval with in-context learning, providing crucial context without requiring model fine-tuning, and has become foundational for many conversational systems. The workflow involves three stages: corpus segmentation and indexing via an encoder, retrieval of fragments based on similarity to the query, and synthesis of responses considering the recovered context.

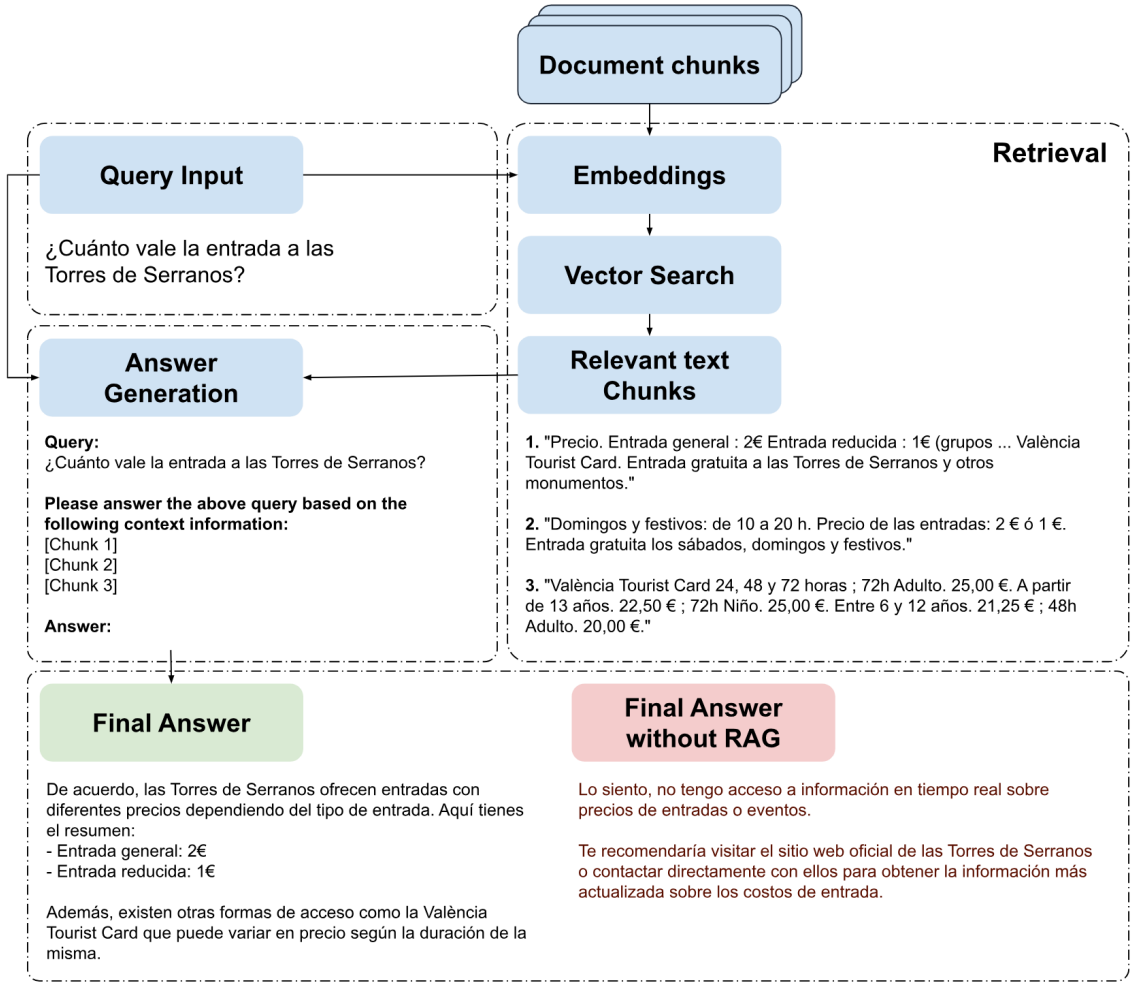

Figure 2. A representative instance of the RAG process applied to question answering. Based on the research [51] (Fig. 2).

Technological advancements have refined RAG along three axes: what, when, and how to retrieve and utilize information. Retrieval has evolved from individual tokens [22] and entities [23] to more structured representations, including slices [24] and knowledge graphs [25], balancing granularity with precision and efficiency. Strategies for when to retrieve range from single retrievals [26,27] to adaptive [28,29] and multiple retrieval methods [30], trading information richness for computational efficiency. Techniques for how to integrate retrieved data span input-level [31], intermediate [32], and output-level [33] integration, with trade-offs in effectiveness, training complexity, and efficiency.

The evolution of RAG can be divided into four phases. The initial phase, starting in 2017 with the Transformer architecture [16], focused on incorporating additional knowledge into pre-trained models (PTMs) to enhance language modeling, emphasizing improvements in pretraining methods.

1.3. Agents and Orchestration Frameworks

In artificial intelligence, an agent is an autonomous system that perceives its environment and acts toward defined goals. Within LLM-based systems, agents operate through a reasoning-acting cycle, invoking external tools such as APIs, databases, or search engines as needed. This paradigm, in which the language model learns to select and utilize external tools to enhance its capabilities, is foundational to tool-augmented LLMs [34]. Complex applications often employ multi-agent frameworks, where specialized components collaborate to solve multi-step queries [35]. LangChain [7] is a widely used framework that provides modular components—including chains, agents, and tools—for orchestrating such pipelines. Building on these orchestration frameworks, ReAct (Reasoning and Acting) [36] introduces an advanced methodology that combines logical reasoning with tool-based actions, enabling

iterative problem-solving for more accurate and complete responses. The ReAct process involves: (1) receiving a user query, (2) reasoning to determine necessary information, (3) acting using external tools to search, retrieve, and process data, (4) iterating reasoning and actions if the initial attempt is insufficient, and (5) generating a final, precise response.

Figure 3 illustrate this workflow, which integrates reasoning traces and actions to refine the internal context while responding adaptively to external observations. ReAct RAG extends traditional RAG systems by embedding a ReAct agent within the retrieval-augmented generation loop, enhancing response accuracy through stepwise reasoning and interaction with multiple documents and tools, thereby producing more precise and detailed outputs than single-step retrieval methods.

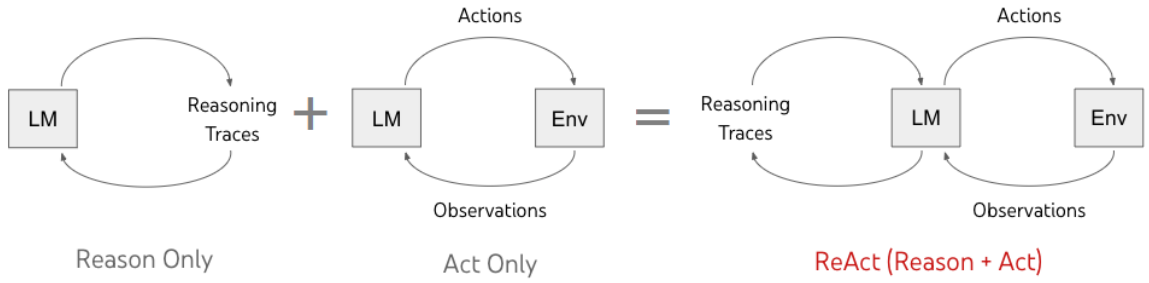

Figure 3. Figure of React flow from the original source [**?**].

