A RAG-Powered Academic Chatbot with Ontology-Driven Factual Verification

Vitor Mateus R. do Amaral, Luiza Espinosa, Paula D. Rigo, Gabriel M. Lunardi

[1]Centro de Tecnologia - Universidade Federal de Santa Maria (UFSM)

{luiza.silveira, vitor.romancini}@acad.ufsm.br

{paula.rigo, gabriel.lunardi}@ufsm.br

Abstract. *This paper presents an academic chatbot developed to support services at UFSM, with a Retrieval-Augmented Generation (RAG) architecture. The system uses PyPDF2 for processing official documents, LangChain and FAISS for vector retrieval, and Zephyr 7B for response generation. To ensure factual accuracy, our ontology enables automated fact-checking via SPARQL queries. The chatbot is integrated into Telegram with structured conversation flows tailored for Computer Engineering students as case study. Experimental results show gains in factual accuracy through ontology-based verification, reducing hallucinations and increasing system reliability.*

1. Introduction

The advancement of artificial intelligence (AI) is revolutionizing various fields, including education. Large language models (LLMs), such as GPT, LLama, and Deepseek, are powerful tools for generating and understanding natural language, enabling applications ranging from student assistance to the automation of administrative processes in educational institutions [de O. Figueiredo et al. 2023]. At the Federal University of Santa Maria (UFSM), the academic offices are responsible for services such as issuing documents, making enrollment adjustments, and addressing academic inquiries. However, the high volume of requests, combined with the limited digitalization of processes, leads to an overload in service delivery.

To address this problem, this work proposes the development a chatbot based on Retrieval Augmented Generation (RAG), an approach that combines the search for precise information in databases with the contextualized text generation capabilities provided by LLMs [Lewis et al. 2020]. Unlike traditional chatbots, which rely exclusively on predefined responses or free text generation, the use of RAG allows the system to access official documents, such as academic regulations, curricula, and information on enrollments, internships, and TCCs, ensuring accurate, reliable, and contextualized responses. Furthermore, RAG proves especially advantageous for handling the dynamic nature of academic data, which changes over time (e.g., updates to regulations or curricula), as it does not require retraining the LLM, only updating the reference database.

Nonetheless, one of the main challenges in deploying RAG-based systems lies in ensuring the factual accuracy of the generated responses. To address this, we propose the integration of a domain-specific ontology as a structured and verifiable source of truth. This ontology models key academic entities and relationships, such as prerequisites, course topics, and semester allocations, allowing for automated fact-checking of chatbot responses and increasing user trust in the system. This solution is particularly relevant for UFSM, as it aims not only to automate service delivery but also to reduce the workload of the academic office and provide students with more agile and standardized support, initially focusing on Computer Engineering students.

2. Related Work

Based on previous work implemented within the educational ecosystem, chatbots are widely used to optimize academic support. For example, [Silva 2023] develops the **Polímata** chatbot, which assists in answering questions regarding TCC regulations and internships, provides basic information about the coordination, the Athletics, and the Computer Science Academic Center at Unifap, and helps users locate spaces within the university. Although efficient in its functionalities, Polímata relies on predefined responses, which limits its ability to handle more complex or dynamic queries. Other examples include **GURIBO**, developed by [Fontanari 2022], which uses the Rasa tool to support the academic office at the Alegrete campus of UNIPAMPA, especially for inquiries related to internships; the **UFC chatbot**, developed by [Brito 2017] for the Information Systems course website, implemented using the IBM Watson Conversation platform with a content management system (CMS) built in Django; and **HELENA**, developed by [Monteiro 2021] to assist students in the Department of Computing (DECOM) at UFOP, implemented using a combination of IBM Watson Assistant for Natural Language Processing (NLP) and AWS cloud services for scalability and availability.

