The Thesis Committee for Utkarsh Mujumdar
certifies that this is the approved version of the following thesis:

Designing a Multi-Perspective Search System Using Large Language Models and Retrieval Augmented Generation

SUPERVISING COMMITTEE:

Matthew Lease, Supervisor

Ashwin Rajadesingan, Second Reader

Designing a Multi-Perspective Search System Using Large Language Models and Retrieval Augmented Generation

by

Utkarsh Mujumdar

Thesis

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Information Studies

The University of Texas at Austin

May 2024

Acknowledgments

I would like to express my deep and sincere gratitute to my supervisor, Dr. Matt Lease for giving me this opportunity and providing me with his guidance and expertise throughout the duration of my thesis. Special thanks to Dr. Ashwin Rajadesingan, the second reader, for providing his valuable inputs and identifying key improvements that have made this final work possible.

I owe a great deal of appreciation for the undergraduate and graduate students I've had the chance to work with while working on this project. Special mentions for Yian Wong, Li Shi and Houjiang Liu for their inputs and constructive feedback that enabled me to execute the project successfully.

It wouldn't have been possible for me to reach this stage without the help of the UT Austin research ecosystem, and the School of Information's resources that have given me the opportunity to develop my fundamentals and grow my research aptitude during the last two years of my studies.

Last, but not the least I would like to thank my emotional ecosystem consisting of my parents, family, my partner Keerat and my close friends who have constantly supported me through the ups and downs in the past year and enabled me to do my best work possible.

Abstract

Designing a Multi-Perspective Search System Using Large Language Models and Retrieval Augmented Generation

Utkarsh Mujumdar, MSIS
The University of Texas at Austin, 2024

SUPERVISOR: Matthew Lease

In the context of information retrieval, multi-perspective search is a desired solution when the search query focuses on contentious topics that might not have clear factual grounds for an answer - "Should humans colonize space" being an example of such a search query. Although the explicit intent of this query might require a definitive answer ("Yes"/"No"), an ideal search result of the query should add the necessary context, or *perspective* along with the definitive answer. Added to this is the facet that there can be multiple such *perspectives* that can be used to answer the question, and hence the need for multi-perspective search systems. However, seeking diverse perspectives in information-seeking contexts is a challenging problem to solve - traditional search engines, while effective in aggregating data, often fall short in providing a cohesive context, particularly when addressing complex, contentious topics.

Motivated by these shortcomings, this thesis introduces a multi-perspective search system that leverages the capabilities of Large Language Models (LLMs) and Retrieval Augmented Generation (RAG). Given a search topic of interest, the proposed system employs LLMs to generate and embody diverse personas that represent

different perspectives on the topic of interest. Each persona then presents their perspective as part of a simulated debate format, resembling a hypothetical discussion between the different stakeholders. Retrieval Augmented Generation is employed to provide substantiating evidence as part of each argument presented in the debate. This innovative approach allows users to explore a topic through a dialogue that synthesizes multiple perspectives, offering a richer and more nuanced understanding of the topic of interest. The system is designed with a user interface that supports this complex interaction, making it accessible and engaging for users. The development of this system not only advances the field of multi-perspective search but also opens new avenues for potential applications in conversational interfaces, decision-making support systems, and online discussions on digital platforms. This thesis discusses the motivation for multi-perspective search systems, conceptualization of the proposed approach, the design of the system's interface and architecture, implementation challenges, and potential use-cases of the proposed system, setting a robust foundation for future enhancements and wider application.

Table of Contents

Chapter 1: Introduction

The advent of Large Language Models in the past two years is changing the way we interact with and consume information today. Tools like OpenAI's ChatGPT and Microsoft's Copilot are enabling users to search for information in a more accessible way by integrating a conversational flow to present information to the users. However, these technologies are still nascent when compared to traditional search systems such as search engines that are used widely to search for information online. These traditional systems present isolated pieces of information without tying them together under a unified context, and thus can be hard to digest - especially when the search topic is contentious and can have multiple perspectives or viewpoints.

This work is motivated by the need for a system that can provide users with information in a conversational flow similar to LLM-based tools while also preserving the underlying context of information related to multi-faceted search topics. The system that we have developed enables users to view information from different perspectives through personas that engage in a simulated debate focussed on the search topic of interest. These personas can be representative of the stakeholders or interest groups relevant to the topic of interest, or be custom-defined based on the user's input. A demonstrated example of the system can be seen in Fig. 1.1, with the topic of interest being "Should animals be used for scientific or commercial testing?".

At a high-level, the system works as follows. The user enters a topic of interest that they want to know more about, and an LLM agent is then tasked with creating personas relevant to the topic. This is followed by assigning of different personas to separate individual LLM agents that are prompted with the topic of interest combined with all the perspectives presented by other agents as context, in order to generate their own perspective. All the generated outputs are rendered as part of a user interface as shown in Fig. 1.1.

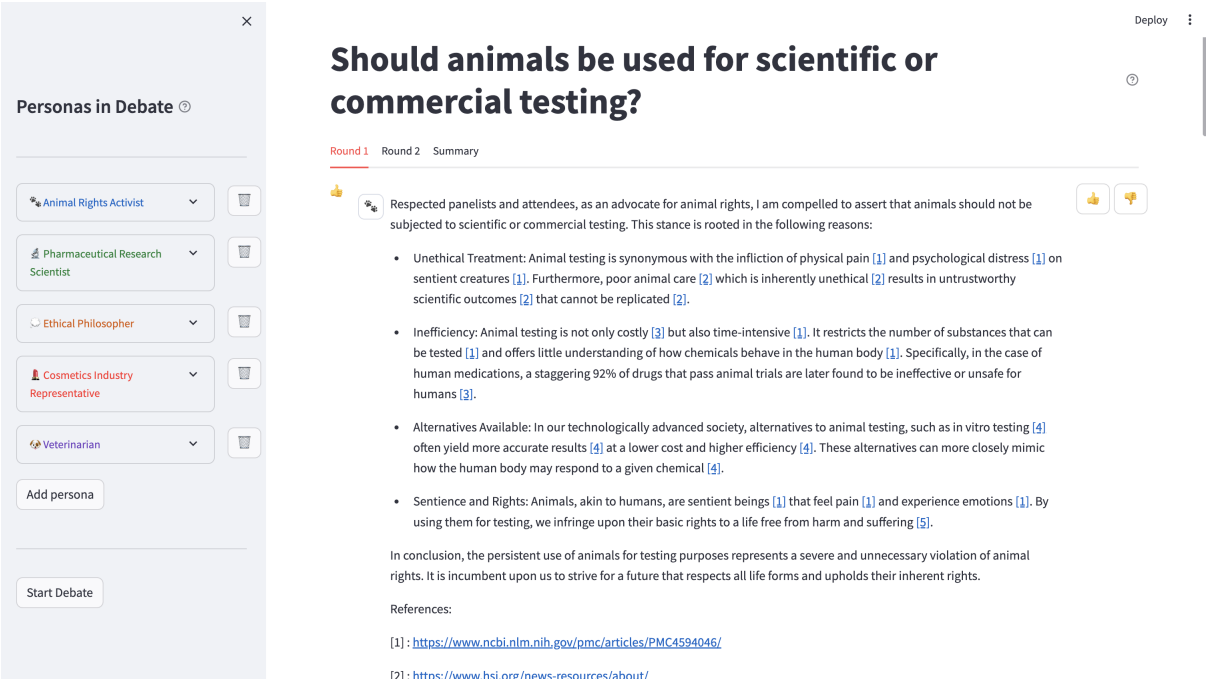


Figure 1.1: Interface showing the output when the user searches for the topic "Should animals be used for scientific or commercial testing?"