2. Materials and Methods

As mentioned in the Introduction section 1, all the data handled in this work are inherent to the tourist information of the Municipality of Valencia (Spain). Consequently, the optimization of the algorithm is based on the relevant data collected about this city.

The entire project has been developed taking advantage of the open source libraries and frameworks available today. To create a custom application driven by a language model, we leverage LangChain [37]. It is an open source software library that allows developers to easily use other data sources and interact with other applications. Each of the libraries used is described in the section corresponding to its use.

2.1. Data collection

In this chapter we describe the data collection process and the sources used to collect information for the development of our tool.

If we want to offer our tourists useful, reliable and up-to-date advice, the first step is to carry out a detailed search for information material on the city's official channels. The most important of them is Visit València, a non-profit foundation in which the Valencia City Council, the Chamber of Commerce, Feria Valencia, the Valencian Business Confederation, Turismo Comunitat Valenciana and the Tourist Board of the Provincial Council participate, as well as most local tourism companies. Its objective is the strategic management and promotion of the city of Valencia in the tourism sector, with a professional approach that combines public and private interest.

The official Visit València portal [38] provides access to downloadable resources in Portable Document Format (PDF), including five different official tourist guides, each targeting a different type of traveler. In addition to the general city guide, more specific guides are available, such as "Guide to Valencia in three days" for short itineraries, "Valencia with your family" for trips with children, or guides focused on romantic getways, sports activities, and nightlife. Leveraging these authoritative documents allows us to incorporate updated and verified content into our tool, enhancing the credibility of the information provided to users.

In parallel, we employed web scraping techniques to extract data from official and reputable sources, including Wikipedia articles on historical references to the city, selected blogs, and event agendas. All data collection was performed after obtaining the necessary permissions and in strict

compliance with ethical and legal standards. In particular, we adhered to the corresponding robots.txt protocols and the Terms of Service of all indexed websites, ensuring that data extraction was both legal and non-intrusive. By combining heterogeneous data from these authorized sources, we built a comprehensive dataset comprising 84 documents covering attractions, monuments, cultural heritage, gastronomy, shopping, nature, sports activities, tourist recommendations, practical Valencia travel tips, and upcoming events. This approach aligns with our main objective of unifying diverse, reliable, and officially sanctioned sources to provide tourists with accurate and trustworthy guidance when planning or exploring Valencia's vibrant cultural landscape.

Finally, the knowledge base is composed entirely of texts in Spanish, ensuring maximal relevance from local sources and standardization for the integration phase. Each document is stored separately according to its origin, facilitating traceable and reliable retrieval. The dataset comprises a total of 54,400 tokens (approximately 100 pages), representing a pragmatic compromise for the CitCom.ai prototyping environment. It encompasses 443 dining establishments and restaurant chains, 715 distinct tourist attractions, and more than 10 event categories curated by the Events Agent. Coverage is geographically balanced, with a strong concentration in key historical districts such as Ciutat Vella, Eixample and Russafa, while also including peripheral areas such as El Saler and El Palmar within the Albufera Natural Park. This distribution reflects a deliberate and focused scope for the initial evaluation phase.

As noted above, RAG combines the power of an LLM with external data. If the dataset contains conflicting or redundant information, retrieval may struggle to provide the correct context, potentially leading to suboptimal generation by the LLM. To mitigate this, rigorous data integration and cleaning procedures were applied across all sources to harmonize content and eliminate redundancies or inconsistencies, ensuring a reliable and high-quality knowledge base for downstream processing.

2.2. Preprocessing

The pipeline continues through the cleaning process to remove noise, encoding errors, and duplicates. Text is normalized via case folding, punctuation standardization, and number handling, then tokenized using subword methods (e.g., BPE, WordPiece). Optional linguistic steps, such as lemmatization or stop-word removal, may further refine context.

The first step was to convert all documents from heterogeneous sources (e.g., databases, PDFs) to plain text (TXT). Paragraphs are identified by carriage returns to better organize file contents. All text is human-readable and represented as a sequence of characters.

Preprocessing applied to guides differed slightly from that applied to scraped text. Titles, subtitles, indexes, page numbers, legends, hyperlinks, and paragraph summaries next to illustrations were removed, retaining only the main body of the text. Each paragraph is separated by a carriage return, and blank lines were removed. For extracted text, additional normalization was performed: for instance, typical numbers and words in parentheses were removed from Wikipedia articles, and tables and references were excluded, leaving only pure text.

To better integrate information from all sources, Coreference Resolution (CR) [39] was applied. CR identifies all linguistic expressions (mentions) referring to the same real-world entity, replacing pronouns with noun phrases to avoid ambiguities that could lead to hallucinations in the RAG system.

Given the lack of open-source coreference resolution tool capable of processing Spanish efficiently, this task was delegated to GPT-4o. The following prompt was used to perform CR on tourism-related textual data prior to their use in the RAG system:

You are a highly capable NLP assistant specializing in coreference resolution. Given documents containing information about Valencia's tourist attractions, restaurants, events, and cultural heritage, identify all expressions that refer to the same entity (including pronouns, definite descriptions, and repeated mentions) and rewrite the text so that each entity is consistently referenced with a single canonical form. Preserve the original meaning, context, and factual details, including locations, events, and services. Do not introduce new information or modify existing facts.

This approach significantly streamlined the preprocessing workflow: GPT-4o was able to generate reformulated version of the original texts (Figure 4) within seconds. Although several open-source CR tools exist, most are either designed for English or show limited performance on heterogeneous Spanish corpora, such as tourism documents with mixed registers. GPT-4o was chosen for its high-quality, context-aware resolution across diverse sources, minimizing errors that could propagate to the RAG system. While using a closed model introduces additional computational cost, the substantial gains in speed, reliability, and preprocessing consistency justified this trade-off for the prototyping phase.

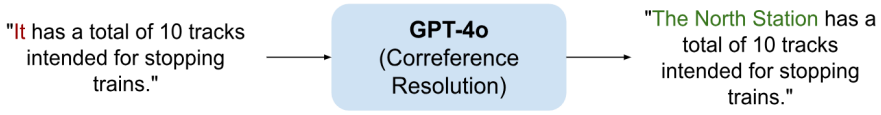

Figure 4. Example of asking GPT-4o to apply coreference resolution.

Repeating this procedure across all documents enhanced the consistency and quality of the text fragments, thereby improving retrieval performance. As a final step, a rigorous human review was conducted to resolve discrepancies and validate content accuracy, ensuring reliable data for the LLM and increasing confidence in generated responses.

2.3. Embeddings and Storage

Documents were first split into semantically coherent chunks, with overlap to preserve contextual continuity, enabling retrieval of specific tourist information such as attractions, restaurants, or events. Each chunk was converted into dense vector representations using pretrained encoders and indexed in ChromaDB along with metadata (e.g., source, timestamps) for efficient semantic search. Query preprocessing mirrored the corpus normalization, ensuring that user queries could retrieve relevant fragments accurately. Validation included both chunk coherence checks and retrieval accuracy tests, guaranteeing robustness for downstream RAG-based tourism recommendations.