However, like Polímata, these chatbots also face limitations in handling the dynamic nature of academic information, as their knowledge bases need constant manual updates. Unlike these approaches, this work proposes the use of RAG to develop a chatbot that accesses up-to-date information in real time (academic regulations, curricula, etc.) without requiring model retraining. This solution aims to provide precise and contextualized responses, reduce the workload of the academic office, and improve support for students.

Beyond chatbot architectures, recent research has emphasized the role of ontologies and knowledge graphs in enhancing intelligent systems. Bax and Gonçalves [Bax and de Assis Gonçalves 2019] propose a semantic integration framework for scientific data using foundational ontologies to create reusable knowledge graphs. Silva [Silva 2022] presents a fact-checking model that uses knowledge graphs to validate the accuracy of chatbot responses, showing the benefits of structured semantic reasoning in natural language systems. Furthermore, Oliveira et al. [Souza Filho 2024] propose a chatbot framework to assist scientific research, highlighting the value of conversational agents in research environments, although without yet incorporating dynamic fact validation.

3. The RAG-Powered Academic chatbot

This section presents the complete architecture of the system, detailing the stages of data preparation, the integration with RAG, the user interaction interface, and the use of an academic ontology for factual verification.

3.1. Data Preparation and RAG Architecture

The data pipeline begins with extracting content from PDF documents using PyPDF2, which converts files into structured formats such as Markdown or JSON. Once the documents are processed, they are passed to LangChain[1] for segmentation. The textual content is divided into chunks based on two criteria: whenever a double line break is detected, indicating a natural section break, and, if there are still chunks exceeding the 512-token limit, they are further subdivided with overlapping segments to preserve context across boundaries. This approach helps maintain contextual continuity, improving performance in retrieval.

Each chunk is then transformed into an embedding, a vector representation of the text, using a language model. These embeddings are stored in a vector database to enable fast similarity, based retrieval. When the user submits a query, it is also converted into an embedding and compared to the stored vectors using cosine similarity. The most relevant chunks are selected as context for the generation stage.

To optimize information retrieval, we adopt the Maximum Marginal Relevance (MMR) method, which balances relevance and diversity among retrieved documents. The selected chunks, along with the original user query, are then passed to a language model (LLM), specifically Zephyr 7B, a 7-billion-parameter GPT-like model fine-tuned on a mix of publicly available and synthetic datasets. As our goal is to use the LLM solely to interpret the retrieved context and formulate an accurate response, Zephyr 7B demonstrated satisfactory results compared to other models, standing out for being lightweight when compared to LLaMA and freely available unlike proprietary models such as ChatGPT, as noted in its official documentation [Tunstall et al. 2023]. The entire process is illustrated in Figure 1.