Application use cases for such a system can be varied, such as debating public controversies, tools for personal decision making and analyzing the different aspects of ethical quandaries. While the current prototype of the system has been developed as a separate tool in itself, these use cases can also be made possible in the form of integrations in existing systems, with our system acting as an additional layer on top of them.

The main contributions of this study are highlighted below:

- Creation of a Large Language Model (LLM) based framework to generate multiple perspectives in the case of contentious or ambivalent topics of interest through LLM agents acting as different personas
- Integration of an evidence retrieval module based on Retrieval Augmented Generation (RAG) that can provide linked evidence supporting the generated per-

spectives

- Systemic implementation of the framework that can be integrated with a front-end interface that enables users to engage with it

The remainder of the thesis is organized as follows. In Chapter 2, relevant prior work relevant to the scope of this project is presented. Chapter 3 highlights the survey of similar tools and the design goals that were identified for the system. Implementation specifics such as the interface design, system architecture and the challenges encountered during development are discussed in Chapter 4. Relevant results, application use cases, limitations and future scope of research are addressed in Chapter 5 with an appropriate conclusion in Chapter 6.

Chapter 2: Related Work

2.1 Large Language Models: Overview

Large Language Models (LLMs) have been instrumental in advancing the field of natural language processing (NLP), enabling significant progress in how machines understand and generate human language. These prevalent models, which are based on the transformer architecture (Vaswani et al. (2023)), are trained on large-scale data sets collected from the internet. They have become increasingly pivotal in solving natural language tasks that require the generation of coherent and contextually relevant text, showcasing a deep understanding of language patterns.

The underlying methodology guiding these models is a self-supervised learning goal aimed at predicting the subsequent words (tokens) given the context of a sequence of tokens. They are able to learn to produce logical and context-sensitive answers by way of a training regime that involves learning the common associations of words and phrases in a given language. They are trained to produce a natural language output as a response to a natural language instruction, also termed as a "prompt" that contains a task instruction and might also contain additional context like example outputs for the task.

The quality of the output generated by LLMs is governed by several elements, such as the initial prompt, the model's specific parameters, and the diversity of training data used to initially train the model. While the GPT Series of models (GPT-3.5, GPT-4) by OpenAI have been the most widely used by both users and researchers alike, there have been a number of other models developed in recent times that can achieve comparable results on relevant natural language task metrics. These include Meta's LLaMa2, Google's BARD (now renamed Gemini) and Anthropic's Claude, among others.

While these models have driven substantial progress, they are not without

challenges. The issues of bias, fairness, and ethical use are increasingly scrutinized, as reflected in discussions by Bender et al. (2021) and Blodgett et al. (2020). They highlight the inherent biases present in the training data of LLMs, which can perpetuate and amplify these biases in their outputs. Addressing these concerns is critical for the responsible development and deployment of LLMs.

2.2 Prompt Engineering Methods: Overview

Instruction strategies for LLMs, called as "prompt engineering" , has become an increasingly growing area of research, as it presents the easiest ways to control the outputs generated by LLMs as compared to altering their underlying model weights or re-training them on additional training data. Modern prompt engineering includes a variety of strategies, extending from basic methods like role-prompting (Shanahan et al. (2023)) to advanced ones like "chain of thought" (Wei et al. (2023)) prompting. This field is evolving drastically, with ongoing research consistently introducing new approaches and uses for prompt engineering. The significance of prompt engineering is underlined by its role in directing the responses of models, enhancing the adaptability and applicability of LLMs across different industries.

The most common prompting techniques in the field are one-shot and few-shot prompting techniques. One-shot prompting is characterized by presenting the model with just a single example from which to learn, whereas few-shot prompting supplies the model with several examples (Logan IV et al. (2022)). The decision to use one-shot or few-shot prompting typically hinges on the complexity of the task at hand and the model's proficiency. The former might be better suited for simpler tasks and smaller models, whereas the latter might be better suited for more complex tasks and larger, advanced models.

The "Chain of Thought" (CoT) prompting approach (Wei et al. (2023)) has emerged as a popular advanced prompting strategy in recent times. CoT prompting entails supplying a model with intermediary reasoning stages to shape its outputs,

which can be achieved using straightforward phrases like "Let's think step by step" or through detailed examples that feature both a query and a sequential reasoning path leading to a resolution. This method not only aids the model in structuring its reasoning process but also enables the users to better understand the logic behind the model's outputs.

More such advanced strategies have been developed to improve the common sense reasoning ability of LLMs, including the "generated knowledge" approach (Liu et al. (2022)) - a technique that leverages the ability of LLMs to generate potentially useful information about a given question or prompt before generating a final response as well as the "tree of thoughts" (ToT) prompting technique (Yao et al. (2023)) that employs a structured approach to guide LLMs in their reasoning by organizing the prompts in a hierarchical manner, akin to a tree structure, that then guiding the problem-solving process of the LLM.

2.3 Multi-perspective Search Systems

The integration of opinion analysis and diverse perspective gathering into information retrieval systems represents a significant evolution in how we address complex, open-ended, and controversial queries. This is highlighted by the drawbacks of traditional search systems, as highlighted by Chen et al. (2022), who go on to discuss the ideas for a multi-perspective search engine and their efforts to extend document retrieval systems to better handle controversial or open-ended questions. Through user surveys and prototype evaluation, their work assesses the utility of delivering diverse viewpoints in response to complex queries, indicating a demand for retrieval systems that synthesize a broad spectrum of perspectives.

Cardie et al. (2003) present an innovative approach to multi-perspective question answering (MPQA), viewing it as an opinion-oriented information extraction task. They introduce an annotation scheme for opinions, and outline an automatic method for constructing opinion-based summaries. Their framework supports various

MPQA tasks by organizing and presenting multiple opinions, providing users with a nuanced understanding of the subject matter.

Chen et al. (2019)'s work focuses on the contamination of information with biases and the necessity for a system that provides substantiated perspectives on contentious issues. By proposing the task of substantiated perspective discovery, the authors aim to compile well-supported viewpoints on various claims, underpinned by a rigorously constructed dataset. This initiative underscores the importance of breadth and evidence in perspective analysis, offering a methodological framework for dissecting contentious topics comprehensively.

The study done by Metzler et al. (2021) critiques the limitations of current question-answering systems and information retrieval models, emphasizing the need for systems that offer domain expertise and evidential support. By integrating ideas from classical information retrieval and advanced language models, the authors advocate for a new generation of systems that combine the breadth of knowledge with deep, evidence-based understanding, addressing users' information needs with expert-like advice.

2.4 Multi-agent LLM frameworks

Research in the fields of Large Language Models and Natural Language Processing has seen a significant shift towards more collaborative and dynamic approaches in order to enhance the capabilities of large language models (LLMs) through multi-agent systems combined with advanced prompting strategies.

Du et al. (2023) introduce an innovative approach where multiple LLM instances engage in rounds of debate, proposing and refining responses, which they demonstrate improves their mathematical and strategic reasoning, and the factual accuracy. This method, likened to a "society of minds," indicates substantial potential for advancing LLMs' understanding and generation capabilities without requiring specialized adjustments for different tasks.

