Web Semantics: Science, Services and Agents on the World Wide Web 85 (2025) 100853

Contents lists available at ScienceDirect

Web Semantics: Science, Services and Agents on the World Wide Web

journal homepage: www.elsevier.com/locate/websem

LLM experimentation through knowledge graphs: Towards improved management, repeatability, and verification

John S. Erickson [a], Henrique Santos [a,*], Vládia Pinheiro [b,a], Jamie P. McCusker [a], Deborah L. McGuinness [a]

[a] Rensselaer Polytechnic Institute, Troy, 12180, NY, United States
[b] Universidade de Fortaleza, Fortaleza, 60811-905, CE, Brazil

ARTICLE INFO

Keywords:
Generative large language models
Knowledge graphs
Retrieval-Augmented Generation
Explainability and governance in AI

ABSTRACT

Generative large language models (LLMs) have transformed AI by enabling rapid, human-like text generation, but they face challenges, including managing inaccurate information generation. Strategies such as prompt engineering, Retrieval-Augmented Generation (RAG), and incorporating domain-specific Knowledge Graphs (KGs) aim to address their issues. However, challenges remain in achieving the desired levels of management, repeatability, and verification of experiments, especially for developers using closed-access LLMs via web APIs, complicating integration with external tools. To tackle this, we are exploring a software architecture to enhance LLM workflows by prioritizing flexibility and traceability while promoting more accurate and explainable outputs. We describe our approach and provide a nutrition case study demonstrating its ability to integrate LLMs with RAG and KGs for more robust AI solutions.

1. Introduction

Generative large language models (LLMs) have become pervasive in recent years [1], permitting free access to virtually any web user. They are revolutionary in generating human-like text with unprecedented scale and speed. The application of LLMs to seemingly all areas of knowledge and domains of computing has exposed multitudes to the capabilities and limitations of today's emerging AI. Easy and cost-effective access to LLMs, whether through web-accessible APIs, sandboxes, or open-source toolkits, has enabled a generation of developers and researchers to integrate modern LLM-based AI into everyday applications.

Despite their successes, LLMs frequently generate inaccurate or misleading information [2], which is especially worrisome when their use is carried out on a large scale and without proper governance. Typically, these digressions from reality and logical reasoning occur because LLMs, despite their massive sampling of language, cannot replace the knowledge and experience of domain experts or the common sense of humans [3]. To mitigate these "hallucinations", a variety of prompt engineering techniques and frameworks have been proposed, including Chain-of-Thought (CoT) [4]; Self-Consistency (SC) [5]; Chain-of-Verification (CoVe) [6]; Tree-of-Thought (ToT) [7]. Complementing these prompt strategies, Retrieval-Augmented Generation (RAG) [8,9] frameworks aim to control responses by providing more task-specific information derived from corpora or a domain knowledge base. On the other hand, due to their inherent semantic structure, Knowledge Graphs (KGs) based on authoritative domain ontologies are a promising way to enhance the reliability and explainability of LLM-based solutions [10].

In this paper, we propose two complementary pathways. First, we argue that there is an urgent need for robust tools and testbed environments to explore and enhance the governance of experiments, allowing for flexibility, traceability, and repeatability. Flexibility, in the context of this paper, refers to the ability to support a range of use cases, model configurations, and interaction scenarios, which include the support of diverse prompt strategies, different LLM architectures, different data inputs, and experimental setups while maintaining ease of modification. Traceability refers to the capability to log, monitor, and trace all interactions and experiments performed with LLMs from the input data provided, such as prompts, files, or structured data, along with the pre-processing steps applied before being sent to the model, through the configurations and parameters used in each execution, such as temperature, repetition penalty, and maximum token count; to the output data, such as the responses generated by the LLMs, including metadata like timestamps, interaction context, and the specific version of the model used, allowing each step of the process to be traced. Finally, the control of repeatability refers to the ability to reproduce the same results under the same experimental conditions. In science, technology, and computational systems, it refers to the consistency of

* Corresponding author.
E-mail address: oliveh@rpi.edu (H. Santos).

https://doi.org/10.1016/j.websem.2024.100853
Received 19 November 2024; Received in revised form 9 December 2024; Accepted 17 December 2024
Available online 31 December 2024

results when an experiment or interaction is repeated using the same methods, tools, configurations, and data. So, LLM experimentation needs tools to reproduce the exact conditions of a given experiment and to evaluate the extent to which LLMs alter their responses through logged data and repeated executions, highlighting cases where these responses cannot be reliably repeated.

In a second way, we emphasize the necessity for tools that support the delivery of more provenance-aware, accurate, validated, and explainable responses from LLMs. We argue that the integration of authoritative Knowledge Graphs (KGs) and ontologies can significantly enhance the potential for validating responses, identifying provenance, and improving explanation capabilities. By leveraging structured, semantically rich representations of knowledge, these resources provide a robust framework for cross-verifying the outputs of language models [11].

As part of our exploration, we describe a promising architecture designed to facilitate the exploration of LLMs and their integration with external components like RAG systems and Knowledge Graphs. We present a nutrition case study to exemplify the promising functionalities of this architecture, being built on top of ChatBS [12], an exploratory tool for LLMs.

