SHORYA SETHIA

22B2725

Engineering Physics

Hallucinations in InstiGPT

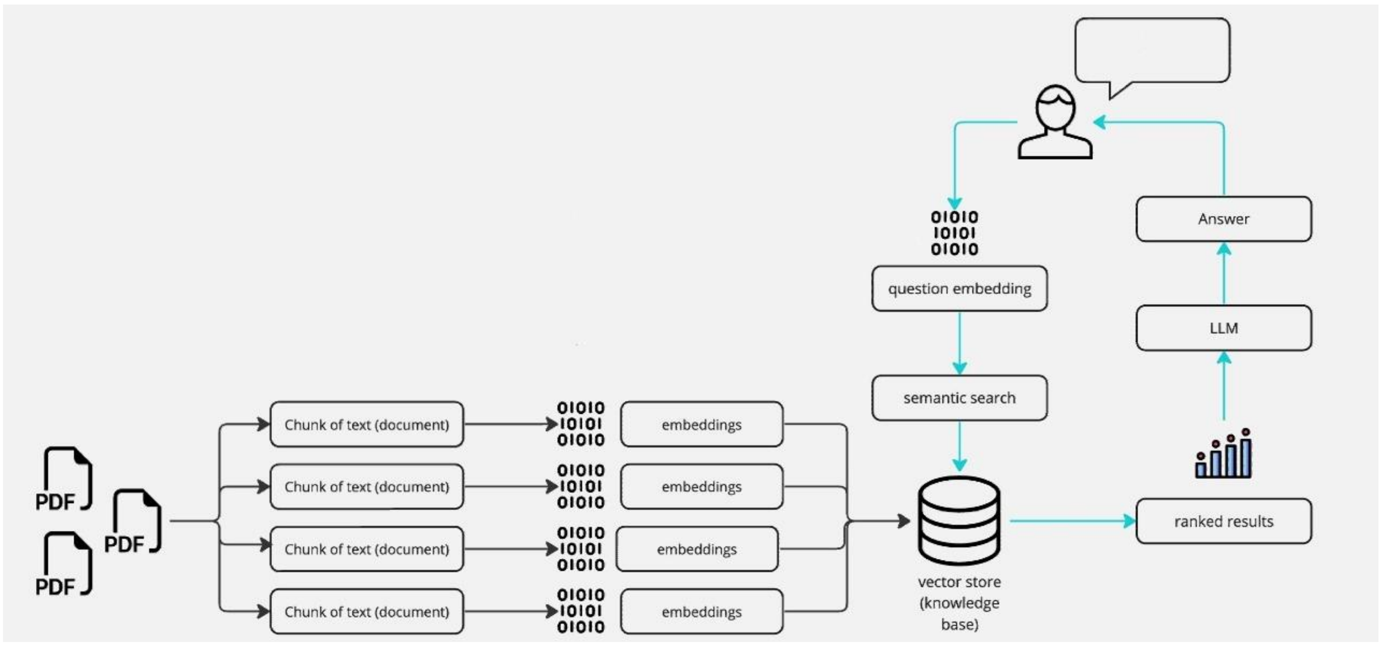
Abstract

Large language models (LLMs) have witnessed significant advancements in generating coherent, intelligent, and contextually relevant responses. However, the presence of hallucinations – inaccurate or unmotivated claims – remains a persistent challenge, motivating the development of automated metrics for the detection of hallucinations in LLM outputs.

Hallucinations are possibly the single largest impediment to widespread practical use of LLMs. This fact means there is a pressing need to identify ways of automatically discovering hallucinations in LLM outputs.