In contrast, Wang et al. (2024) focus on Solo Performance Prompting (SPP) as their prompting strategy, transforming a single LLM into a "cognitive synergist" by simulating multi-turn self-collaboration across various personas. This strategy enhances problem-solving abilities in LLMs by leveraging cognitive synergy, suggesting that engaging diverse personas can significantly reduce errors and boost reasoning performance, particularly in more advanced models like GPT-4.

Liang et al. (2023) introduce the Degeneration-of-Thought (DoT) problem in LLMs as part of their work using a Multi-Agent Debate (MAD) framework. By encouraging divergent thinking through debate, the MAD framework aims to foster deeper contemplation and more nuanced reasoning in LLMs, showcasing its effectiveness in complex reasoning tasks and highlighting the necessity for balanced and adaptive debate dynamics.

Chan et al. (2023) extend the application of multi-agent systems to the field of text evaluation, proposing the ChatEval framework. By emulating human-like collaborative evaluation processes, this multi-agent approach seeks to enhance the accuracy and reliability of LLM assessments, moving closer to human-level evaluation quality in natural language generation tasks.

Sreedhar and Chilton (2024) explore the potential of LLMs in simulating human strategic behavior through game theory applications, specifically the ultimatum game. By comparing single- and multi-agent architectures, the study demonstrates the superior capability of multi-agent LLMs in replicating complex human strategies, underscoring the value of such simulations in strategic planning and policy-making.

Rasal (2024) introduces a novel communication framework employing multiple LLM agents with distinct personas to tackle autonomous problem-solving. This approach underscores the benefits of collaborative agent interaction in enhancing LLMs' adaptability and problem-solving skills, particularly in novel and challenging scenarios.

2.5 Retrieval Augmented Generation: Overview

Retrieval-Augmented Generation (RAG) is an emergent sub-field of Natural Language Processing, focused on enhancing the capabilities of large pre-trained language models (LLMs) by integrating them with an external, non-parametric memory source to improve their performance on knowledge-intensive tasks.

Lewis et al. (2020) discuss the development and application of RAG models that combine a pre-trained sequence-to-sequence model with a dense vector index of Wikipedia, accessed via a neural retriever. This innovative approach allows the model to draw upon an expansive repository of structured knowledge, addressing the limitations of LLMs in accessing and manipulating precise information. By comparing two RAG formulations—one that accesses the same retrieved passages throughout the generation and another that can retrieve different passages at each token—the study showcases how RAG models excel in generating more specific, factual, and diverse content, particularly on open domain question-answering tasks.

Chen et al. (2024) extended the discourse on RAG by introducing a comprehensive evaluation framework, the Retrieval-Augmented Generation Benchmark (RGB), aimed at assessing the impact of RAG across various LLMs. This benchmark is designed to scrutinize four fundamental capabilities crucial for the effectiveness of RAG: noise robustness, negative rejection, information integration, and counterfactual robustness, across English and Chinese languages. Their study managed to identify existing gaps in applying RAG effectively and hinted at the necessity for further innovation to harness the full potential of RAG in enhancing the reliability and informativeness of LLM-generated content.

Shuster et al. (2021) examined the integration of neural-retrieval-in-the-loop architectures within dialogue models to improve their knowledge grounding, aiming to address factual inaccuracies and hallucinations that current state-of-the-art systems exhibit. By incorporating multiple components like retrievers, rankers, and encoder-decoders, the research explored how these models can maintain conversational coher-

ence while improving their ability to reference accurate knowledge, especially in multi-turn dialogue contexts. The study reported advancements in knowledge-grounded conversational tasks, showcasing models that not only excel in open-domain conversations but also effectively generalize beyond their training datasets and minimize knowledge hallucination, as corroborated by human evaluations.

The study by Chang et al. (2024) delved into managing of controversial discussions within LLM-based chatbots by adhering to Wikipedia's Neutral Point of View (NPOV) principle. It introduced a retrieval-augmented generation framework that leveraged multiple perspectives retrieved from a knowledge base. The study identified and addressed common LLM failures such as hallucination and coverage errors, proposing three detection methods based on word overlap, salience, and LLM-based classifiers.

Chapter 3: Design Goals and Benchmarking

3.1 Analysis of Similar Tools

In order to formulate the design and implementation framework of the prototype, we surveyed different tools available online that combine Large Language Model (LLM) generation with an Information Retrieval (IR) capability:

1. **Microsoft Copilot (erstwhile Bing Chat)**: The Copilot tool has been developed by Microsoft as an LLM integration with the search engine Bing (Bing Chat). It has been designed to work in a way similar to a typical search engine where the user is asked to input their search query in a search window. The system then generates an output in conversational English, while also providing citations and references for sources that it borrowed information from (Fig 3.1). Based on our experience while testing this tool, we had two main takeaways:

    - In-line citations: The most useful feature of the tool are the inline citations, which enable the user to hover over a link and navigate to the URL in a separate window. This feature, with its value in terms of transparency and explainability, is an important component in generative experiences and is indicative of the fact that the response generated is grounded in reliable information. It is also a handy way to integrate raw URLs as part of the conversational output.
    - Concurrent generation: While generating the output, the way it is presented makes it seem like the generation is being done concurrently in real-time, and not all in one go. While it may be the latter case in the backend, the concurrent nature at the frontend adds to the conversational effect of the output.