We employed INSTRUCTOR [40] to generate 1024-dimensional embeddings for each text fragment, guided by task and domain instructions. Unlike traditional encoders, INSTRUCTOR produces flexible embeddings suitable for diverse tourism content without additional training.

To store the documents as dense vector embeddings, we utilized the open source ChromaDB [41] embedding database, which supports nuanced semantic retrieval and facilitates fast, context-aware integration with the RAG pipeline for tourims applications.

2.4. Agent Architecture

To create a system resistant to hallucinations and capable of handling complex user queries, we implemented an agent-based architecture (Figure 5) using the LangChain library. LangChain is a widely used framework that enables the construction of modular, scalable NLP pipelines through the orchestration of LLMs and external tools. It also provides seamless integration with a variety of APIs for retrieving external information. In our implementation, LangChain was used to connect the agent with the Events database and OpenStreetMap, allowing the system to access temporal and geospatial data.

For example, a ReAct agent may receive a query such as "*What are the best beaches in Valencia?*" The agent first reasons about which tools or strategies to use (Re), then invokes the appropriate retrieval tools to gather relevant information (Act), and finally generates a detailed, contextually accurate response based on the retrieved data.

For the language model, we used Mistral Small 3.1 [42], a state-of-the-art open-source LLM that powers the reasoning and generation capabilities of the agent. The model comprises approximately 24 billion parameters and is distributed under the Apache 2.0 license, which facilitates its integration into research and experimental environments. It is publicly available through the huggingface model hub

under the tag *mistralai/Mistral-Small-3.1-24B-Base-2503*, ensuring transparent access and reproducibility for the research community.

In its latest release, Mistral Small 3.1 introduces an extended context window of up to 128.000 tokens, enabling the model to process long text sequences—such as full documents or concatenated knowledge sources—while maintaining reasoning coherence. Furthermore, this version includes multimodal capabilities (text + image), broadening its applicability to tasks that require the joint interpretation of textual and visual information. From a methodological standpoint, and in alignment with the retrieval-augmented generation (RAG) architecture adopted in this study, the combination of a large context window and strong reasoning abilities is particularly advantageous. It allows the agent to integrate retrieved knowledge chunks, build coherent reasoning chains, and generate contextualized outputs that effectively combine external information with the model's own inference capabilities.

Additionally, the European origin of Mistral strengthens its relevance to this investigation. Since the present study was conducted within the framework of a European research project, the use of a model developed in Europe aligns with the project's objectives of promoting technological sovereignty and supporting the regional AI ecosystem.

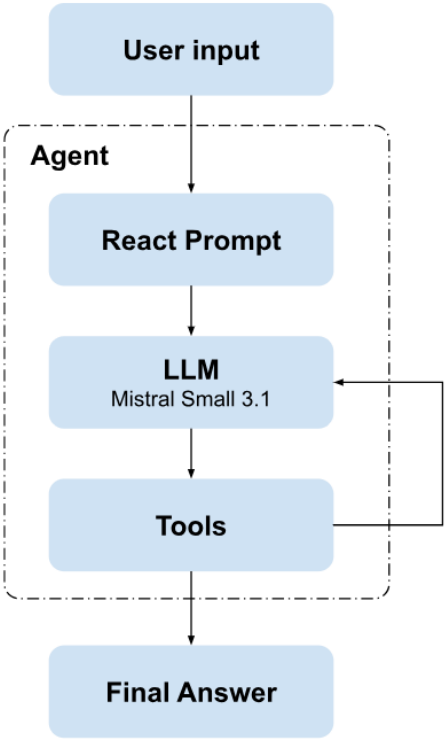

Figure 5. The entire Agent-based architecture integrating LLM and tools for iterative query resolution.

2.5. Tools

In the context of a ReAct agent, a *tool* refers to an external capability or function that the agent invokes during its reasoning and acting cycle. While the language model provides reasoning abilities, tools extend its functionality by enabling access to structured data sources, APIs, or specialized operations such as retrieval, geospatial queries, or event management. In this way, tools act as bridges between the agent's abstract reasoning and concrete actions in the real world, ensuring that responses are grounded, accurate, and contextually enriched.

In our system, we implemented three specialised tools to manage different types of queries and enhance agent responses:

1. *Retrieval Tool*, obtains relevant context from the document knowledge base to support accurate and informative answers;

2. *Event Tool*, retrieves current events relevant to the user's query, ensuring temporal awareness and up-to-date recommendations;

3. *Geospatial Tool*, filters and ranks results based on geographic location, enabling personalized, location-aware suggestions for tourists.

2.5.1. Retrieval Tool

Our main agent tool implements a RAG system to obtain information from the documents described above (Figure 6). The system consists of several key components:

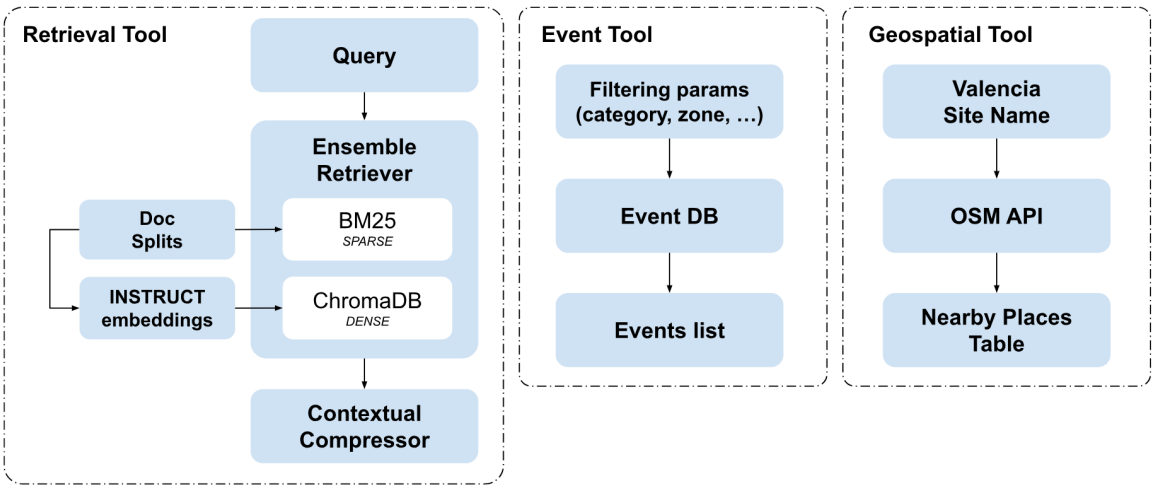

Figure 6. Modular toolset for retrieval, event searching, and geospatial data enrichment.

Vector DataBase: Documents are stored as dense vector *embeddings* in the open-source ChromaDB [43], which supports semantic retrieval and efficient access to relevant content.