InstiGPT

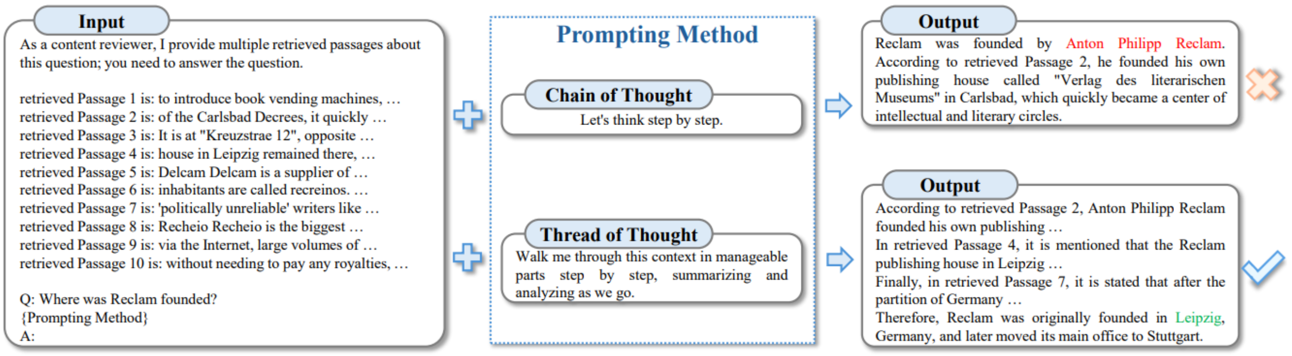
RAG based LLM chatbot



InstiGPT obviously aimed to provide it users with contextually relevant and accurate information retrieved from trained datasets, internal databases and external platforms like Google. InstiGPT has encountered challenges related to hallucinations, wherein the chatbot generates garbage text that lacks factual accuracy or coherence. This issue is not inherent to the chatbot itself but rather it arises from inaccuracies within the data retrieval process (maybe sometimes from Top-5/10 Google results used) or embedding errors or chunk overlaps or overfitting / underfitting or data inconsistencies etc.

Strategies to address Hallucinations could be:

1. Thread of Thought (ThoT)



As a content reviewer, I provide multiple retrieved passages about this question; you need to answer the question.

retrieved Passage 1 is: to introduce book vending machines, …
retrieved Passage 2 is: of the Carlsbad Decrees, it quickly …
retrieved Passage 3 is: It is at "Kreuzstrae 12", opposite …
retrieved Passage 4 is: house in Leipzig remained there, …
retrieved Passage 5 is: Delcam Delcam is a supplier of …
retrieved Passage 6 is: inhabitants are called recreinos. …
retrieved Passage 7 is: 'politically unreliable' writers like …
retrieved Passage 8 is: Recheio Recheio is the biggest …
retrieved Passage 9 is: via the Internet, large volumes of …
retrieved Passage 10 is: without needing to pay any royalties, …

Q: Where was Reclam founded?
{Prompting Method}
A:

Let's think step by step.

Walk me through this context in manageable parts step by step, summarizing and analyzing as we go.

Reclam was founded by Anton Philipp Reclam. According to retrieved Passage 2, he founded his own publishing house called "Verlag des literarischen Museums" in Carlsbad, which quickly became a center of intellectual and literary circles.

According to retrieved Passage 2, Anton Philipp Reclam founded his own publishing …
In retrieved Passage 4, it is mentioned that the Reclam publishing house in Leipzig …
Finally, in retrieved Passage 7, it is stated that after the partition of Germany …
Therefore, Reclam was originally founded in Leipzig, Germany, and later moved its main office to Stuttgart.

Figure 1: Thread of Thought prompting enables large language models to tackle chaotic context problems. In the output depicted, green text denotes the correct answer, while red text indicates the erroneous prediction.

It is inspired by human cognitive processes, which significantly improves the ability of LLMs to segment and analyse extended contexts. ThoT not only improves LLMs' performance in chaotic contexts but also enhances their reasoning

abilities. Basically, it skilfully breaks down and analyses extensive contexts while selectively choosing relevant information.

- **Query Optimization :** Implement techniques (query rewriting, query expansion, and relevance feedback) to improve search efficiency and relevance and refined user queries can enhance the quality of search results and retrieval process. Use this particularly when the model access Google API.