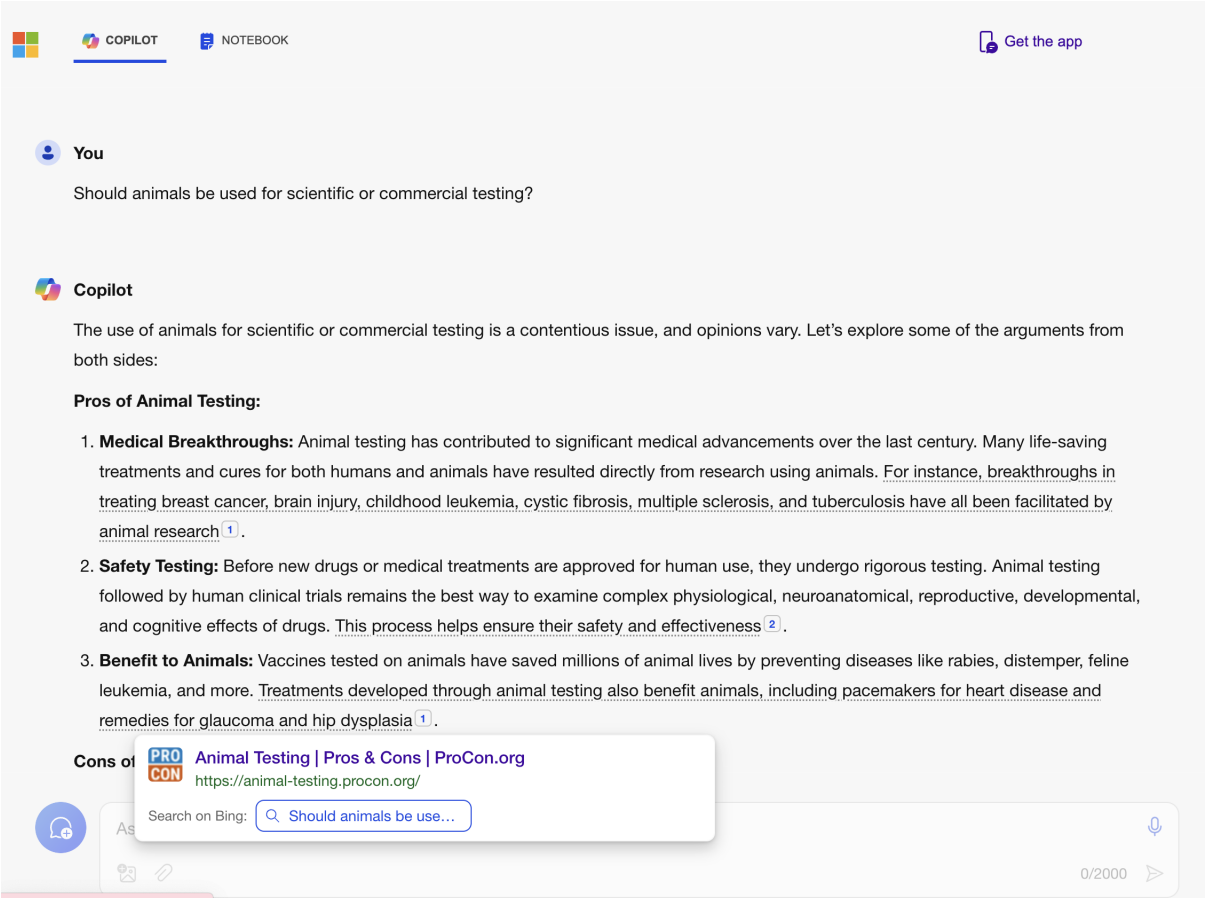


Figure 3.1: Microsoft Copilot's Interface, with the output when the user searches for the topic "Should animals be used for scientific or commercial testing?"

2. **Perplexity**: Perplexity is another AI tool that has been developed with the intention of being an individual's go-to source for all their information discovery and retrieval needs. The interface is akin to that of Microsoft Copilot, with two notable changes:

   (a) The references are displayed at the top of the output instead of at the bottom.

   (b) The generated result has a structured response header that tries to present attributes such as definition for the given search query, followed by the unstructured, conversational output.

The interface for Perplexity is richer than a typical AI chatbot application (Fig. 3.2), with added plugs for image results that are relevant to the search query. Based on our experience while testing this tool, we had two main takeaways:

- Pointwise generation and line-end citations: The generated output is presented in the form of points, which makes the readability a lot better. Also, the citations are presented at the end of each point instead of in-line, which leads to a cleaner interface while sacrificing some element of explainability.
- Non-concurrent generation: As compared to Microsoft Copilot, the generated output isn't presented exactly in a concurrent form. Instead, it has a more static nature to it, with the output appearing on the screen in a much quicker fashion replicating an all-in-one-go approach

3.2 Formulated Design Goals

Keeping in mind our takeaways from researching the design and functional elements of similar AI tools and our multi-perspective search use case scenario, we came up with a list of design goals and rules to guide our development of the prototype:

3.2.1 Clarity in Presentation of Perspectives

The interface should clearly differentiate between the various personas and their perspectives, in order for the system to stay true to its multi-perspective nature.

3.2.2 Presentation of Accessible Evidence

As part of the generated results, there should be an integrated feature that presents evidence supporting the result, as a way for the user to verify the veracity of the result. The relevant evidence should appear in an accessible format that the user can navigate to whenever desired.

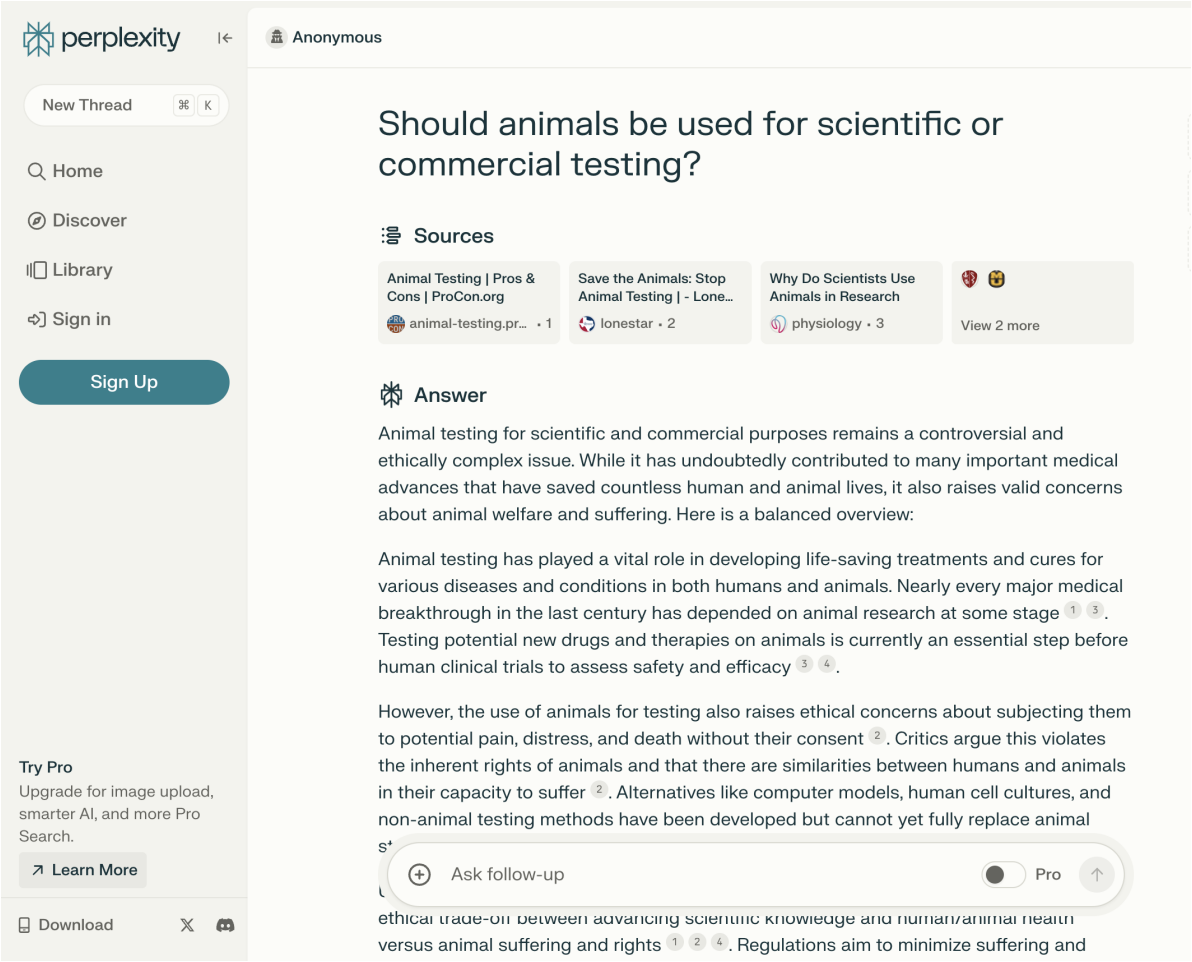


Figure 3.2: Perplexity's Interface, with the output when the user searches for the topic "Should animals be used for scientific or commercial testing?"

3.2.3 Neutrality in Recommendation

The system should strive to be neutral when presenting different perspectives, and avoid prescribing certain perspectives over others. Fairness is an important concern in search systems, especially in the context of multi-perspective search where users are seeking holistic results from varying points of view. Such a system is not meant to be used as a prescriptive tool, and instead should only be used to gain information in an unbiased and holistic manner.

3.2.4 Contextual Continuity

Since the system integrates conversational flow with contextual depth, the design must ensure that users can easily see the connection between different pieces of information and understand how they relate to the overall topic. This could be facilitated through linking mechanisms or threaded discussions.