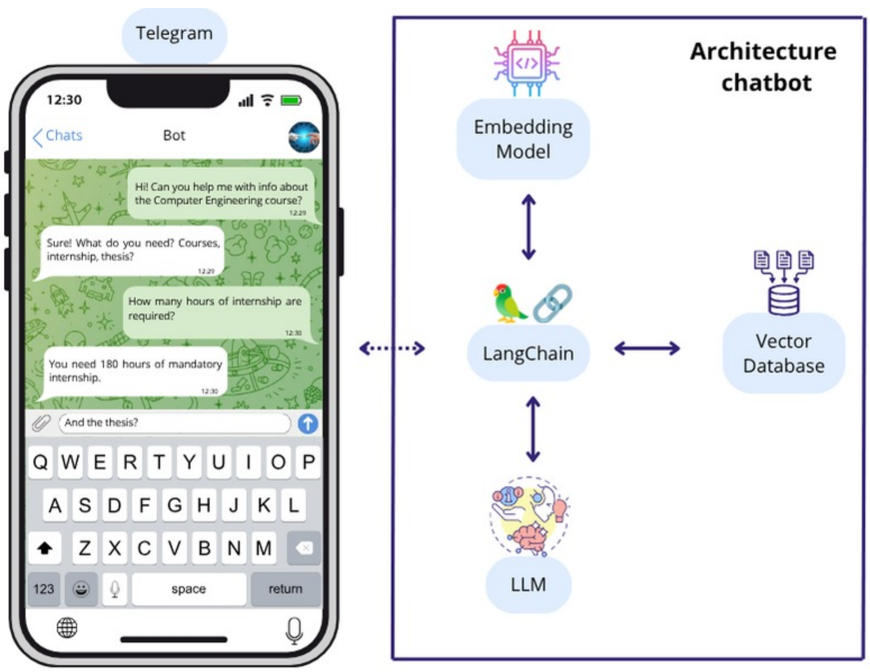

Figure 1. Solution Architecture

To facilitate user interaction, the chatbot was integrated with the Telegram messaging platform, chosen for its accessibility, simplicity, and used in the community. Telegram

[1] Available at: https://www.langchain.com

[2] offers a well-documented Bot API, allows rapid deployment, and is compatible with mobile and desktop platforms, making it an ideal choice for research-oriented conversational systems. Furthermore, it supports message formatting, buttons, and conversational states, which are essential for structured interactions.

The interface follows an adaptive conversation flow that guides the user through the process. At the beginning of the interaction, the chatbot may request basic information, such as the user's name or the type of inquiry, to personalize the experience and contextualize the responses. This flow-based structure is essential to handle the variability of natural language input. Even when users deviate from expected formats or phrasing, the system is designed to adapt dynamically and recover the intended meaning, ensuring that the information provided remains accurate and complete.

3.2. Domain Ontology for Factual Verification

A key contribution of this work is the introduction of an academic ontology to verify the factual accuracy of chatbot responses. The ontology models core elements of the Computer Engineering curriculum at UFSM using tools such as Protégé and is based on the guidelines the authors [Lichtnow et al. 2025] have been proposed.

This structured knowledge base serves as a "source of truth" for verifying the correctness of chatbot answers. When a response is generated by the LLM, it is parsed using prompt engineering to extract factual triples (e.g., course relationships) into a JSON format. These triples are then validated against the ontology using SPARQL queries. If the extracted facts match the ontology, the response is considered factual; otherwise, it is flagged as potentially incorrect. This verification layer enhances the reliability and transparency of the system and opens opportunities for using the ontology not only as an evaluation tool but also as a guide to improve retrieval and generation, an approach we identify as Ontology-Augmented RAG.

To support this process, we designed the ontology using OWL (Web Ontology Language) and it models essential entities from the curriculum domain, including disciplines, professors, courses, semesters, topics, and departments, all organized as subclasses of the general concept owl:Thing. These elements are connected by semantic properties such as hasPrerequisite (temPreRequisito), which relates one course to another as a prerequisite; belongsToSemester (pertenceAoSemestre), linking a discipline to its academic period; isTaughtBy (éLecionadaPor), assigning teaching responsibilities to professors; coversTopic (abordaTopico), connecting courses to their content; and isPartOfProgram (fazParteDoCurso), indicating which course a discipline belongs to.

This ontological structure enables automatic verification of facts, allows for rich semantic querying, and facilitates integration with LLMs through SPARQL and rule-based logic. Because it is based on open standards such as OWL, RDF, and SPARQL, the ontology remains reusable and interoperable across different platforms. It can be extended by researchers from other institutions to model similar academic domains, and also supports applications like curriculum validation, academic planning, and retrieval-augmented generation with semantic grounding. The relationships among the core classes are illustrated in Figure 3.