2. Related works

This paper emphasizes the necessity of tools and testbed environments tailored specifically for LLM experimentation. These environments should incorporate essential capabilities and features to ensure enhanced flexibility, robust traceability, and effective control over repeatability. Furthermore, we advocate for integrating mechanisms to refine and validate the responses generated by LLMs: Retrieval-Augmented Generation (RAG) and Knowledge Graphs (KG).

Several frameworks to Retrieval-Augmented Generation (RAG) [8, 9] aim to mitigate hallucination and the presence of outdated knowledge or missing information in the LLM's responses. Knowledge Graphs (KGs) have been proposed as a structured external source of information to enrich LLMs. In this sense, Pons et al. [13] use KGs to enhance LLMs for zero-shot Entity Disambiguation (ED), leveraging the hierarchical representation of the entities' classes in a KG to gradually prune the candidate space as well as the entities' descriptions to enrich the input prompt with additional factual knowledge. Clay and Jiménez-Ruiz [14] affirm that KGs are commonly used as external knowledge and may provide solutions to these challenges. This work introduces three proposals, utilizing knowledge graphs to enhance LLM generation. Firstly, dynamic knowledge graph embeddings and recommendations could allow for integrating new information and selecting relevant knowledge for response generation. Secondly, storing entities with emotional values as additional features may provide knowledge better emotionally aligned with the user input. Thirdly, integrating character information through narrative bubbles would maintain character consistency and introduce a structure that readily incorporates new information.

Regarding the use of KGs to validate the LLM's responses, GraphEval [15] is a hallucination evaluation framework based on representing information in Knowledge Graph (KG) structures. GraphEval's method identifies the specific triples in the KG that are prone to hallucinations and hence provides more insight into where in the LLM's response a hallucination has occurred. Kau et al. [11] argue that hallucination, misinformation, or gap information issues can be effectively mitigated by incorporating knowledge graphs (KGs), which organize information in structured formats and capture relationships between entities in a versatile and interpretable fashion. In support of this argument, several studies are presented that achieved performance improvement by jointly utilizing KGs and LLMs, especially in the knowledge-driven domain. Models combining KGs and LLMs typically display a better semantic understanding of knowledge, thus enabling them to perform tasks like entity typing better.

3. An architecture for redesigning the landscape of LLM experimentation

ChatBS [12] is an architecture and system introduced in 2022 by the Tetherless World Constellation at Rensselaer Polytechnic Institute. Initially envisioned as a fact-checker for LLMs, ChatBS quickly evolved into an exploratory sandbox, accelerating experimentation and facilitating end users' overall use of LLMs. ChatBS-NexGen (Fig. 1) is our enhanced proposed architecture that will allow users to prioritize aspects of flexibility, traceability, and repeatability when conducting controlled experiments using LLMs, in addition to offering mechanisms for improving, verifying, and analyzing LLM responses using RAG systems and Knowledge Graphs (KGs). Based on our experiences, we argue that the core features we present as *ChatBS-NexGen* are fundamental to the robust management of future LLM-based AI systems.

3.1. Improving the management of LLM experiments

The ChatBS-NexGen architecture allows users to interactively customize and combine advanced prompting strategies without programming knowledge. Through an intuitive interface, users can easily configure the prompt strategy (Few-Shot, Chain of Thought (CoT), Chain of Verification (CoVE)); can inform the System and User Prompts with dynamic slots; and input datasets with values assigned to these slots. Additional options, including setting the model temperature, the number of prompt resubmissions, and selecting from among different LLMs, are also readily accessible. All these features collectively contribute to the flexibility characteristic that the ChatBS-NexGen environment promotes in LLM experimentations. This flexibility is a cornerstone of the platform, enabling adaptation to diverse experimental requirements and use cases.

The Prompt Designer and Controller module generates prompt instances by combining the System Prompt, User Prompt, and input datasets according to the selected prompt strategy. It also manages the submission and re-submission of these instantiated prompts to various LLMs, orchestrating the interaction with each model. Once responses are received from the LLMs, this module organizes and prepares them for the next stage, ensuring a smooth and efficient workflow throughout the process.

The entire process is logged to ensure traceability throughout the experiment. For each LLM response, logs are associated with critical metadata, including (1) Timestamp - Recording the exact time the response was generated; (2) Prompt Data - Capturing the input prompt or query that triggered the response; (3) Model Version - Documenting the specific version or configuration of the LLM used; (4) Response Metadata - number of submissions, latency, response length; (5) Experiment Parameters, such as temperature, max tokens, and other hyperparameters; and (6) External Integrations: Tracking interactions with external tools, such as Retrieval-Augmented Generation (RAG) or Knowledge Graphs (KG), if applicable.

These two modules enhance our exploratory sandbox's flexibility and traceability capabilities while laying the groundwork for establishing repeatability control.