3.2.5 Engaging and Easy-to-Use Interface

Similar to a traditional search engine, the system should have an easily understandable interface and allow for user engagement through selection of topics, user-guided actions to possibly influence the direction of the debate, and user feedback on generated results. This could be supported through clickable elements, dynamic content updates, and responsive design features.

Chapter 4: System Design and Implementation

4.1 Interface Design

To illustrate the functioning of our proposed system, we will walk-through the different interaction modules for an example scenario, where the user wants to know more about the topic "Should animals be used for scientific or commercial testing?".

4.1.1 Search Toolbar for User Input

The entry point of the interface contains the search bar that facilitates user input for the search query or topic of interest (Fig. 4.1).The toolbar supports two kinds of user inputs:

- Text Input (Fig. 4.2): Topic of Interest entered in plain text by the user
- Drop-down Input (Fig. 4.3): For demonstration purposes, the toolbar contains a drop-down with some pre-selected topics that the user can choose to understand how the tool works

In our example scenario, the topic "Should animals be used for scientific or commercial testing?" is chosen by using the latter method.

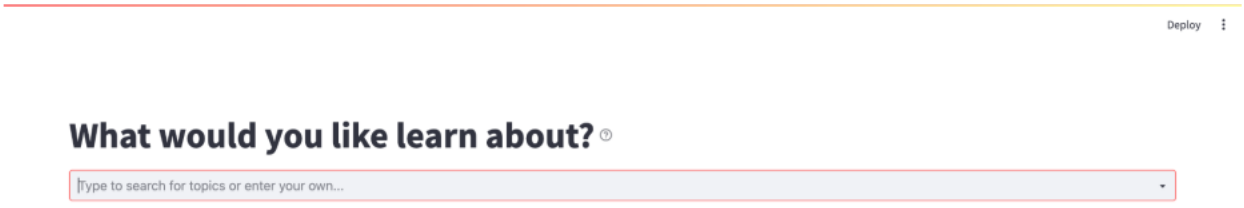


Figure 4.1: Search Bar of the Interface

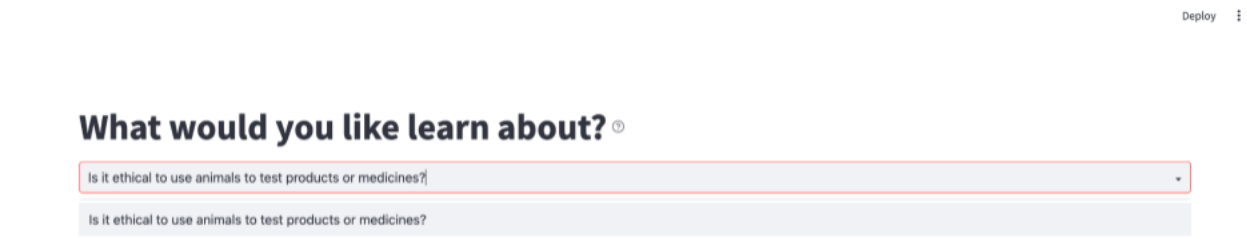


Figure 4.2: User Input via a Typed Entry

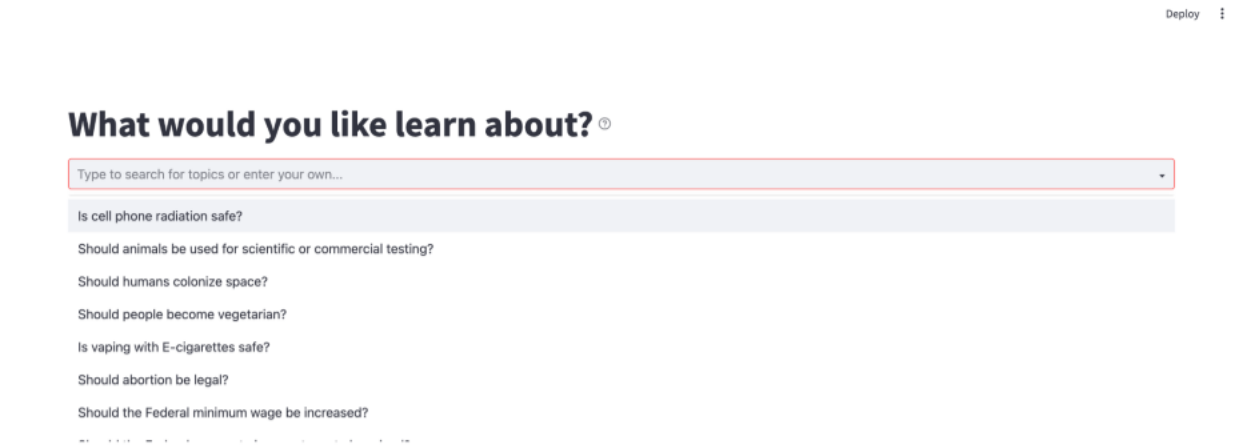


Figure 4.3: User Input via a Dropdown Menu

4.1.2 Persona Sidebar

Once the topic of interest is chosen/entered, the interface transforms into a two-column view. The left-hand side column displays the information regarding the generated personas, with each persona having their own title, description and a representative emoji character (Fig. 4.4). By default, only the title and emoji character of each persona are visible to the user (Fig. 4.5). Each persona title is color-coded with a unique color, and in combination with the emoji character, is intended to help the user differentiate between the personas and their perspectives better - in line with Design Goal 1.

Figure 4.4: Persona Sidebar, as part of the two-column interface

There are five interactive features supported as part of this module:

- **Viewing of persona description** (Fig. 4.5, Point A): Upon clicking the title of any persona, the description of the persona can be viewed by the user. The way the description is presented can be seen in Fig. 4.6a.
- **Deletion of a persona** (Fig. 4.5, Point B): In case not needed, any of the personas can be deleted by the user by clicking on the trash can symbol right next to the persona title.
- **Addition of new personas** (Fig. 4.5, Point C): The user can choose to add new personas to gain additional perspectives on the topic of interest, by clicking on the "Add Persona" button. By default, the system gives the user a pre-generated persona having a defined title and description.
- **Customization of new/existing personas** (Fig. 4.6b): The interface supports modification of existing and newly added personas, by clicking on the persona title and then clicking on either the title or description to

type in the custom title or description. This allows the user the flexibility to guide the system to generate perspectives more aligned with their expectations - in line with Design Goal 5.

- **Initialization of debate** (Fig. 4.5, Point D): Once the user is satisfied with the created personas, they can choose to start the simulated debate between them by clicking on the "Start Debate" button.