[2] Available at: https://core.telegram.org

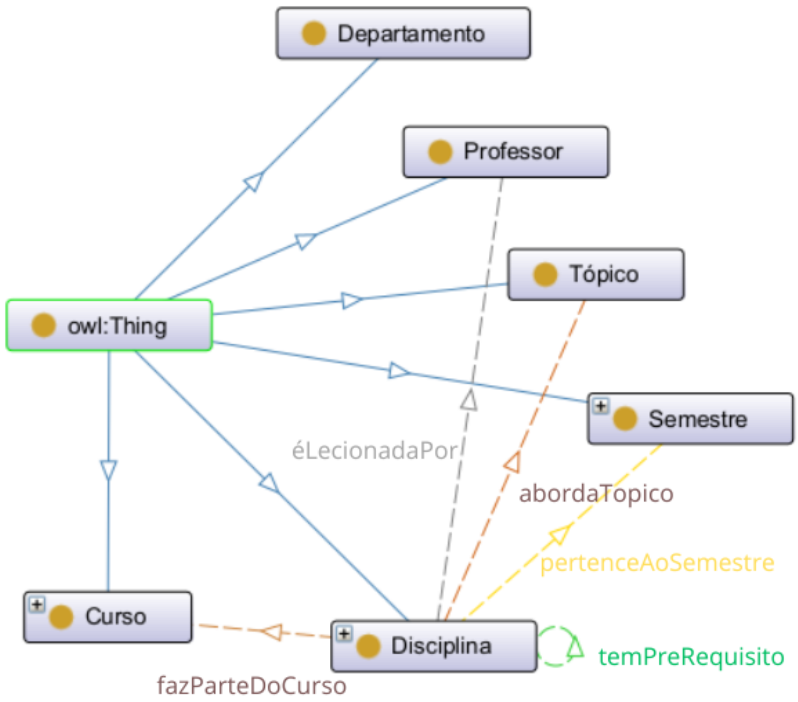

Figure 2. Ontology Diagram

4. Experimental Evaluation

This section presents the methodology and procedures we adopted to evaluate the effectiveness and factual accuracy of the system, combining a RAG-based retrieval pipeline with ontology-based fact verification.

4.1. Experiment Setup

The evaluation environment includes a vector retrieval module implemented using the FAISS library [Jégou et al. 2025], which is optimized for high-performance similarity search. Cosine similarity is used to compare the embeddings of user queries and document chunks. All vectors are normalized to ensure unit length, as required for accurate cosine similarity calculations. As an alternative, dot product or Euclidean distance could be used, but cosine similarity was selected due to its interpretability and robustness in high-dimensional spaces.

The language model responsible for generating responses is Zephyr 7B, selected for its balance between performance and computational cost. The model processes the retrieved context and the user query to generate a contextualized answer. To further enhance precision and reduce hallucinations, a re ranking step is applied using the ColBERTv2 model [Santhanam et al. 2022], which evaluates candidate chunks using a cross-encoder approach for improved matching quality.

4.2. Baseline: Generation Quality Assessment

In this initial stage, the evaluation of the proposed chatbot focuses on the quality of the responses generated by the base model, which is built upon a RAG architecture. At this point, the analysis is primarily qualitative, aiming to assess the fluency, coherence, and linguistic structure of the responses produced in natural language.

Figures 3 and 4 present examples of the responses. The RAG model demonstrates remarkable proficiency in constructing logically articulated sentences, as well as presenting information in a clear, objective, and well-organized manner. The responses adhere to the grammatical and semantic conventions of the Portuguese language and, on the surface, appear convincing, effectively simulating the responses of a human expert when addressing topics related to the academic curriculum.

```
=================================Answer=================================
Sim, existem disciplinas com foco em matemática computacional. De acordo com o
texto fornecido, as disciplinas "Matemática Computacional I", "II", "III" e "IV"
têm como objetivo melhorar o aprendizado de matemática, promovendo a integração
entre disciplinas básicas por meio da prática de resolução de programas,
auxiliada por software matemático e programação. Além disso, as disciplinas com
foco em projeto, como a "Projeto Integrador em Engenharia de Computação I",
"II", "III" e "IV", além da interdisciplinaridade, colocam o acadêmico como
protagonista de sua formação e podem ser empregadas como uma das principais
ferramentas para o desenvolvimento de habilidades pessoais, interpessoais e
profissionais. As disciplinas de projeto têm complexidade crescente ao longo do
curso, visando que os acadêmicos desenvolvam o ciclo de vida de produtos,
sistemas, processos ou serviços, e assim consolidassem habilidades científicas,
técnicas, pessoais, interpessoais e profissionais.
```