Model embedding: We employed INSTRUCTOR XL [40] to generate *embeddings* for both documents and query instructions. Each chunk is paired with task and domain-specific guidelines to clarify its intended application. Unlike earlier specialized encoders, INSTRUCTOR XL produces versatile *embeddings* across multiple tasks and domains without additional *fine-tuning*, while remaining parameter-efficient.

BM25: To capture exact keyword matches and term-based relevance, we apply the Best Match 25 (BM25) ranking algorithm [44,45] to all document fragments. This highlights named entities and ensures precise retrieval for queries relying on specific terms.

Ensemble Retriever: The Ensemble Retriever [46,47] is a hybrid information retrieval approach that combines the BM25 sparse retriever, which excels at exact keyword matching, with a dense retriever that captures semantic nuances and contextual relationships. This hybrid strategy leverages the strengths of both techniques, improving overall retrieval performance.

The dense retriever operates in the embedding space, grouping semantically similar documents to capture contextually relevant information. To efficiently manage tourist guides, we employed a Contextual Compression Retriever. Often, the most relevant information is embedded within lengthy documents containing a lot of irrelevant text. Passing entire document to the LLM increases computational cost and can reduce response quality. Contextual compression mitigates this by filtering documents based on the query context, returning only the most pertinent chunk. Specifically, we implemented a pipeline applied to all ChromaDB embeddings, which compresses each document and retains only the most relevant fragments using a similarity threshold filter.

By combining these two complementary retrieval mechanisms, the hybrid search achieves a more holistic understanding of relevance, improving both the accuracy and robustness of the retrieval process. Although the Ensemble Retriever introduces additional parameters and hyperparameters that require careful tuning (Table 2), its advantages in handling both exact keyword matches and semantic nuances make it a highly effective approach for optimizing RAG performance.

2.5.2. Geospatial Tool

The Geospatial Tool is based on OpenStreetMap (OSM) [48] and was developed to enrich the RAG model with up-to-date and detailed geographic information. OSM is a collaborative and continuously updated mapping platform that provides comprehensive data on locations, routes, points of interest, and geographic features worldwide. Integrating OSM enables the model to access accurate and current spatial data, such as location, directions, and topological features, which is essential for tourism-oriented applications.

Furthermore, within the ReAct architecture, the agent can not only retrieve specific data from OSM but also reason about it both before and after each query. This allows spatial information to be incorporated contextually and adaptively, which is critical for complex tasks such as generating personalized routes, describing environments in detail, or answering questions about accessibility, distances, and travel logistics.

2.5.3. Event Tool

To address tourists' demand for discovering ongoing and upcoming activities within the city, an event tool was developed. The system integrates information from the official Visit València event calendar, which is regularly updated and publicly available. The collected data are stored in a structured databae that allows the model to execute parameterized queries based on attributes such as geographic location, event timing, and category, among others. This approach ensures that the model maintains access to reliable, up-to-date, and contextually relevant information, accomodating the dynamic and time-sensitive nature of urban events.

2.6. Evaluation with RAGAS

The evaluation of Retrieval-Augmented Generation (RAG) systems has recently attracted increased attention, leading to the development of several benchmarking frameworks that assess different aspects of retrieval and generation performance. For instance, the BEIR benchmark [49] provides a standardized suite for evaluating retrieval methods across a wide range of domains and datasets, while the RGB benchmark [50] extends this paradigm by jointly assessing retrieval, grounding, and generation quality in RAG-based large language models. These frameworks offer valuable foundations for comparative evaluation and reproducibility in the field. In this study, we employ the RAGAS (Retrieval-Augmented Generation Assessment) framework [9] due to its flexibility, reference-free design, and widespread adoption as the de facto standard for evaluating RAG systems in contemporary research [51]. RAGAS allows efficient, data-driven assessment of both the retriever and generator components, evaluating them independently as well as within an integrated pipeline. This component-wise analysis provides for a granular understanding of system performance. Furthermore, RAGAS introduces several key metrics for holistic evaluation, including *faithfulness*, which measures the factual accuracy of the generated answer against the retrieved context; *answer relevancy*, which quantifies how well the output answer addresses the user's query; and *context precision* and *context recall*, which assess the quality and completeness of the retrieved documents.

3. Results

We developed a Retrieval-Augmented Generation (RAG) agent based on the Mistral Small 3.1 language model, equipped with the three tools described in Section 2.5. The central retrieval module employs a vector store containing approximately 54,400 tokens, segmented into 220 chunks, and embedded in a 1024-dimensional space to enable efficient semantic search. Complementary geospatial and event tools provide continuously updated geographic and structured cultural data, while the ReAct-based reasoning loop integrates retrieval and generation to ensure answer relevancy.

3.1. Assessment

Evaluating RAG architectures is inherently challenging, as it requires assessing multiple aspects simultaneously: the retrieval system's ability to locate relevant and focused contextual passages, the LLM's capacity to effectively leverage this information, and the overall quality of the generated responses. Following the approach of previous studies [52], we constructed a *ground truth* (GT) dataset to systematically evaluate our RAG system. Using the Google Gemini API, we automatically generated 994 question–answer pairs based on 84 documents containing official tourism information about Valencia using the *gemini-2.5-flash* model. The dataset covers diverse topics such as trip planning, historical and cultural references, traditions, monuments, accommodations, gastronomy, and nightlife. It also includes multi-hop questions that require integrating information from multiple sources. All generated answers were subsequently reviewed and validated by two human experts to ensure factual accuracy, coherence, and consistency. Both datasets are publicly available in the Zenodo repository at https://zenodo.org/records/17384690. Although the RAGAS framework enables evaluation without requiring manually annotated ground truth data, the inclusion of this expert-validated GT dataset enhances the robustness of our assessment and provides an independent benchmark for cross-validating and interpreting the automated RAGAS metrics. Figure 7 presents several representative examples from the question dataset.