3.2. Accomodating RAG and KG-augmented LLM experimentation

In general terms, the Evaluation module aims to accommodate verification, benchmarking, and expert analysis by connecting the main entities in the LLM's responses with domain knowledge graphs or the Open Web. The KG platform for our experiments was Whyis [16], a next-generation provenance-aware blackboard architecture with dynamic agent interactions. The evaluation module comprises submodules for entity recognition and for building labeled entity graphs leveraging advanced capabilities of Whyis, enabling the comparison of responses across multiple LLMs and among the responses generated

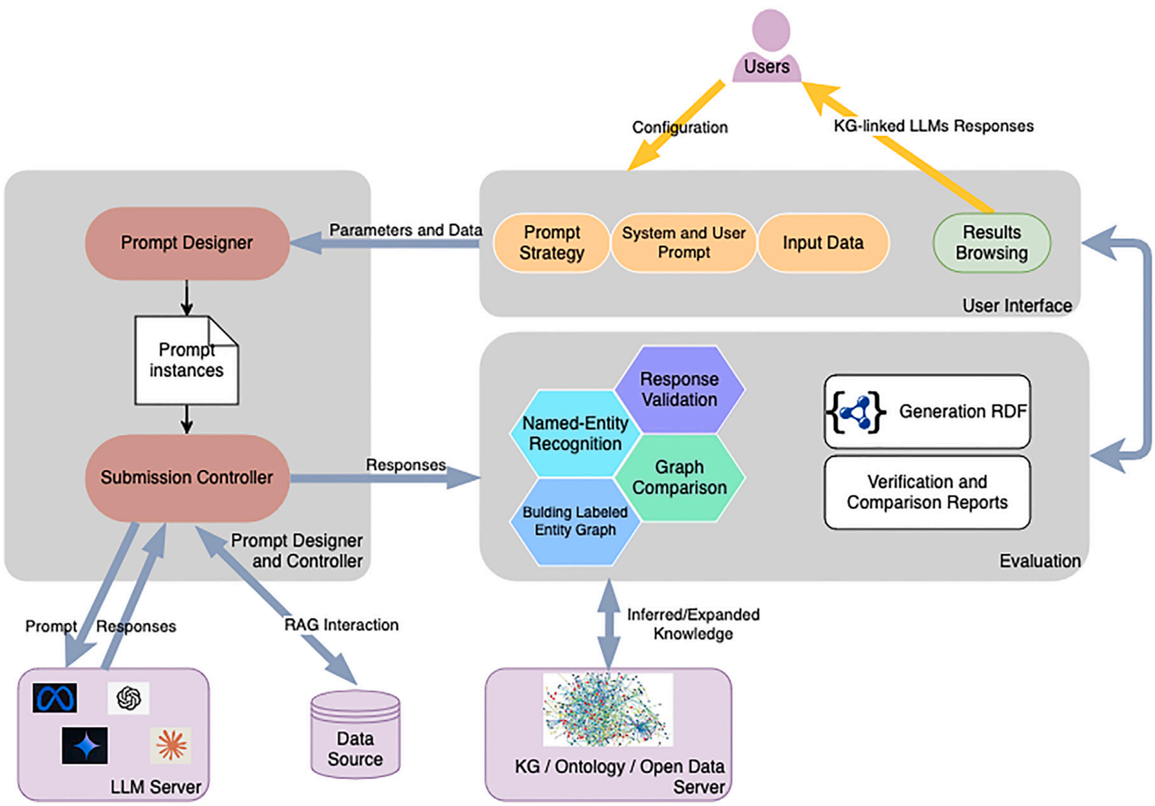

Fig. 1. A promising system flow architecture for ChatBS-NexGen.

by a single LLM from various re-submissions and response validation. These sub-modules are designed to ensure that LLM responses are traceable, improving transparency and support for explanations and providing users with consistent and verifiable KG-linked LLM responses. Finally, ChatBS-NexGen can generate the validation and comparison results in RDF format.

3.3. A nutrition case study

We experimented with our proposed architecture using a dataset containing 100 patient data of a healthcare program in a usage scenario in the food and nutrition domain. In our test scenario, a nutrition research group conducts an evaluation of LLM-generated dietary recommendations for participants in a health program. The goal was to assess the suitability of the LLM-generated recommendations, taking into account individual factors such as sex, age, gender, weight, health indicators, chronic disease diagnoses, as well as dietary restrictions and preferences.

1) In a typical test run for this scenario, the user (a member of the nutrition research group) entered the following information and selected data sources using the ChatBS-NexGen User Interface:

- Prompt Strategy: User selected “Chain of Thought (CoT)” among others strategies - Few-Shot Learning, Chain of Verification (CoVe);
- LLMs: The LLMs to be used in the experiments - for example, GPT-4o mini and Llama 3.1-8B;
- Number of prompt resubmissions: For example, 10 re-submissions to the same LLMs;
- Link for the KG: This link can be an endpoint or a Whyis link. For this example, we used the public FoodKG endpoint.[1]
- System prompt: *You are a health care assistant and you will respond to the user {name}, {age} years old and {gender}, who is a beneficiary of the health care program. Adapt dietary guidelines considering the diagnosis of {diagnosis}, recent data on height: {height}, weight: {weight}, body mass index: {bodyMassIndex}, glycated hemoglobin: glycatedH, blood pressure: {bloodPressure}, hdl: {hdl}, ldl: {ldl}, score of Framingham: {scoreF}, waist circumference: {waistC}. Also consider the food preferences: {preferences}; and allergies: {allergies}.*
- User prompt: *I need suggestions for light and energy-boosting foods to eat before my gym workout.*
- Data file with values according to the slots in the above System_Prompt and User_prompt.