Figure 3. LLM's response

```
=================================Answer=================================
As normas e requisitos para a realização do estágio supervisionado obrigatório,
conforme estabelece o Ministério da Educação e a Universidade Federal de Santa
Maria, como detalhado nos extratos dos documentos 0, 1, 2 e 3, são:  1.
Conclusão de todas as disciplinas obrigatórias; 2. Contabilização de 390 horas
de extensão; 3. Firmagem de convênio com a empresa ou instituição, se houver,
desde que garanta todas as condições legais necessárias para a realização das
atividades de estágio; 4. Elaboração do Plano de Estágio, com acordo entre o
estagiário, o orientador e o supervisor de estágio, entregue no início do último
semestre do curso; 5. Realização das atividades diretamente relacionadas às
tarefas em desenvolvimento na empresa ou instituição, conhecida como campo de
Estágio; 6. Entrega de relatórios parciais a cada 6 meses, caso o estágio se
estenda por mais do que esse tempo; 7. Apresentação do relatório final ao final
do período de realização das atividades.  Espero que tenha respondido à sua
pergunta com exatidão.
```

Figure 4. LLM's response

However, despite their high linguistic quality, these responses do not necessarily guarantee factual accuracy. This limitation stems from the lack of internal mechanisms in the RAG model to validate the truthfulness of the information provided. As a result, well-written answers may still contain inaccuracies regarding critical data, such as course prerequisites, workload, or credit requirements. In light of this gap, this study proposes a complementary solution focused on automated and quantitative factual verification, which is presented in the following section.

4.3. Proposal: Ontology-Based Factual Assessment

To overcome the limitations of qualitative analysis, the propose an evaluation method centered on factual verification of the chatbot's responses. This approach constitutes the main contribution of this work and leverages a domain-specific academic ontology as the ground truth to automatically validate the information generated. The proposed methodology enables objective, scalable, and replicable measurement of response accuracy.

To validate the chatbot's factual accuracy, we constructed a custom test set composed of 400 manually curated question-answer pairs, all derived from the official course documentation of the Computer Engineering program at UFSM, available in the github repository [3]. This dataset includes factual, verifiable questions whose answers are explicitly represented in the ontology. The creation process involved analyzing the full course syllabus (PPC), extracting key curricular information, and formulating questions that span a range of semantic relationships modeled in the ontology, such as course prerequisites (e.g., "What is the prerequisite for the Calculus B course?"), workload and credit requirements (e.g., "How many hours of Complementary Academic Activities are required for graduation?"), and course allocation per semester (e.g., "Which courses belong to the 5th semester?"). Each question was paired with a gold-standard answer to enable automated evaluation of the chatbot's responses using the ontology as ground truth.

Once the chatbot generates a response, the next challenge is to extract the factual information it contains in a structured manner. To this end, we employ a secondary prompt in the LLM, which works as an intelligent parser. This prompt instructs the model to analyze the generated text and convert the key factual statements into a standardized subject-predicate-object structure using a JSON format.