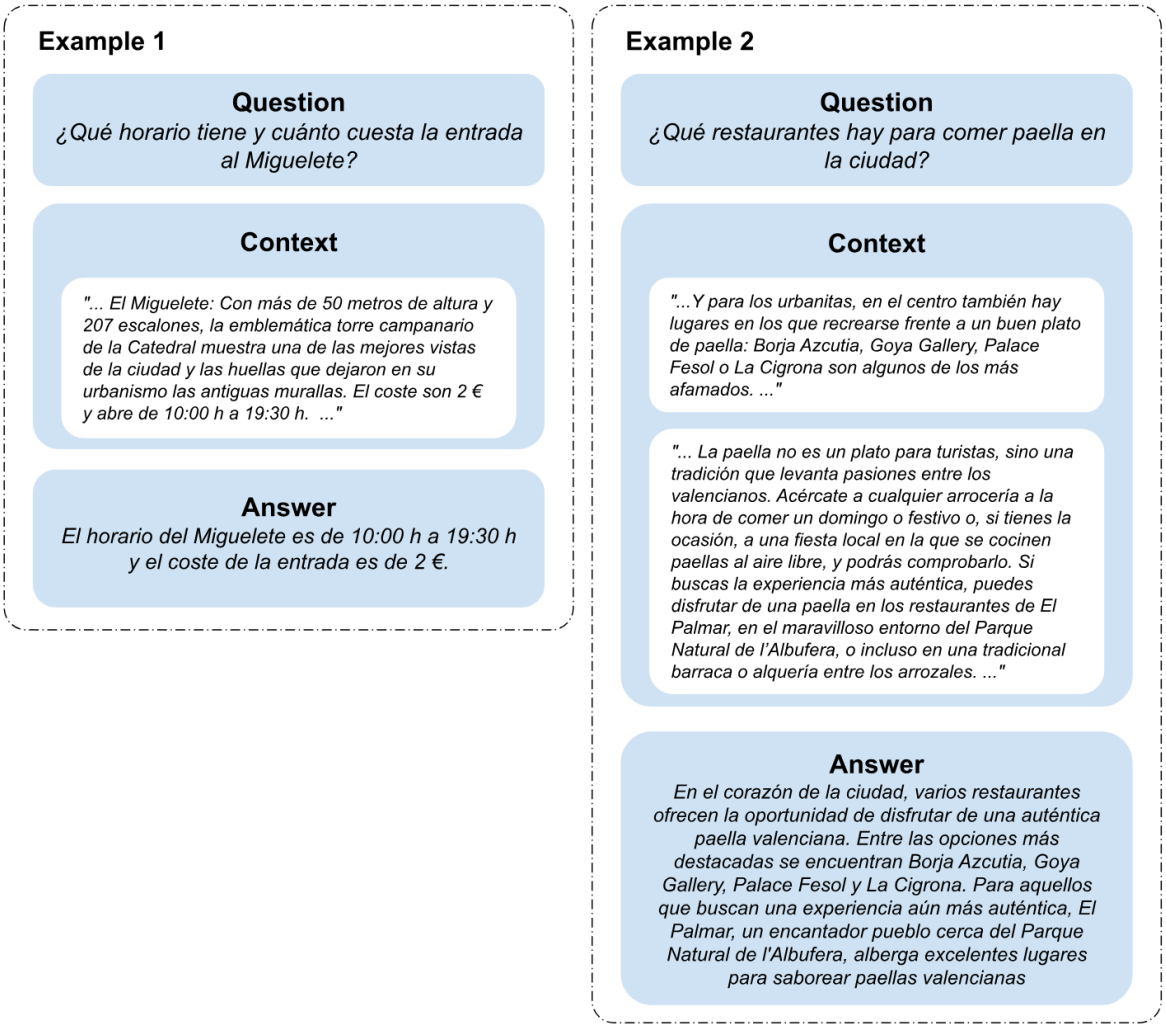

Figure 7. Two instruction examples from test reference dataset.

The following section reports the quantitative metrics obtained when evaluating the retriever with the RAGAS framework, using GPT-4o as the judge model for response assessment.

3.2. Retrieval evaluation

The retrieval component was evaluated in terms of its ability to return relevant contextual passages that directly support the GT answers. As shown in Table 1, the retriever achieved a Context Precision Mean of 0.714 and a Context Recall Mean of 0.563. These values indicate that, on average, more than 70% of the top-K retrieved passages are relevant to the query, while the system is able to cover slightly more than half of the reference claims with the retrieved context. For this evaluation, the parameter K was set to 6, meaning that the first six retrieved passages were considered. These results suggest that the retriever effectively prioritizes relevant documents but still leaves room for improvement in terms of recall, i.e., ensuring that all relevant information is consistently included in the retrieved set.

Table 1. Metrics obtained from the evaluation of the Context Retriever using the RAGAS framework.

Retriever	Context Precision Mean	Context Recall Mean
Simple (Semantic)	0.734	0.545
Ensemble (Semantic + BM25)	0.732	0.563
Ensemble + Compression	0.727	0.545

The metrics reported in Table 1 were computed for each query in the test dataset and then averaged across all samples.

- **Context Precision@K**: Quantifies the weighted precision of the top-K retrieved items, accounting for their relevance:

$$\frac{\sum_{k=1}^{K}(\text{Precision@k} \times v_k)}{\text{Total number of relevant items in the top } K \text{ results}}$$

  where $v_k \in \{0, 1\}$ indicates whether the item at rank k is relevant.
- **Precision@k**: Measures the proportion of true positives among the top-k retrieved items.

$$\frac{\text{true positives@k}}{\text{true positives@k} + \text{false positives@k}}$$

- **Context Recall**: Evaluates the coverage of relevant claims in the reference that are supported by the retrieved context.

$$\frac{\text{Number of claims in the reference supported by the retrieved context}}{\text{Total number of claims in the reference}}$$

3.3. Response evaluation

The goal of this step is to evaluate the response generated by the LLM to ensure that the information returned is accurate, contextually faithful, and appropriate to the user's query. Using the ground truth question dataset described above, each query was passed to the RAG-based chatbot to generate answers, which were then compared with the GT answers. The comparison process was automated using the RAGAS evaluation framework, enabling a systematic assessment of response quality and facilitating fine-tuning of system parameters and hyperparameters. The specific configuration of hyperparameters used in this study is summarized in Table 2.

Table 2. Hyperparameter Configuration.

Component	Parameter	Value
Embedding	Model	INSTRUCTOR-XL
	Embedding size	1024
Splitter	Chunk size	1000 chars
	Overlap	200 chars
Hybrid Ensemble	BM25 weight	0.4
	BM25 k	3
	ChromaDB weight	0.6
	ChromaDB k	3
Compressor	Similarity threshold	0.6
LLM	Model	Mistral Small 3.1
	Temperature	0

This evaluation process allowed us to detect and analyze hallucinations and factual inaccuracies at the level of individual model responses. For our main experiments, we selected Mistral Small 3.1 [42] as the primary model and conducted a detailed analysis of its outputs. To contextualize its performance, we also evaluated several alternative setups, including architectural variations and a commercial model (GPT-4o Mini [53]). The architectural variants included: (1) a baseline without retrieval (No RAG), and (2) a standard RAG implementation without the ReAct mechanism.

Table 3 summarizes the results in terms of faithfulness and answer relevancy, computed using the RAGAS evaluation framework. For each query in the test set, these metrics were calculated and then averaged. The results illustrate the comparative performance of the models and configurations, confirming the effectiveness of Mistral Small 3.1 —particularly in its RAG+ReAct architecture- as the core model in our system.

Table 3. Metrics obtained from the evaluation of answer generation using the RAGAS framework.

Model Evaluated	**Faithfulness**		**Answer relevancy**	
	mean	std	mean	std
Mistral Small 3.1 (No RAG)	—		0.643	0.426
Mistral Small 3.1 (RAG)	0.918	0.160	0.831	0.297
Mistral Small 3.1 (RAG+ReAct)	0.858	0.246	0.897	0.194
GPT-4o Mini (RAG+ReAct)	**0.922**	0.159	**0.899**	0.186

To determine the statistical significance of these differences, we performed an ANOVA analysis on the metric scores. The results indicate significant variation across models for both faithfulness ($F = 27.97$, $p < 0.001$) and answer relevancy ($F = 137.58$, $p < 0.001$). A Tukey HSD post-hoc analysis further reveals that for faithfulness, the RAG+ReAct configuration with Mistral Small 3.1 underperforms compared to both RAG and GPT-4o Mini, with the latter two being statistically indistinguishable. In terms of answer relevancy, GPT-4o Mini significantly outperforms both No RAG and RAG, but shows no significant difference compared to RAG+ReAct with Mistral Small 3.1. These findings suggest that while the ReAct mechanism may slightly reduce factual accuracy in Mistral Small 3.1, it enhances answer relevancy, and GPT-4o Mini delivers consistently strong performance across both evaluation dimensions.