2) Prompt Designer and Controller: According to the submitted user information, ChatBS-NexGen executes the Prompt Designer submodule and instantiates 100 prompts to be submitted to the selected GPT-4o mini and Llama 3 APIs. For example, a prompt instance was generated for a specific patient ANONYMOUS098 - a woman, 53 years old, Diabetic, Vegan, and with no allergies:

You are a health care assistant and you will respond to the user ANONYMOUS098, 53 years old and Female, who is a beneficiary of the health care program. Adapt dietary guidelines considering the diagnosis of DIABETIC, recent data on height: 170.4, weight: 103.7, body mass index: 35.7, glycated hemoglobin: 8.4, blood pressure: 120/80 mmHg, hdl: 58.9, ldl: 109.0, score of Framingham: 12.0, waist circumference: 88 cm. Also consider the food preferences: VEGAN; and allergies: NONE.

Next, the Submission Controller sub-module submits (and resubmits 10 times) each instantiated prompt to all selected LLMs, receives their responses, and passes them along to the Evaluation step. For example, for the patient ANONYMOUS098 described above, Llama 3.1-8B gave the following recommendation in the first submission:

Result 1 : [MetaLlama3.18BInstruct] Based on your dietary preferences and restrictions, I'd like to suggest some light and energy-boosting foods that are perfect for pre-workout fueling: 1. Banana with almond milk: A

[1] https://inciteprojects.idea.rpi.edu/foodkg/#query.

classic combo that's easy to digest and provides sustained energy. Choose a ripe banana and almond milk that's unsweetened and fortified with vitamin D. 2. Avocado toast: Whole-grain toast topped with mashed avocado, salt, and pepper is a satisfying and filling option. Look for vegan-friendly bread options. 3. Oatmeal with fruit: Steel-cut oats or rolled oats cooked with plant-based milk and topped with your favorite fruits (such as berries, sliced banana, or diced apples) provide complex carbohydrates for sustained energy. 4. Energy balls: Mix rolled oats, nut butter (such as peanut butter or almond butter), and ***honey*** *or* ***maple syrup*** *to create bite-sized energy balls. These no-bake bites are easy to make and pack. 5. Smoothie bowl: Blend your favorite fruits, spinach, and plant-based milk, then top with granola, nuts, or seeds for added crunch. Choose a bowl that's low in sugar and rich in fiber. 6. Whole-grain crackers with hummus: Spread hummus on whole-grain crackers for a satisfying snack that's high in protein and fiber. 7. Coconut water: This natural electrolyte drink helps replenish fluids and provides a refreshing pick-me-up. 8. Dark chocolate: Enjoy a small piece of dark chocolate (at least 70% cocoa) for a mood boost and energy kick. Just be mindful of portion sizes! 9. Trail mix: A mix of nuts, seeds, and dried fruits is a convenient snack that's easy to devour on-the-go. Look for vegan-friendly options without added sugars. 10. Green juice: Juice fresh greens like kale, spinach, or collard greens with lemon juice and apple cider vinegar for a nutrient-dense drink that's easy to digest. Remember to stay hydrated by drinking plenty of water before, during, and after your workout. Aim to eat your snack 30-60 min prior to exercise to allow for digestion and optimal energy levels. Enjoy your workout*

3) Evaluation of Responses: ChatBS-NexGen workflow can accommodate RAG-style augmentation using data sources and KGs to validate the responses. First, ChatBS-NexGen recognizes the food names mentioned in the Llama-3.1 response. Next, ChatBS-NextGen links each of the food names into the FoodKG [17] and our WhatToMake ontology.[2] Table 1 presents the list of 34 food names recognized in the Llama3.1 response to ANONYMOUS098, the indication of which food was linked into FoodKG (24 foods) and the respective Glycemic Index (GI). For example, "HONEY" was found in FoodKG with GI = 60.

ChatBS-NexGen can leverage this association with FoodKG and ontologies to facilitate validation, analysis, and comparison of responses from LLMs by domain experts.

Highlighted analysis leveraged by ChatBS-NexGen and FoodKG include:

- Validate the recommendation using the KG and ontologies: as FoodKG and WhatToMake ontology encodes recipes and nutritional information, it is possible to validate any food based on its glycemic index (GI) and, consequently, provide information that can help nutrition professionals assess whether LLM recommendations are appropriate for a particular health condition – DIABETES, for example.
  In this experiment with 100 patient data, we submitted the 100 systems_prompt instantiated to LLMs GPT-4o mini and Llama, and we obtained 200 responses (100 from each LLM). Table 2 presents some statistics extracted after analyzing the responses. In general, the LLMs recommended 483 distinct types of foods to the 100 patients, and it was possible to validate 178 foods (36.8%) with FoodKG. Considering the universe of 30 diabetic patients, the LLMs recommended 260 distinct types of foods, and it was possible to validate 123 foods (47.3%) with FoodKG. Besides this, we encountered the recommendation of 10 different foods with high GI ≥ 50[3] to diabetic patients: "brown rice" (GI = 50), "crackers"(GI = 80), "honey"(GI = 60), "kiwi"(GI = 50), "mango"(GI = 56), "maple syrup"(GI = 65), "pineapple"(GI = 66), "popcorn"(GI = 55), "rice cake"(GI = 85), and "watermelon" (GI = 75).

[2] https://foodkg.github.io/whattomake.html#conceptualModel.

[3] This lower limit for the GI was an arbitrary choice to illustrate validations to be performed on the LLMs' responses.