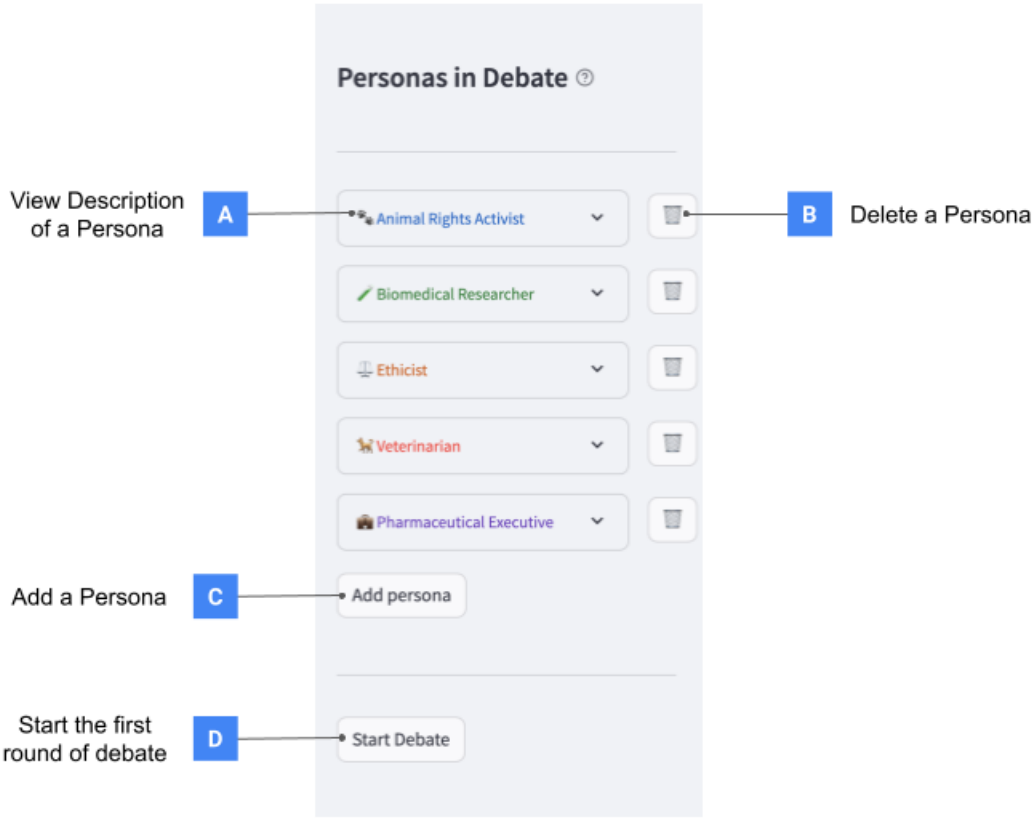
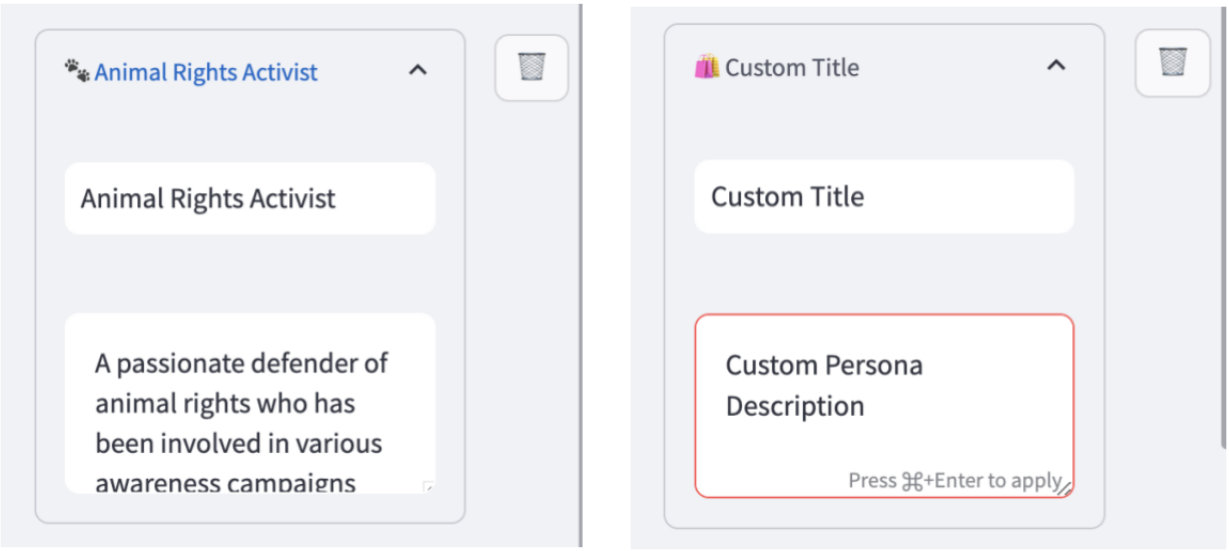


Figure 4.5: Persona Sidebar, with annotated interface features

4.1.3 Debate Window

Once the "Start Debate" button is clicked, the user's attention is diverted to the right-hand side column of the screen, where a loading icon along with a



(a) Description of a Persona (b) Feature to Customize a Persona

Figure 4.6: Viewing and Customization of a Persona's Title and Description

brief explanatory message is displayed to indicate that the results for the first round of the debate are being generated (Fig 4.7).

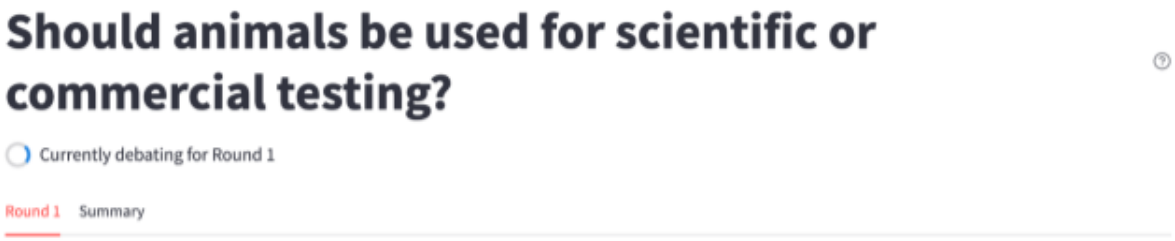


Figure 4.7: Debate Window View while the Results are being Generated

The debate results have the following characteristics:

(a) The results are displayed in a sequential manner, with each persona responding by addressing all the perspectives presented before.