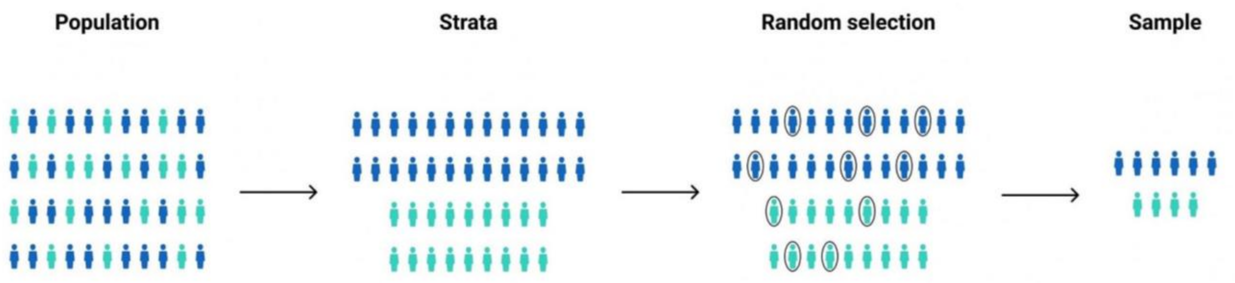
2. For Large Datasets

When working with a large corpus of documents, conventional retrieval methods may fail to prioritize relevant information effectively. As the dataset size increases, the probability of relevant chunks being overlooked becomes more and more considerable, leading to potential hallucination occurrences. Simply increasing the top-n retrieval threshold is not a sustainable solution and may introduce inefficiencies elsewhere in the system.

- **Maintain High Quality Embeddings**

  Information can be categorized as either existing in continuous streams or in discrete chunks. LLMs dealing with large datasets, such as Bert, GPT, and others, exemplify continuous knowledge representation. Since these embedding vectors are continuous, they permit an infinite range of values within specified dimensions. This continuity results in a certain 'fuzziness' in

the embeddings' coordinates, allowing for nuanced and context-sensitive interpretation of word meanings.
Hence, high-quality embeddings are crucial for ensuring the fidelity of retrieved information.
Word Embedding Techniques : Count-based methods like TF-IDF(emphasis on words unique to document) and BM25(probabilistic modelling, considering document length and term saturation) focus on word frequency and document uniqueness, offering basic information retrieval capabilities. Co-occurrence based techniques such as Word2Vec, GloVe, and fastText analyse how words co-occur in large corpora, capturing semantic relationships and morphological details.
Stratified Embedding Search : Check do we have a broad corpus spanning a variety of topics? For large datasets, off course True, consider splitting them up, and apply a stratified embedding search, where we have embedding indices, but they're categorized by topics spanned in our corpus. Each topic has a topic description that is itself embedded - so first find the relevant topic(s), and then look inside those topics to find the actual documents needed.



Also, if corpus include documents or data in multiple languages or formats, consider harmonizing language and style before storing embeddings.

- ## Implementing a Thresholding System

  Dynamic thresholding mechanisms adapt document retrieval based on dataset size and query complexity. Machine learning techniques optimize retrieval, ensuring the model receives contextually relevant information.
  Or,
  Implement a confidence score mechanism to assess the relevance of the generated response to the query and the provided content. If the score is below a certain threshold, default to "Sorry, I am unable to answer your query."

- ## Chunk Re-listing

  Chunk relisting involves re-evaluating retrieved chunks to prioritize relevant information. By relisting chunks based on contextual relevance, this approach reduces the probability of relevant content being overlooked.

Consequently, hallucination occurrences are minimized as the model receives more focused and pertinent data inputs.

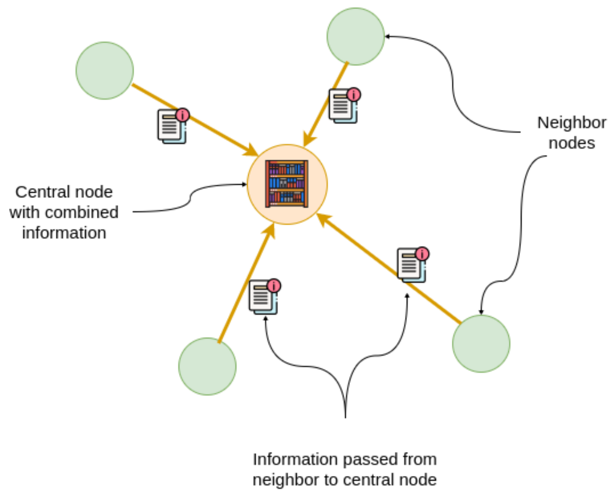
- **Pre-processing Data and Summarization Techniques**
  Pre-processing techniques, such as extractive summarization and MAMBA processing with SSM6, condense large text volumes while preserving context. By summarizing information without losing context, these techniques mitigate hallucination risks associated with information overload.