Table 1
Foods recommended by Llama3.1-8B for ANONYMOUS098 patient associated with the respective ingredient in FoodKG and Glycemic Index (GI).

Food recommended	FoodKG.ingredientName	GI
"almond milk"	"almond milk"	30
"apple cider vinegar"	"apple cider vinegar"	5
"apple"	"apple"	
"avocado"	"avocado"	10
"banana"	"banana"	48
"berry"		
"coconut water"	"coconut water"	
"dark chocolate"	"dark chocolate"	23
"dried fruit"	"dried fruit"	
"energy ball"		
"fruit"	"fruit"	
"granola"		
"green juice"		
"honey"	"honey"	60
"hummus"	"hummus"	25
"kale"	"kale"	
"lemon juice"	"lemon juice"	20
"maple syrup"	"maple syrup"	65
"nut butter"	"nut butter"	
"nut"		
"oatmeal"	"oatmeal"	40
"peanut butter"	"peanut butter"	
"pepper"	"pepper"	
"plant-based milk"		
"rolled oat"	"rolled oat"	
"salt"	"salt"	0
"seed"		
"smoothie bowl"	"smoothie bowl"	
"spinach"	"spinach"	5
"steel-cut oat"		
"trail mix"	"trail mix"	
"water"	"water"	
"whole-grain cracker"		
"whole-grain toast"		

- Compare inter-LLMs KG-linked responses - ChatBS-NexGen enables comparing responses generated by different LLMs, such as Llama-3.1 and GPT-4o mini. In this case study, it was observed that the coverage of FoodKG for distinct foods recommended by Llama-3.1 was higher than the coverage for foods recommended by GPT4o-mini (42% compared to 39%, to all patients, and 56% compared to 48%, to diabetic patients). Regarding foods with a high glycemic index recommended to diabetic patients, both LLMs suggested six distinct items. Fig. 2 illustrates, in a graph, the foods with a high glycemic index (GI) ≥ 50), recommended by each LLM -Llama3.1 and GPT4o mini. The edges are annotated with the number of distinct diabetic patients for whom the respective LLM recommended that food. For instance, Llama-3.1 recommended "Honey" to 10 diabetic patients, corresponding to 33% of the diabetic patients, while GPT4o-mini recommended "pineapple" to 9 diabetic patients, corresponding to 30% of the diabetic patients.
- Compare intra-LLM KG-linked responses - ChatBS-NexGen can also compare multiple responses from the same LLM, such as ten GPT-4o mini responses generated through repeated prompt submissions. This feature can, for instance, calculate a faithfulness index for GPT-4o by analyzing the consistency of responses, such as the variation in foods mentioned across the ten responses.
  In this case study, we analyzed the 10 responses provided by GPT-4o mini for each of the 100 patients by calculating the Jaccard coefficient, which measures the similarity between the lists of recommended foods across all pairs of responses. For instance, two lists of recommended foods are displayed below, extracted from GPT-4o mini's responses for patient ANONYMOUS098, specifically from prompt re-submissions 1 and 3.
  submission1-ANONYMOUS098-GPT4o-mini = ["almond butter", "almond milk", "apple", "avocado", "banana", "berry", "carrot",