(b) The order in which responses are given, or the order of the "debate" is

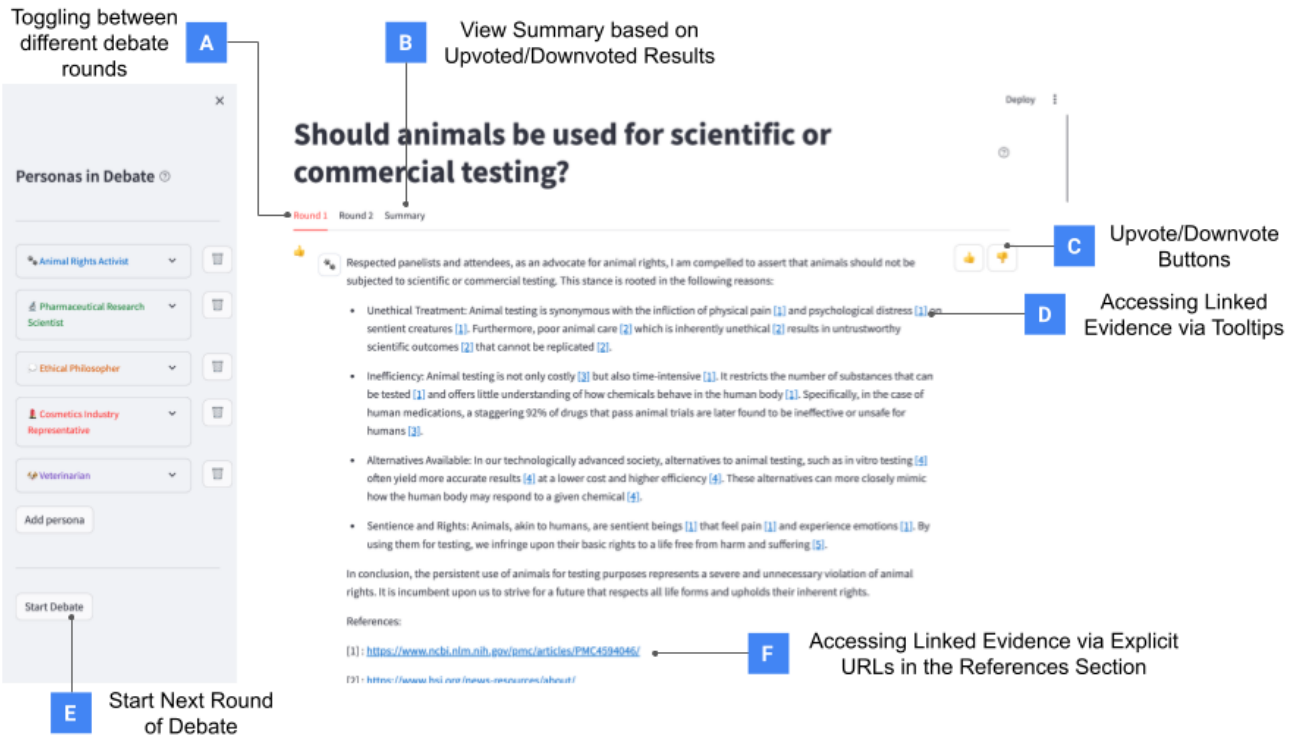


Figure 4.8: Debate Window View after results are generated, and annotated with the interface features

determined by the order in which the personas were aligned in the Persona Sidebar.

(c) For each persona in order, the result is displayed as and when it is generated at the back-end.

(d) The results are presented in a point-wise manner, to enhance readability and to aid easy consumption by the user.

(e) Each result contains an initial 2-3 sentence prologue that briefly addresses the persona's position on the argument, and in some cases also focuses on points made by other personas previously in the debate. This is intended to add a layer of continuity in the debate, and make it easy to follow for the user - in line with Design Goal 4.

(f) All perspectives presented by the personas are constrained to a fixed word limit, in order to avoid giving any implicit recommendation to the user (longer responses might get interpreted as being better or vice versa) - in line with Design Goal 3.

Once the results for all personas are generated, the debate window supports the following interactive features:

- **Accessing Linked Evidence** (Fig. 4.8, Point D): The supporting evidence for each result is added as in-line citations for compactness and readability. The user can hover over the tooltip and then navigate to the evidence source by clicking on the URL which opens in a new window. The "References" section at the end of the result (Fig. 4.8, Point F) can also be used to navigate to the evidence sources.
- **User Feedback via Upvote/Downvote Buttons** (Fig. 4.8, Point C): As a way to obtain user feedback for the generated results, the upvote/downvote buttons are intended to signify agreement/disagreement of the user with the generated perspective. Upon clicking either button, the argument also appears in the "Summary" section, which is discussed in further detail in the next subsection.
- **Start Next Round of Debate** (Fig. 4.8, Point E): The button that was originally used to start the first round of the debate, can be used repeatedly by the user to generate further rounds of debate. Although not a part of the debate window, the placement of this button is kept unchanged to minimize the complexity of the interface.
- **Toggle Between Multiple Debate Rounds** (Fig. 4.8, Point A): Once the user has generated more than one round of debate, they can view the results for the desired debate round by toggling the respective buttons for each debate round, near the top of the debate window.

4.1.4 Summary Section

The summary section is nested under the main debate window interface (Fig. 4.9), and is meant as an independent module to facilitate user feedback and review. This section can be accessed by clicking on the "Summary" toggle button (Fig. 4.8, Point B), which is located near the individual debate round toggles.

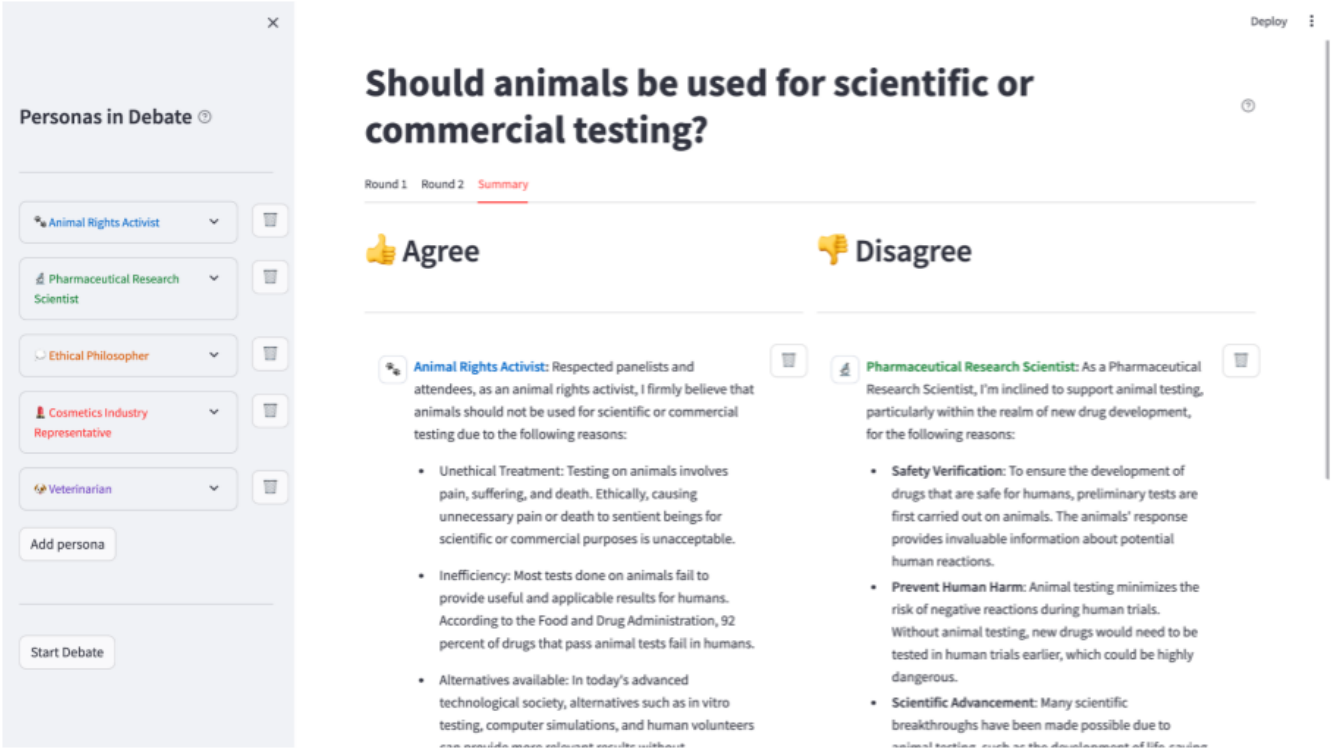


Figure 4.9: Summary Tab View

The section has a two-column view with the columns representing the user's upvoted and downvoted results, respectively. This module does not have any interactive features, and is intended to serve two main purposes:

(a) **Review of Results**: The section provides an easy way for the user to review the arguments they agreed/disagreed with and can be considered

akin to a "bookmark" feature. It is intended to enable the user to capture and summarize the important takeaways from the debate results.

(b) **User Engagement**: This module serves an important user engagement need as part of the system - the upvote/downvote buttons serve as a proxy for the user being engaged throughout the debate rounds. This can be helpful for the researchers in analyzing user behaviour at a very crude level.