3. Changes in Context Window

Maintain the chat history, but don't send it to the model (with the question) – instead, use it to create a standalone question which is to be send to the model along with the context documents, this would reduce memory usages and reduces hallucinations significantly.

4. Creating Knowledge Graphs

Integrating a graph database management system (Neo4j) parallelly with traditional vector databases can mitigate hallucinations by leveraging graph-based relationships for semantic validation and contextual understanding. The rules we want to be taken care of are enforced and are stored in the Neo4j database.
We can also set it up where it only queries the knowledge graph for problematic questions.
Besides, knowledge graphs usage parallel to vector embeddings, we can parallelize search operations across multiple processing units or nodes to improve scalability and performance. Apache Solr or Elasticsearch, can be deployed to distribute search queries and index updates across a cluster of machines.


Neighbor nodes
Central node with combined information
Information passed from neighbor to central node

5. Overdependence on retrieved documents

Response generation becomes particularly difficult when retrieving documents featuring conflicting data. The model must determine which information is credible or relevant, despite contradictions.

For example, InstiGPT hallucinates in telling branch changer's present department, for me it says 4 yrs BS Chemistry but I branch changed to Engineering Physics.

6. Use metadata to trace chunk location

These tags assist in prioritizing chunks during retrieval operations by incorporates relevance tagging, wherein tags or labels are assigned to each chunk based on its importance, relevance to user queries, or thematic significance taken from context window.
But for large unlabelled data, adding labels or metadata involves

- Named Entity Recognition (**NER**) models, trained on annotated datasets or pre-trained on large corpora, can automatically identify and tag location mentions within text.
- Keyword matching
- **Human Annotations**: Consider manual annotation by human annotators to label a subset of the unlabelled data with location information. Use this annotated subset as a training dataset to develop and refine automated labelling algorithms such as classifiers or sequence labelling models.

7. RHFL (Response Hierarchy with Feedback Loop)

Implementing a user-feedback-driven approach is a fundamental strategy for refining the capabilities of chatbots like ChatGPT-3.5, through feedback mechanisms, such as response selection, users can identify deviations from their expectations or inaccuracies in responses. And model utilizes this feedback to identify and rectify errors in its training data or generation algorithms, thereby enhancing the accuracy and reliability of its responses.

By analysing patterns in feedback, developers can identify areas of improvement. Furthermore, model can adapt to individual user preferences and provide a personalized experience. This might enhance user satisfaction and engagement with the InstiGPT.

8. Mixture of Experts (MoE)

9. It has been proven that using multiple agents to check each other reduces hallucinations and corrects mistakes. Therefore, employing a secondary, more cost-effective model for initial evaluation and subsequently passing filtered results to the primary LLM, facilitates a robust fact-checking mechanisms. This methodology not only enhances the efficiency of the process but also mitigates the occurrence of hallucinations and rectifies errors.

10. Incorporating a Discriminator

InstiGPT serves as the "generator" and aims to produce coherent responses to user queries. A hallucination detector acts as the "discriminator" and assesses the accuracy of InstiGPT responses.
InstiGPT and the hallucination detector are trained simultaneously in a competitive manner. The hallucination detector's objective is to accurately identify hallucinated or inaccurate responses generated by InstiGPT.
The hallucination detector provides feedback on the accuracy and coherence of responses. InstiGPT adjusts its parameters based on the feedback to improve response quality and reduce hallucinations.

11. Continuous Data Updating

Given the current infrastructure limitations of not having our own container and ITC container gets filling but incorporating new data for continual learning while utilizing a virtual machine is also a necessity, so following approaches may be useful;

- Implement a data sampling and filtering mechanism to prioritize relevant or high-value data for continual learning
- Instead of processing data in real-time, adopt a batch processing approach where data is collected and processed in batches at regular intervals. This allows for more efficient resource utilization and reduces the overhead associated with continual data updates.
  Schedule batch processing jobs during off-peak hours to avoid resource contention and optimize system performance.
- Incremental learning algorithms (Online Learning) that can update existing model with new data without retraining from scratch. These algorithms incrementally incorporate new data into the existing model, reducing the computational cost down by 50 to 90%.
- Implement caching mechanisms to store previously computed embeddings and reuse them when processing similar data. This reduces redundant computations and accelerates the embedding generation process.
  Using cache embeddings at various levels of granularity (e.g., document-level, paragraph-level, or sentence-level) helps to optimize retrieval and reduce latency during inference.
  Also, caching frequently accessed documents or query results reduces the need for redundant computations and improves response times.