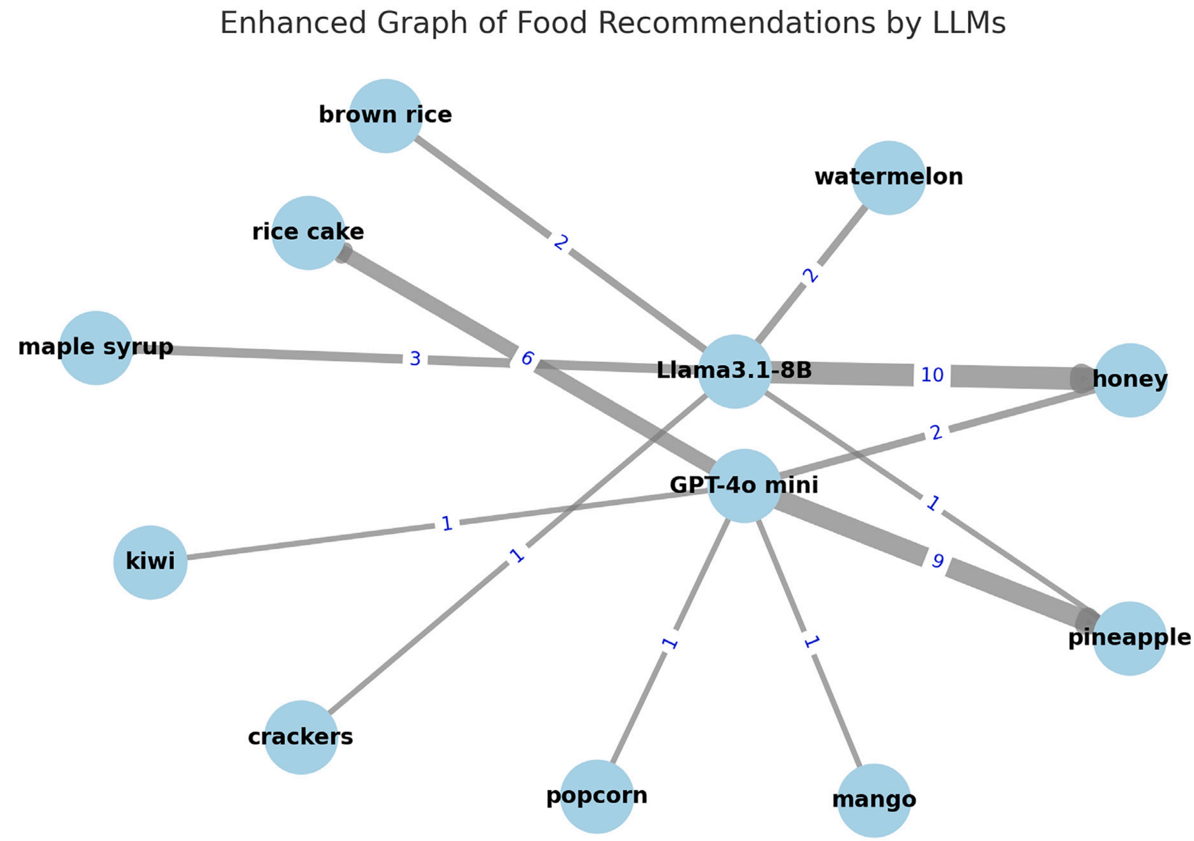

Fig. 2. A Graph with High Glycemic Index Foods (GI ≥ 50) recommended by LLMs: Llama3.1-8B vs. GPT-4o Mini.

"chia seed pudding", "chia seed", "coconut yogurt", "cucumber", "edamame", "energy bite", "fruit salad", "fruit", "granola", "hummus", "maple syrup", "oat", "oatmeal", "peanut butter", "pear", "plant-based protein powder", "rice cake", "sliced banana", "smoothie", "spinach", "vanilla extract", "vegetable stick", "walnut", "water", "whole grain bread", "whole grain toast"]
submission3-ANONYMOUS098-GPT4o-mini = ["almond butter", "almond", "apple", "avocado", "banana smoothie", "banana", "bell pepper strip", "berry", "carrot stick", "chia seed pudding", "chia seed", "cucumber slice", "homemade energy ball", "hummus", "mixed nut and seed", "nut butter", "oat", "oatmeal", "pumpkin seed", "spinach", "stevia", "sunflower seed", "unsweetened almond milk", "walnut", "whole grain rice cake"]
For this pair of food lists, the Jaccard coefficient is 0.2608, indicating that 26% of the recommended foods were identical between the two responses. We extended this analysis to calculate the similarity coefficient for all 10 responses from both GPT-4o mini and Llama3.1-8B. The global Jaccard similarity index for GPT-4o mini was 0.05, meaning only 5% of the foods were shared across all responses to the same patient. For Llama3.1-8B, the global Jaccard similarity index was slightly lower at 0.044, indicating a similar level of variability in the recommendations.
Figs. 3 and 4 present the heatmaps of the Jaccard coefficients calculated for each pair of responses from GPT-4o mini and Llama3.1-8B, respectively. In both LLMs, the results show low similarity or high variability, suggesting a significant challenge in repeating the recommendations consistently across responses.

Table 2
Analysis of the food recommendation for all patients and specific cases of 30 diabetic patients.

Description	Llama3.1-8B	GPT-4o mini	Total	%
All patients (100)				
Distinct food recommendations	315	308	483	
Coverage of FoodKG	135	122	178	36.8%
Diabetic patients (30)				
Distinct food recommendations	172	164	260	
Coverage of FoodKG	97	80	123	47.3%
Foods with GI ≥ 50	6	. 6	10	8.3%

4. Exploring knowledge graphs: Limitations, research challenges, and proposed solutions

Observed Limitations of Knowledge Graphs in the Application Area: The utilization of Knowledge Graphs (KGs) within the specified application area revealed limitations. A significant issue is the incompleteness of the data. For instance, while KGs contained certain food items, many were missing entirely or lacked critical attributes such as glycemic index (GI) values. This gap in data coverage constrains the applicability of KGs for comprehensive analyses. Another limitation is the query formulation for domain-specific applications. While this aspect did not present a direct challenge in the present use case, designing queries to extract relevant data from KGs can pose substantial difficulties in more general or less structured scenarios. These limitations underline the necessity for advancements in data comprehensiveness and user-friendly, ontology-driven KG query generation mechanisms.

Research Questions and Challenges in Knowledge Graph Research: Several research questions emerge from the observed limitations. First, "How can KGs be enriched to include comprehensive and authoritative data across diverse domains?" since many KGs lack critical attributes or are incomplete, which limits their applicability for detailed and reliable analyses. Second, adopting standardized and authoritative ontologies is crucial to enhancing interoperability across diverse systems. Ontologies that adhere to standardized frameworks