- **Faithfulness Score**: Evaluates the degree to which the claims made in a generated response are substantiated by the retrieved context.

$$\frac{\text{Number of claims supported by the retrieved context}}{\text{Total number of claims in the response}}$$

- **Answer Relevancy**: Represents the average semantic similarity between the embeddings of the generated answers and those of the reference (ground truth) answer, computed via cosine similarity.

$$\frac{1}{N}\sum_{i=1}^{N}\cos(\mathbf{E}_{g_i}, \mathbf{E}_o)$$

4. Discussion

The present study demonstrates the practical implementation of an agentic RAG system for urban tourism information. The analysis focuses on the impact of dataset design, preprocessing strategies, and model selection on retrieval and generation performance, providing insights into best practices for developing scalable, context-aware AI assistants in real-world scenarios.

The first aspect to consider is the dataset. As discussed in Section 2, its size reflects a deliberate compromise: although relatively compact, the dataset was carefully curated to maximize coverage of essential tourism information while remaining manageable for thorough manual verification. This choice was intentional to maintain feasibility and data quality during prototyping, ensuring that the dataset prioritizes high-quality, reliable, and contextually relevant information for the RAG system. The primary goal at this stage was to validate the technical viability and operational coherence of the agentic RAG system in a controlled setting. Once all parameters and hyperparameters are optimized for the retrieval and generation, the system can integrate additional sources, such as detailed accommodation offerings, further enriching the knowledge base.

Building upon the curated dataset, preprocessing procedures, including PDF document handling and integration of data from official tourism sources, must be adapted to the structure and format of each information source. Incorporating a diverse set of content helped expand coverage beyond central areas. Notably, the removal of titles and subtitles from each section enhanced retrieval performance by reducing redundancy, enabling the hybrid search to focus solely on relevant chunks. This improvement in context quality simultaneously increases the accuracy, relevance, and factual integrity of responses generated by the language model. Coreference resolution (CR) was another critical step for mitigating hallucinations. In tourism-related texts, frequent use of pronouns and ambiguous references often results in text chunks lacking clear context, which can lead the LLM to incorrectly associate facts with the wrong landmark. By systematically replacing these references with the specific entities they denote, each chunk became a self-contained, factually precise unit, further improving retrieval accuracy and ensuring reliable outputs.

We evaluated different configurations of the Mistral Small 3.1 language model to assess their impact on RAG system performance. Table 3 summarizes the results, highlighting notable differences in answer relevancy and faithfulness across configurations:

- Without RAG, the model exhibits low answer relevancy (0.643), reflecting its limited ability to retrieve relevant information without external context.

- With simple RAG, the system achieves high faithfulness (0.918) and good relevancy (0.831), demonstrating that retrieved chunks effectively anchor generation and reduce hallucinations.

- With ReAct + RAG, answer relevancy improves slightly (0.897) while faithfulness decreases somewhat (0.858). This configuration produces more direct and contextually aligned answers; however, it introduces a small amount of content not fully supported by the retrieved sources.

These observations are consistent with prior work on reasoning architectures such as chain-of-thought (CoT) and Reason and Act (ReAct) [54], which can enhance fluency and multi-hop reasoning but may complicate the anchoring of outputs to retrieved evidence. From our perspective, the ReAct + RAG configuration represents the most suitable choice for a tourism assistant: its higher *Answer Relevancy* ensures responses are useful and contextually aligned, while the slight decrease in faithfulness remains acceptable for providing coherent, actionable guidance.

Beyond these technical results, deploying agentic RAG systems in urban tourism raises important operational and ethical considerations. Maintaining up-to-date information requires continuous

dataset updates, particularly for dynamic events and seasonal activities, to ensure reliability. The system's recommendations may influence tourist flows and local businesses, highlighting the importance of balancing efficiency with sustainable urban tourism practices. Ethical aspects, including potential biases in recommendations, transparency of AI-generated guidance, and equitable treatment of underrepresented groups, must be addressed to ensure trustworthy user experiences. It's important to hightlight that the system relies on official tourism sources, which pay particular attention to inclusivity, helping to minimize biases against underrepresented populations.

5. Conclusions

In this work we present an AI-based conversational system that allows users to query travel information using natural language, while the intelligent agent generates personalized travel plans respecting user-specific constraints. By integrating multiple sources of information into a single interactive interface, the system reduces the need to consult multiple platforms, streamlines trip planning, and enhances overall traveler experience.

The primary contribution of this study is the development of an agentic architecture integrated within a RAG framework, which orchestrates specialized tools for text, geospatial, and event retrieval. This design enables cohesive semantic, spatial, and temporal reasoning and demonstrates the potential of combining retrieval-augmented generation techniques with task-specific agents to address complex information needs in the tourism domain.

Our evaluation highlights several key insights. The use of ReAct + RAG with Mistral Small 3.1 provides a strong balance between Answer Relevancy and Faithfulness: while the reasoning-action loop slightly reduces strict adherence to retrieved context, it significantly improves the relevance and contextual alignment of responses, making the system more useful for real-world queries. In contrast, configurations without RAG or without ReAct exhibited lower relevance or required stronger grounding to maintain faithfulness, underscoring the importance of combining retrieval and reasoning mechanisms in agentic systems.

The modular and scalable design of the system facilitates the integration of additional types of information, such as local events, seasonal activities, and emerging points of interest, supporting flexible deployment in other cities or domains. Furthermore, the approach provides a foundation for iterative improvements, including dataset expansion, enhanced preprocessing, multilingual support, and the incorporation of smaller, efficient language models with targeted hallucination control to optimize computational efficiency without compromising response quality. Future work will also involve validation with professional tour guides and end-users, allowing us to evaluate the system's practical effectiveness in real-world tourism scenarios and guide further refinements.

Overall, this study demonstrates that intelligent agent-driven travel assistants, powered by retrieval-augmented generation, ReAct reasoning, and specialized tools, can provide context-aware, practical guidance in urban tourism, bridging the gap between AI research and real-world user experiences.

Author Contributions: J.V. and E.S.; methodology, J.V.; software, A.S. and A.B.; validation, A.S. and E.F.; formal analysis, X.B.; investigation, A.B. and A.S.; resources, X.B.; data curation, A.B.; writing—original draft preparation, A.B. and A.S.; writing—review and editing, X.B.; visualization, A.S.; supervision, J.V.; project administration, R.M. and E.S.; funding acquisition, R.M. All authors have read and agreed to the published version of the manuscript."

Funding: This work was conducted as part of the CitCom.ai project, funded by the European Union's Digital Europe program under the Grant Agreement No. 101100728. The funding body had no role in the design of the study, data collection and analysis, or the preparation of the manuscript. This research also was funded by MCIN/AEI/10.13039/501100011033 "ERDF A way of making Europe" through grant number PID2021-127946OB-I00.