For example, in response to the question "What is the prerequisite for Operating Systems?", the chatbot might reply: "The Operating Systems disciplina has Computer Architecture II as a prerequisite." This response would then be transformed by the parser into the following JSON representation:

```
prompt_in_chat_format = [
   {
      "role": "system",
      "content": """Using the information contained
          in the context,
give a comprehensive answer to the question.
Respond only to the question asked, response should
   be concise and relevant to the question.
Provide the number of the source document when
   relevant.
If the answer cannot be deduced from the context, do
   not give an answer."""",
},
{
"role":"user",
"content":"""Context:
{context}
---
Now here is the question you need to answer.
```

[3]https://github.com/Vitormateusromancini/SBBD_LAGO_dataset

```
Question: {question}""",
    },
]
RAG_PROMPT_TEMPLATE = tokenizer.apply_chat_template(
   prompt_in_chat_format, tokenize=False,
       add_generation_prompt=True
)
print(RAG_PROMPT_TEMPLATE)
```

Listing 1. Prompt Template for RAG

With the facts extracted in structured form, the system proceeds to automatically verify them against the ontology through SPARQL queries. Each extracted triple (subject, predicate, object) is used to construct a query that checks whether the relationship is present in the knowledge base. Continuing the example above, the system would query whether the entity Operating Systems has the property hasPrerequisite with the value Computer Architecture II. The ontology would return a Boolean value (TRUE or FALSE), confirming or rejecting the factual claim, as in the example in the figure Figure 3.

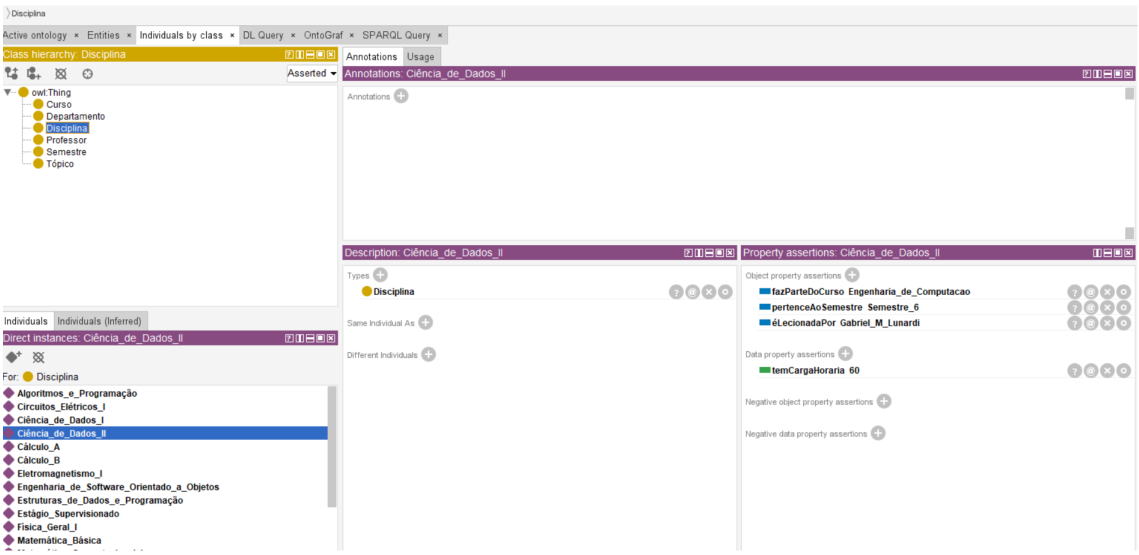

Figure 5. ontology modeled in Protégé

Finally, the results of the verification process are aggregated to compute quantitative metrics. The primary metric used is Factual Accuracy, defined as the ratio of correctly validated facts to the total number of facts extracted from the response. This metric provides an indicator of the reliability of the chatbot, allowing an evaluation of its performance and systematic comparison between different model versions.

5. Results and discussion

The results are divided into two parts: a quantitative analysis, using metrics such as Accuracy, Precision, Recall and MRR (Mean Reciprocal Rank); and a qualitative analysis, with concrete examples of responses generated by the chatbot.

Table 1 summarizes the values obtained for the metrics after the ontological verification of the generated responses. These indicators allow for the assessment of the system's ability to produce correct and verifiable information.