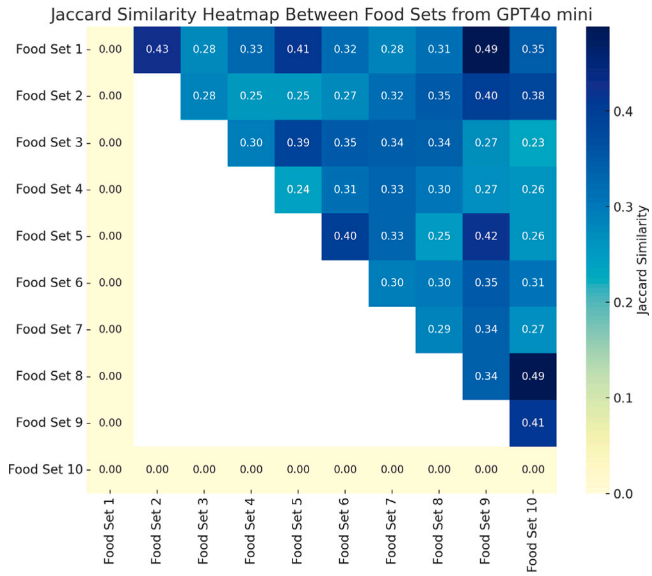

Fig. 3. A heatmap expressing the similarity level (Jaccard coefficient) between each pair of responses from GPT-4o mini.

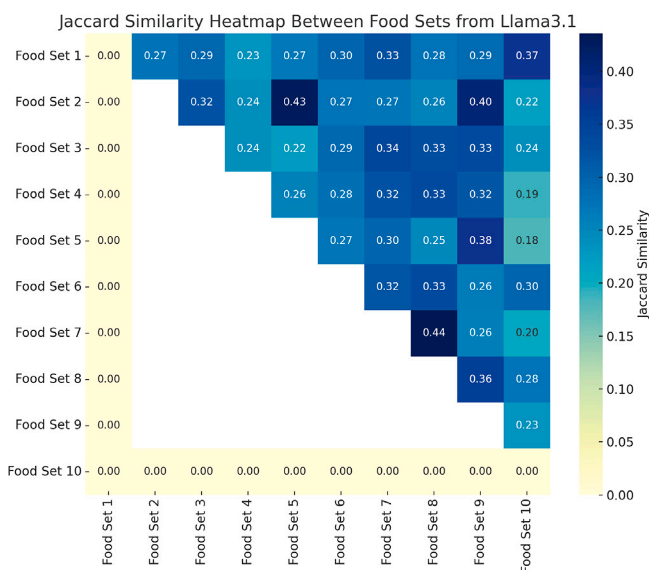

Fig. 4. A heatmap expressing the similarity level (Jaccard coefficient) between each pair of responses from Llama3.1-8B.

can bridge gaps in data compatibility and facilitate seamless integration across various datasets and domains. Another critical challenge involves addressing the limited provenance within KGs, which complicates the verification of data sources. Provenance is fundamental for ensuring the reliability of information within KGs, particularly in domains where accuracy and traceability are essential. Future research must focus on developing methods to embed detailed provenance metadata within KGs and demonstrate the application of provenance metadata to improve reliability and utility. Finally, the fourth question is how KG query mechanisms can be improved to simplify domain-specific data retrieval for users.

Claims and Supporting Evidence: The case study supports a pivotal claim: the utility of an expanded architecture for KG-powered LLMs experimentation. This claim is substantiated by using the ChatBS-NexGen prototype in a real-world scenario within a healthcare project to evaluate the adequacy of LLM dietary recommendations in addressing patients' health conditions. In *ad hoc* experiments with LLMs conducted by healthcare specialists, dietary recommendations were observed to include foods unsuitable for diabetic patients. The prototype demonstrated the potential benefits of integrating KGs into a structured framework for managing LLM experimentation workflows. Through 100 prompts resubmitted 10 times to two LLMs, it was possible to identify inappropriate recommendations and high variability in responses. The proposed architecture aims to transition from the current state of manual experimentation to a more robust testbed environment for LLM experimentation. By leveraging KG-based methodologies, this architecture promises enhanced automation and consistency in handling LLM responses.

5. Conclusion and future directions

We believe there is a wide range of opportunities for knowledge graphs in the evolving AI landscape and that exploration architectures and frameworks will be crucial for the emerging field. Our ChatBS-NexGen architecture proposes advancements in the interaction, management, and evaluation of LLM responses by offering modularity and parametrization, multi-model support, integration with structured knowledge, and a dedicated module for the automated assessment inter- and intra-LLMs. The proposed architecture ensures experimental traceability and repeatability control through extensive logging.

We see many directions for developing this vision, including support for multi-modality, enriching experiments with more complex inputs and outputs; incorporation of auto-prompting strategies; the evolution of evaluation support through integration with external tools, such as XAI (Explainable AI) frameworks with advanced explanation capabilities (multi-level and causal); incorporation of logical consistency validators and adversarial scenario simulators; dynamic integration of diverse metrics for contextual evaluation (e.g., factuality, relevance, coherence); incorporation of components focused on security validation and bias mitigation.

CRediT authorship contribution statement

John S. Erickson: Writing – review & editing, Writing – original draft, Software, Conceptualization. **Henrique Santos:** Writing – review & editing, Software, Conceptualization. **Vládia Pinheiro:** Writing – review & editing, Writing – original draft, Methodology, Data curation, Conceptualization. **Jamie P. McCusker:** Software, Conceptualization. **Deborah L. McGuinness:** Writing – review & editing, Project administration, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work is partially funded by the National Institute of Mental Health "Support for the RCADS Data Collection Measure" project, grant number 75N95022C00018-0-9999-1, the DARPA Environment-driven Conceptual Learning (ECOLE) program, grant number HR00112390059, and Rensselaer's Tetherless World Constellation. The third author is supported by FUNCAP-CE-Brazil.