Moreover, re-training pipelines should be in place to ensure that the model continues to perform well on new and unseen data.

12. RAG vs Fine-Tuning

RAG may be more cost-effective than fine-tuning an LLM, as powerful computes are required for fine-tuning and hosting. The charges are applied based on the size of data being trained, training hours and /or charges for the hosting hours. But the accuracy of the RAG approach depends on the quality and relevance of the data sources. Poorly curated data can lead to inaccurate or irrelevant responses.
Rather we should opt for fine tuning particularly when the objective is to do tasks that require a level of specificity and customization that general models may not readily provide, where the guiding information is too voluminous or intricate to be encapsulated within a single prompt, and this is not in case of InstiGPT, so we could avoid fine tuning it.
Hosting cost is integral for both the methods but the data acquisition cost required in RAG is comparatively less than training cost and compute power required in fine-tuning LLM.

13. Cost Management & Resource Allocation

Not having our own AIC container and relying on the ITC container, along with the shift towards using virtual machines;

- **Compute Resources:**
  Virtual Machine Type: Standard_D2s_v3 should selected based on its balanced configuration of 2 vCPUs and 8 GB RAM, which is suitable for running InstiGPT inference tasks efficiently.
  Hourly Rate: The hourly rate of $0.043 can be chosen as a reasonable estimate based on the pricing structure of virtual machines offered by cloud providers like Azure.
  Usage Duration: Assuming 24/7 usage for a month (720 hours) to account for continuous availability of InstiGPT to users.
  Compute Cost per month: $0.043 * 720 = $30.96 = ₹ 2600 (approx)
- **Data Storage:**
  Cost of data storage within the ITC container based on the anticipated storage requirements.
  Storage Type: Object Storage (Azure Blob Storage) can be selected as it provides scalable and cost-effective storage for the documents and data required by InstiGPT.
  Storage Capacity: An estimated storage capacity of 100 GB chosen based on the size of the dataset (13-17 PDFs of not more than 50 text pages each). This allows for ample storage space to accommodate future growth.

Monthly Storage Cost: The cost per GB per month was estimated at $0.02, which is a common pricing tier for object storage services.
Data Storage Cost per Month: $0.02 * 100 = $2 = ₹ 170 (approx)

- **Model Deployment:**
  Hosting Platform: Gymkhana servers would be similar to Azure Kubernetes Service (AKS). AKS offers scalable and managed Kubernetes clusters for deploying containerized applications.
  Deployment Cost: The hourly rate of $0.10 for the AKS.
  Average Deployment Duration: Assuming continuous deployment for a month (720 hours) to ensure uninterrupted availability.
  Deployment Cost per Month: $0.10 * 720 = $72 = ₹ 6000 (approx)
- **Total Monthly Costings = 2600 + 170 + 6000 = ₹ 8770**
  Considering fluctuations and some errors, total costings for an month would be around **₹10,000**.

Sources and References

- https://www.linkedin.com/feed/update/urn:li:activity:7179865689959079936/?utm_source=share&utm_medium=member_android
- https://arxiv.org/pdf/2109.07958.pdf
- https://arxiv.org/pdf/2311.08734v1.pdf
- https://www.genoshi.io/
- https://techcommunity.microsoft.com/t5/ai-azure-ai-services-blog/cost-optimized-hosting-of-fine-tuned-llms-in-production/ba-p/4062192
- https://azure.microsoft.com/en-in/pricing/details/kubernetes-service/
- https://aman.ai/primers/ai/word-vectors/
- https://youtu.be/B5B4fF95J9s?si=sdRtfuepkREpT3rw
- https://www.linkedin.com/pulse/why-does-gpt-hallucinate-martin-treiber/