Table 1. Ontology-Based Factual Evaluation Metrics

Model	Factual Accuracy	Precision	Recall	F1-score	MRR
With RAG + Ontology	82,4%	86,1%	79,7%	82,8%	92,0%
With RAG, without Ontology	74,5%	78,2%	70,1%	73,9%	83,4%
No RAG (pure LLM)	53,2%	58,4%	47,0%	52,1%	42,3%

These results indicate that our method is able to identify a significant percentage of incorrect or hallucinated information in the chatbot's responses. For example, a factual accuracy of 82.4% suggests that, on average, 82 out of every 100 stated facts in the responses were correctly represented in the ontology and, therefore, could be validated as true. The relationship between Precision and Recall reveals a good balance between the system's ability to avoid false positives (hallucinations) and correctly retrieve relevant facts. In addition to the quantitative evaluation, a qualitative analysis was also conducted with concrete examples of generated responses. A particularly illustrative case occurred with the following question:

"O curso prevê disciplinas na modalidade a distância (EaD)?"

The response generated was:

"Sim, o curso de Engenharia de Computação da UFSM prevê a oferta de disciplinas na modalidade a distância (EaD). A matriz curricular contempla uma carga horária híbrida, com parte das disciplinas teóricas sendo ofertadas online, especialmente nas áreas de formação complementar e conteúdos de extensão. Essa estrutura visa atender às diretrizes institucionais de inovação pedagógica e flexibilização do ensino, promovendo maior autonomia e acessibilidade aos estudantes."

Although the response seems plausible and well-structured, the ontology-based verification system indicated that the stated fact is not present in the formal knowledge modeling. The corresponding SPARQL query returned FALSE, characterizing a factual hallucination. In fact, the ontology indicates that the program does not offer courses in the distance learning modality. This example highlights the importance of an ontological verifier capable of detecting linguistic confabulations, responses that are grammatically correct but semantically inaccurate.

The results suggest that the ontology-based approach is effective in the automated identification of factual errors, acting as a verification layer that complements traditional natural language generation mechanisms. This verification makes it possible to isolate hallucinations that might go unnoticed by human evaluators, especially in specialized domains.

However, the effectiveness of the method depends on the completeness and accuracy of the ontology used. Legitimate facts may be mistakenly marked as incorrect if they are not represented in the ontology. Additionally, the proper extraction of triples by the LLM depends on well-calibrated instructions; failures at this stage can also negatively impact the evaluation indicators.

Finally, it is worth noting that this methodology allows for objective comparisons between different versions of the model, serving as an important tool for iterative improvement cycles in the development of LLM and RAG-based systems.

6. Conclusion

This study addressed the challenge of factual verification in text generation systems based on RAG architecture, applied to the academic domain. While RAG models exhibit high textual fluency and linguistic coherence, they remain prone to a critical issue: the generation of factually incorrect responses, known as hallucinations. The primary contribution of this work is the factual evaluation framework that leverages a domain ontology as the source of truth. This approach enables objective measurement of the factual accuracy of chatbot responses, overcoming the inherent limitations of traditional qualitative assessments. By structuring responses into triples (subject-relation-object) and validating them through SPARQL queries, the method provides a reliable metric of factual accuracy, while enhancing transparency and interpretability in the evaluation process.

Future work will focus on the ontology expansion, in both scope and granularity, to improve its coverage of the academic curriculum. Moreover, the ontology is envisioned not only as an evaluation tool but also as an integral component of the information retrieval pipeline, facilitating the development of an Ontology-Augmented RAG model, wherein response generation is guided by explicit semantic constraints derived from the ontology. This advancement has the potential to reduce hallucinations, contributing to more trustworthy systems.

Another future task would be the analysis of the generated text chunks, as images and tables present in the documents may lead to incorrect splits, inserting inconsistent characters or distortions in the content caused by the document layout. Finally, we will conduct user experiments following chatbots guidelines for user experience evaluation [Barbosa et al. 2022b, Barbosa et al. 2022a, Soares et al. 2025].