Data availability

Used data is publicly available and cited within the manuscript.

References

[1] J. Yang, H. Jin, R. Tang, X. Han, Q. Feng, H. Jiang, S. Zhong, B. Yin, X. Hu, Harnessing the power of LLMs in practice: A survey on ChatGPT and beyond, ACM Trans. Knowl. Discov. Data 18 (6) (2024) 160:1–160:32.

[2] S. Kabir, D.N. Udo-Imeh, B. Kou, T. Zhang, Is stack overflow obsolete? An empirical study of the characteristics of ChatGPT answers to stack overflow questions, in: Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems, CHI '24, Association for Computing Machinery, New York, NY, USA, 2024.

[3] M. Kejriwal, H. Santos, A.M. Mulvehill, K. Shen, D.L. McGuinness, H. Lieberman, Can AI have common sense? Finding out will be key to achieving machine intelligence, Nature 634 (8033) (2024) 291–294.

[4] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E.H. Chi, Q.V. Le, D. Zhou, Chain-of-thought prompting elicits reasoning in large language models, in: Proceedings of the 36th International Conference on Neural Information Processing Systems, NIPS '22, Curran Associates Inc., Red Hook, NY, USA, 2024, pp. 24824–24837.

[5] X. Wang, J. Wei, D. Schuurmans, Q. Le, E. Chi, S. Narang, A. Chowdhery, D. Zhou, Self-consistency improves chain of thought reasoning in language models, 2023, arXiv:2203.11171.

[6] S. Dhuliawala, M. Komeili, J. Xu, R. Raileanu, X. Li, A. Celikyilmaz, J. Weston, Chain-of-verification reduces hallucination in large language models, in: Findings of the Association for Computational Linguistics: ACL 2024, Association for Computational Linguistics, Bangkok, Thailand, 2024, pp. 3563–3578.

[7] S. Yao, D. Yu, J. Zhao, I. Shafran, T.L. Griffiths, Y. Cao, K. Narasimhan, Tree of thoughts: deliberate problem solving with large language models, in: Proceedings of the 37th International Conference on Neural Information Processing Systems, NIPS '23, Curran Associates Inc., Red Hook, NY, USA, 2024, pp. 11809–11822.

[8] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive NLP tasks, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20, Curran Associates Inc., Red Hook, NY, USA, 2020.

[9] W. Fan, Y. Ding, L. Ning, S. Wang, H. Li, D. Yin, T.-S. Chua, Q. Li, A survey on RAG meeting LLMs: Towards retrieval-augmented large language models, in: Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD '24, Association for Computing Machinery, New York, NY, USA, 2024, pp. 6491–6501.

[10] S. Chari, O. Seneviratne, D.M. Gruen, M.A. Foreman, A.K. Das, D.L. McGuinness, Explanation ontology: A model of explanations for user-centered AI, in: The Semantic Web – ISWC 2020, in: Lecture Notes in Computer Science, Springer International Publishing, Cham, 2020, pp. 228–243.

[11] A. Kau, X. He, A. Nambissan, A. Astudillo, H. Yin, A. Aryani, Combining knowledge graphs and large language models, 2024, arXiv:2407.06564. URL: https://arxiv.org/abs/2407.06564.

[12] J.S. Erickson, H. Santos, J.P. McCusker, S. Shirai, D.L. McGuinness, J.A. Hendler, ChatBS: An exploratory sandbox for bridging large language models with the open web, in: Proceedings of the ISWC 2024 Posters, Demos and Industry Tracks: from Novel Ideas To Industrial Practice, in: CEUR Workshop Proceedings, vol. 3828, CEUR, Maryland, USA, 2024, ISSN: 1613-0073.

[13] G. Pons, B. Bilalli, A. Queralt, Knowledge graphs for enhancing large language models in entity disambiguation, in: G. Demartini, K. Hose, M. Acosta, M. Palmonari, G. Cheng, H. Skaf-Molli, N. Ferranti, D. Hernández, A. Hogan (Eds.), The Semantic Web – ISWC 2024, Springer Nature Switzerland, Cham, 2025, pp. 162–179.

[14] A. Clay, E. Jiménez-Ruiz, Information for conversation generation: Proposals utilising knowledge graphs, 2024, arXiv:2410.16196. URL: https://arxiv.org/abs/2410.16196.

[15] H. Sansford, N. Richardson, H.P. Maretic, J.N. Saada, Grapheval: A knowledge-graph based LLM hallucination evaluation framework, 2024, arXiv:2407.10793. URL: https://arxiv.org/abs/2407.10793.

[16] J. McCusker, D.L. McGuinness, Whyis 2: An open source framework for knowledge graph development and research, in: The Semantic Web, in: Lecture Notes in Computer Science, Springer Nature Switzerland, Cham, 2023, pp. 538–554.

[17] S. Haussmann, O. Seneviratne, Y. Chen, Y. Ne'eman, J. Codella, C.-H. Chen, D.L. McGuinness, M.J. Zaki, FoodKG: A semantics-driven knowledge graph for food recommendation, in: The Semantic Web – ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part II, Springer-Verlag, Berlin, Heidelberg, 2019, pp. 146–162.