Data Availability Statement: The datasets generated and analyzed during the current study are publicly available in the Zenodo repository at https://zenodo.org/records/17384690. The released resources include: - Tourism dataset: a comprehensive collection of attractions, restaurants, cultural heritage sites, event categories, and related information for the city of Valencia, curated from official and reputable sources. - Ground truth (question-answer pairs): a set of manually created queries and corresponding reference answers designed to evaluate and

benchmark the retrieval and generation capabilities of the proposed RAG-based system. All datasets are openly accessible under a Creative Commons Attribution 4.0 International (CC BY 4.0) license, which allows others to share, copy, redistribute, adapt, and build upon the datasets for any purpose, including commercial use, provided that appropriate credit is given to the original authors. These resources enable reproducibility and benchmarking of retrieval-augmented generation systems in tourism applications.

Acknowledgments: This work is part of the project: CitCom.ai TEF - European artificial intelligence and robotics testing and experimentation facility for smart and sustainable cities and communities (Project number: 101100728). CitCom.ai's contribution is to build a lasting facility in support of the EU's global position in cities and communities. By expanding existing infrastructure and expertise across the Union, CitCom.ai will provide real-world conditions for Test and Experimentation Facilities (TEFs), relevant for AI and robotics solutions in cities and communities, to accelerate the transition towards a Greener and more digital Europe. The University of Valencia, as a member of the project, is the entity in charge of tourism management, and all AI solutions developed during the project will be tested in the city of Valencia, which will act as a real world.

Conflicts of Interest: The authors declare that they have no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

BM25	Best Match 25
CR	Coreference Resolution
GPT	Generative Pre-trained Transformer
GT	Ground Truth
LLM	Large Language Model
OSM	OpenStreetMap
Q&A	Question and Answer
RAG	Retrieval-Augmented Generation
RAGAS	Retrieval Augmented Generation Assessment
ReAct	Reasoning and Acting
SQL	Structured Query Language
TEF	Testing and Experimentation Facility
CoT	Chain of Thought

References

1. Grundner, L.; Neuhofer, B. The bright and dark sides of artificial intelligence: A futures perspective on tourist destination experiences. *Journal of Destination Marketing & Management* **2021**, *19*, 100511.
2. Gretzel, U. Intelligent systems in tourism: A social science perspective. *Annals of tourism research* **2011**, *38*, 757–779.
3. Gretzel, U.; Hwang, Y.H.; Fesenmaier, D.R. Informing destination recommender systems design and evaluation through quantitative research. *International Journal of Culture, Tourism and Hospitality Research* **2012**, *6*, 297–315.
4. Li, Z.; Grossman, M.; Eric.; Qasemi.; Kulkarni, M.; Chen, M.; Chiang, Y.Y. MapQA: Open-domain Geospatial Question Answering on Map Data, 2025, [arXiv:cs.CL/2503.07871].
5. Oche, A.J.; Folashade, A.G.; Ghosal, T.; Biswas, A. A Systematic Review of Key Retrieval-Augmented Generation (RAG) Systems: Progress, Gaps, and Future Directions, 2025, [arXiv:cs.CL/2507.18910].
6. Wang, J.; Fu, J.; Wang, R.; Song, L.; Bian, J. PIKE-RAG: sPecIalized KnowledgE and Rationale Augmented Generation, 2025, [arXiv:cs.CL/2501.11551].
7. LangChain. LangChain Framework: Building Applications with Large Language Models. https://www.langchain.com/. Accessed: 2025-06-22.
8. CITCOMTEF. CITCOMTEF — Centro de Investigación y Transferencia en Computación y Tecnologías Educativas. https://citcomtef.eu/. Accessed: 2025-10-02.
9. Es, S.; James, J.; Espinosa-Anke, L.; Schockaert, S. Ragas: Automated Evaluation of Retrieval Augmented Generation, 2025, [arXiv:cs.CL/2309.15217].

10. Jelinek, F. *Statistical Methods for Speech Recognition*; Language, Speech, & Communication: A Bradford Book, The MIT Press, 1998.
11. Stolcke, A. Srilm — An Extensible Language Modeling Toolkit. *Proceedings of the 7th International Conference on Spoken Language Processing (ICSLP 2002)* **2004**, *2*.
12. Rosenfeld, R. Rosenfeld, R.: Two decades of statistical language modeling: where do we go from here? Proceedings of the IEEE 88(8), 1270-1278. *Proceedings of the IEEE* **2000**, *88*, 1270 – 1278. https://doi.org/10.1109/5.880083.
13. Operationnelle, D.; Bengio, Y.; Ducharme, R.; Vincent, P.; Mathematiques, C. A Neural Probabilistic Language Model **2001**.
14. Mikolov, T.; Karafiát, M.; Burget, L.; Cernocký, J.; Khudanpur, S. Recurrent neural network based language model. 09 2010, Vol. 2, pp. 1045–1048. https://doi.org/10.21437/Interspeech.2010-343.
15. Kombrink, S.; Mikolov, T.; Karafiát, M.; Burget, L. Recurrent Neural Network Based Language Modeling in Meeting Recognition. 08 2011, pp. 2877–2880. https://doi.org/10.21437/Interspeech.2011-720.
16. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need, 2023, [arXiv:cs.CL/1706.03762].
17. Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language Models are Unsupervised Multitask Learners. 2019.
18. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, 2019, [arXiv:cs.CL/1810.04805].
19. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners, 2020, [arXiv:cs.CL/2005.14165].
20. OpenAI.; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F.L.; Almeida, D.; Altenschmidt, J.; Altman, S.; et al. GPT-4 Technical Report, 2024, [arXiv:cs.CL/2303.08774].
21. Jiang, A.Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D.S.; de las Casas, D.; Hanna, E.B.; Bressand, F.; et al. Mixtral of Experts, 2024, [arXiv:cs.LG/2401.04088].
22. Khandelwal, U.; Levy, O.; Jurafsky, D.; Zettlemoyer, L.; Lewis, M. Generalization through Memorization: Nearest Neighbor Language Models, 2020, [arXiv:cs.CL/1911.00172].
23. Li, B.; Nishikawa, Y.; Höllmer, P.; Carillo, L.; Maggs, A.C.; Krauth, W. Hard-disk pressure computations—a historic perspective. *The Journal of Chemical Physics* **2022**, *157*. https://doi.org/10.1063/5.0126437.
24. Ram, O.; Levine, Y.; Dalmedigos, I.; Muhlgay, D.; Shashua, A.; Leyton-Brown, K.; Shoham, Y. In-Context Retrieval-Augmented Language Models, 2023, [arXiv:cs.CL/2302.00083].
25. Kang, M.; Kwak, J.M.; Baek, J.; Hwang, S.J. Knowledge Graph-Augmented Language Models for Knowledge-Grounded Dialogue Generation, 2023, [arXiv:cs.CL/2305.18846].
26. Wang, Y.; Kordi, Y.; Mishra, S.; Liu, A.; Smith, N.A.; Khashabi, D.; Hajishirzi, H. Self-Instruct: Aligning Language Models with Self-Generated Instructions. In Proceedings of the Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers); Rogers, A.; Boyd-Graber, J.; Okazaki, N., Eds., Toronto, Canada, jul 2023; pp. 13484–13508. https://doi.org/10.18653/v1/2023.acl-long.754.
27. Shi, W.; Min, S.; Yasunaga, M.; Seo, M.; James, R.; Lewis, M.; Zettlemoyer, L.; tau Yih, W. REPLUG: Retrieval-Augmented Black-Box Language Models, 2023, [arXiv:cs.CL/2301.12652].
28. Jiang, Z.; Xu, F.F.; Gao, L.; Sun, Z.; Liu, Q.; Dwivedi-Yu, J.; Yang, Y.; Callan, J.; Neubig, G. Active Retrieval Augmented Generation, 2023, [arXiv:cs.CL/2305.06983].
29. Huang, L.; Yu, W.; Ma, W.; Zhong, W.; Feng, Z.; Wang, H.; Chen, Q.; Peng, W.; Feng, X.; Qin, B.; et al. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *ACM Transactions on Information Systems* **2025**, *43*, 1–55. https://doi.org/10.1145/3703155.
30. Izacard, G.; Lewis, P.; Lomeli, M.; Hosseini, L.; Petroni, F.; Schick, T.; Dwivedi-Yu, J.; Joulin, A.; Riedel, S.; Grave, E. Atlas: Few-shot Learning with Retrieval Augmented Language Models, 2022, [arXiv:cs.CL/2208.03299].
31. Khattab, O.; Santhanam, K.; Li, X.L.; Hall, D.; Liang, P.; Potts, C.; Zaharia, M. Demonstrate-Search-Predict: Composing retrieval and language models for knowledge-intensive NLP, 2023, [arXiv:cs.CL/2212.14024].
32. Hoffmann, J.; Borgeaud, S.; Mensch, A.; Buchatskaya, E.; Cai, T.; Rutherford, E.; de Las Casas, D.; Hendricks, L.A.; Welbl, J.; Clark, A.; et al. Training Compute-Optimal Large Language Models, 2022, [arXiv:cs.CL/2203.15556].
33. Wang, L.; Chen, H.; Yang, N.; Huang, X.; Dou, Z.; Wei, F. Chain-of-Retrieval Augmented Generation, 2025, [arXiv:cs.IR/2501.14342].