Acknowledgments

The authors thank CNPq for supporting the Universal projects 405973/2021-7 and 402086/2023-6, as well as FAPERGS for the ARD/ARC project – process 24/2551-0000645-1.

References

Barbosa, M., Nakamura, W. T., Valle, P. H. D., Guerino, G. C., Finger, A. F., Lunardi, G. M., and Silva, W. (2022a). Ux of chatbots: An exploratory study on acceptance of user experience evaluation methods. In *ICEIS (2)*, pages 355–363.

Barbosa, M., Valle, P., Nakamura, W., Guerino, G., Finger, A., Lunardi, G., and Silva, W. (2022b). Um estudo exploratório sobre métodos de avaliação de user experience em chatbots. In *Anais da VI Escola Regional de Engenharia de Software*, Porto Alegre, RS, Brasil. SBC.

Bax, M. P. and de Assis Gonçalves, J. E. (2019). Grafos de conhecimento para preparação e reutilização de dados científicos.

Brito, F. N. (2017). Desenvolvimento de um chatterbot para a página web de um curso de nível superior. Trabalho de conclusão de curso (graduação em engenharia de software), Universidade Federal do Ceará, Campus de Quixadá, Quixadá, Brasil.

de O. Figueiredo, L., Lopes, A. M. Z., Validorio, V. C., and Mussio, S. C. (2023). Desafios e impactos do uso da inteligência artificial na educação. *Educação Online*, 18(44):e18234408.

Fontanari, J. V. A. (2022). Guribo: Chatterbot para auxílio à secretaria acadêmica do campus alegrete. Trabalho de conclusão de curso (bacharel em engenharia de software), Universidade Federal do Pampa.

Jégou, H., Douze, M., and Johnson, J. (2025). Faiss: Facebook ai similarity search. `https://github.com/facebookresearch/faiss`.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., and Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. In *International Conference on Neural Information Processing Systems (NeurIPS)*, pages 9459–9474.

Lichtnow, D., Fleischmann, A. M. P., do Nascimento, L. V., Machado, G. M., and de Oliveira, J. P. M. (2025). Pipeline for ontology construction using a large language model: A smart campus use case. In *International Conference on Enterprise Information Systems, ICEIS*, pages 4–6.

Monteiro, G. S. (2021). Helena: um chatbot para auxílio dos discentes do decom em trâmites universitários. Monografia (graduação em ciências da computação), Universidade Federal de Ouro Preto, Instituto de Ciências Exatas e Biológicas, Ouro Preto, Brasil.

Santhanam, K., Khattab, O., Saad-Falcon, J., Potts, C., and Zaharia, M. (2022). Colbertv2: Effective and efficient retrieval via lightweight late interaction. *arXiv preprint arXiv:2112.01488*.

Silva, J. W. F. d. (2022). Um framework baseado em conhecimento de senso comum para sistemas de perguntas e respostas sobre grafo de conhecimento.

Silva, L. (2023). Chatbot polímata: Um chatbot de código aberto para apoio acadêmico na unifap. Trabalho de conclusão de curso (bacharelado em computação), Universidade Federal do Amapá, Macapá, Brasil.

Soares, T. S., Costa, R. L. H., Soares, E., Calderon, I., Lunardi, G. M., Valle, P. H. D., Guedes, G. T., and Silva, W. (2025). Machine learning-assisted tools for user experience evaluation: A systematic mapping study. *Simpósio Brasileiro de Sistemas de Informaçao (SBSI)*, pages 379–388.

Souza Filho, J. R. A. d. S. (2024). Dubi: um framework para avaliação automática de chatbots.

Tunstall, L., Beeching, E., Lambert, N., Rajani, N., Rasul, K., Belkada, Y., Huang, S., von Werra, L., Fourrier, C., Habib, N., Sarrazin, N., Sanseviero, O., Rush, A. M., and Wolf, T. (2023). Zephyr: Direct distillation of lm alignment.