34. Schick, T.; Dwivedi-Yu, J.; Dessi, R.; Raileanu, R.; Lomeli, M.; Hambro, E.; Zettlemoyer, L.; Cancedda, N.; Scialom, T. Toolformer: Language Models Can Teach Themselves to Use Tools. In Proceedings of the Advances in Neural Information Processing Systems; Oh, A.; Naumann, T.; Globerson, A.; Saenko, K.; Hardt, M.; Levine, S., Eds. Curran Associates, Inc., 2023, Vol. 36, pp. 68539–68551.
35. Park, J.S.; O'Brien, J.; Cai, C.J.; Morris, M.R.; Liang, P.; Bernstein, M.S. Generative Agents: Interactive Simulacra of Human Behavior. In Proceedings of the Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology, New York, NY, USA, 2023; UIST '23. https://doi.org/10.1145/3586183.3606763.
36. Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; Cao, Y. ReAct: Synergizing Reasoning and Acting in Language Models, 2023, [arXiv:cs.CL/2210.03629].
37. Topsakal, O.; Akinci, T.C. Creating Large Language Model Applications Utilizing LangChain: A Primer on Developing LLM Apps Fast. *International Conference on Applied Engineering and Natural Sciences* **2023**, *1*, 1050–1056. https://doi.org/10.59287/icaens.1127.
38. Visit Valencia. Official Tourism Website of the City of Valencia, 2025. Accessed: 2025-03-15.
39. Ng, V. Supervised Noun Phrase Coreference Research: The First Fifteen Years. 01 2010, pp. 1396–1411.
40. Su, H.; Shi, W.; Kasai, J.; Wang, Y.; Hu, Y.; Ostendorf, M.; tau Yih, W.; Smith, N.A.; Zettlemoyer, L.; Yu, T. One Embedder, Any Task: Instruction-Finetuned Text Embeddings, 2023, [arXiv:cs.CL/2212.09741].
41. Try Chroma. Try Chroma — Vector Database for AI Applications. https://www.trychroma.com/. Accessed: 2025-10-22.
42. Mistral AI. Mistral Small 3.1: SOTA. Multimodal. Multilingual. Apache 2.0. https://mistral.ai/news/mistral-small-3-1, 2025. Accessed: 2025-10-22.
43. Chroma Core. Chroma: Open-source search and retrieval database for AI applications. https://github.com/chroma-core/chroma, 2025. Accessed: 2025-06-22.
44. Robertson, S.; Jones, K. Simple, Proven Approaches to Text Retrieval **1997**.
45. Robertson, S.; Zaragoza, H. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends in Information Retrieval* **2009**, *3*, 333–389. https://doi.org/10.1561/1500000019.
46. Kuzi, S.; Zhang, M.; Li, C.; Bendersky, M.; Najork, M. Leveraging Semantic and Lexical Matching to Improve the Recall of Document Retrieval Systems: A Hybrid Approach, 2020, [arXiv:cs.IR/2010.01195].
47. Hambarde, K.A.; Proença, H. Information Retrieval: Recent Advances and Beyond. *IEEE Access* **2023**, *11*, 76581–76604. https://doi.org/10.1109/access.2023.3295776.
48. OpenStreetMap contributors. OpenStreetMap. https://www.openstreetmap.org/, 2025. Accessed: 2025-07-11.
49. Thakur, N.; Reimers, N.; Rücklé, A.; Srivastava, A.; Gurevych, I. BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models, 2021, [arXiv:cs.IR/2104.08663].
50. Chen, J.; Lin, H.; Han, X.; Sun, L. Benchmarking Large Language Models in Retrieval-Augmented Generation, 2023, [arXiv:cs.CL/2309.01431].
51. Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Wang, M.; Wang, H. Retrieval-Augmented Generation for Large Language Models: A Survey, 2024, [arXiv:cs.CL/2312.10997].
52. Niu, C.; Wu, Y.; Zhu, J.; Xu, S.; Shum, K.; Zhong, R.; Song, J.; Zhang, T. RAGTruth: A Hallucination Corpus for Developing Trustworthy Retrieval-Augmented Language Models, 2024, [arXiv:cs.CL/2401.00396].
53. OpenAI.; :.; Hurst, A.; Lerer, A.; Goucher, A.P.; Perelman, A.; Ramesh, A.; Clark, A.; Ostrow, A.; Welihinda, A.; et al. GPT-4o System Card, 2024, [arXiv:cs.CL/2410.21276].
54. Yao, Z.; Liu, Y.; Chen, Y.; Chen, J.; Fang, J.; Hou, L.; Li, J.; Chua, T.S. Are Reasoning Models More Prone to Hallucination?, 2025, [arXiv:cs.CL/2505.23646]. Submitted on 29 May 2025.