4.2 System Architecture and Integration

An overview of the back-end configuration supporting the generation of results being displayed at the front-end can be seen in the architecture diagram (4.10).

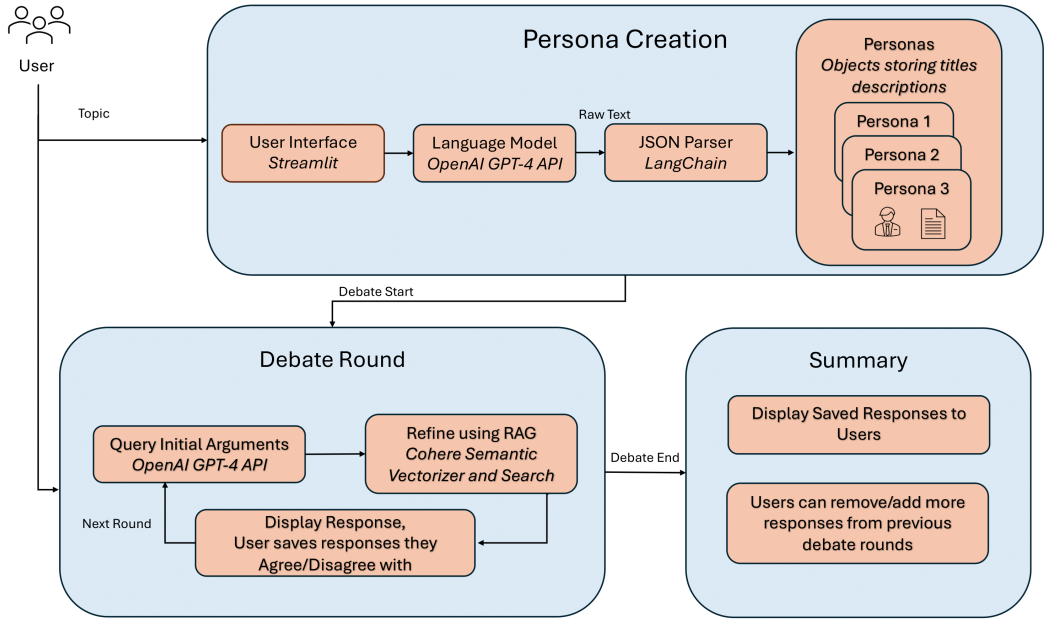


Figure 4.10: Architecture Diagram

The following subsections will cover the working of each module of the system

in detail.

4.2.1 Persona Identification and Generation

The first module in the architecture is responsible for identification of relevant personas given a topic of interest. The module generates the title, description and a representative emoji character for each persona by prompting a language model using the prompt shown below:

Prompt Used:

```
Given the topic [TOPIC], create a roundtable debate of different
personas to show key perspectives on the issue.  Output the
personas as a list of JSON objects.  Each JSON object should
have the following structure:
{ ‘‘title":  <title of the Persona>,
"description":  <Brief Description of the Persona, only describing
their background (rather than their stance)>,
"emoji":  <Single emoji representation of the perspective the
persona represents>}.
Ensure that the output is formatted as a valid JSON.
Please generate exactly [NUM_PERSONAS] personas.
```

This prompt is then used to obtain the list of personas containing their titles, descriptions and representative emojis in the form of a JSON dictionary, which is parsed using Langchain library's JSON parser in the subsequent modules. The prompt controls the number of personas that are to be generated at the start of the debate using the [NUM_PERSONAS] parameter. In case additional personas are to be generated based on the user's needs, the following prompt is used:

```
Given the topic ’[TOPIC]’, we are creating a roundtable debate
of different personas to show key perspectives on the issue.
Currently, we have personas as follows:
CURRENT PERSONAS
```

```
Please output exactly one additional persona.  Your output
should have the following JSON structure:
"title":  <title of the Persona>,
"description":  <Brief Description of the Persona, only describing
their background (rather than their stance)>,
"emoji":  <Single emoji representation of the perspective the
persona represents>.
Ensure that the output is formatted as a valid JSON.
```

4.2.2 Initial Argument Creation

Once the personas are identified and created, and the user clicks on the "Start Debate" button, the system moves onto the initial argument creation module. The topic of interest, the title and the description of the first persona in order are then passed as context in the following prompt:

```
You are in a roundtable debate on the topic [TOPIC]. You are
[NAME], who is [DESC]. Please start the debate by concisely
presenting your argument for your stance on the topic.  [LIMITER]
```

This generates the first argument of the debate, which can then be used as input in the subsequent debate module.

4.2.3 Debate Mechanism

Using the initial argument as context, follow-up arguments are generated using the following prompt:

```
You are in a roundtable debate on the topic [TOPIC]. You are
[NAME], who is [DESC]. Please start the debate by concisely
presenting your argument for your stance on the topic.  [LIMITER]
```

The history of the debate is provided as part of the [HISTORY] parameter, and this can either be the complete history of the debate so far or a summarized version generated using a BERT extractive summarizer (Miller (2019)), depending on the length of the debate history - this is done keeping context windows of LLMs in mind.

4.2.4 Evidence Retrieval using RAG

For each result in the debate, supporting evidence is generated based on a Retrieval Augmented Generation pipeline as shown in Fig. 4.11. The input to the pipeline is an argument generated either via the initial argument creation module, or the subsequent debate module. Since the task at hand is to preserve the perspective elicited in these arguments, and add a layer of linked evidence on top, this module tries to make as few as possible changes to the original input argument. The main purpose of using Large Language Model-based generation here is to ensure semantic alignment of the evidence being found with the perspective present in the argument.

The pipeline includes the following steps:

(a) **Search Query Extraction**: Using the original argument as input, an LLM is prompted to generate a search query using the following zero-shot prompt:

```
Given a certain argument, you are supposed to find evidence
online supporting that argument using a search engine.
Argument - [ARGUMENT]
What search query would you use for this task?  Provide
just the search query and nothing else.
```

The LLM used here is GPT-4 from the OpenAI's GPT series of models.

(b) **Retrieval of Sources**: The generated search query from the previous step is then used to retrieve sources that can possibly act as evidence by

performing a Google Search API call that returns the URLs of the search results.

(c) **Processing of Search Results**: For each search result URL, the webpage text is obtained using Langchain's Document Loader. This text is then converted into smaller chunks to make it simpler for the language model to ingest and process later, with each chunk having a maximum character limit of 500 words. This action is performed using the Unstructured library.

(d) **Vectorizer and Indexer**: The text present in each chunk needs to be encoded into a vector representation in order to assess its semantic similarity with the extracted search query. This action is performed using Cohere's Embed-English-v3.0 embedding model. Following the creation of vectors of all chunks, a vector index is created using the Hnwslib library. This index makes sure that the retrieval is done in a quick and efficient way.

(e) **Vectorization of the Search Query**: Similar to the chunks that were vectorized, the extracted search query is also encoded in a vector representation.

(f) **Retrieval of Top Search Results**: The retrieval step entails computing the semantic similarity between the vectorized search result chunks and the vectorized search query. The similarity is measured by calculating the cosine similarity between the vectors - the higher the cosine similarity, the higher is the semantic similarity as well. After similarity scores are calculated between each of the chunks and the extracted search query, the top ten search results with the highest scores are selected in the end.

(g) **Augmentation of context and Generation** - The retrieved search results are then added as context to a prompt along with the extracted search query. This prompt is then passed on to GPT-4, and the generated result is the rewritten argument with a list of references, and in-line citation markers that are linked with the corresponding reference URL while

rendering at the front-end.

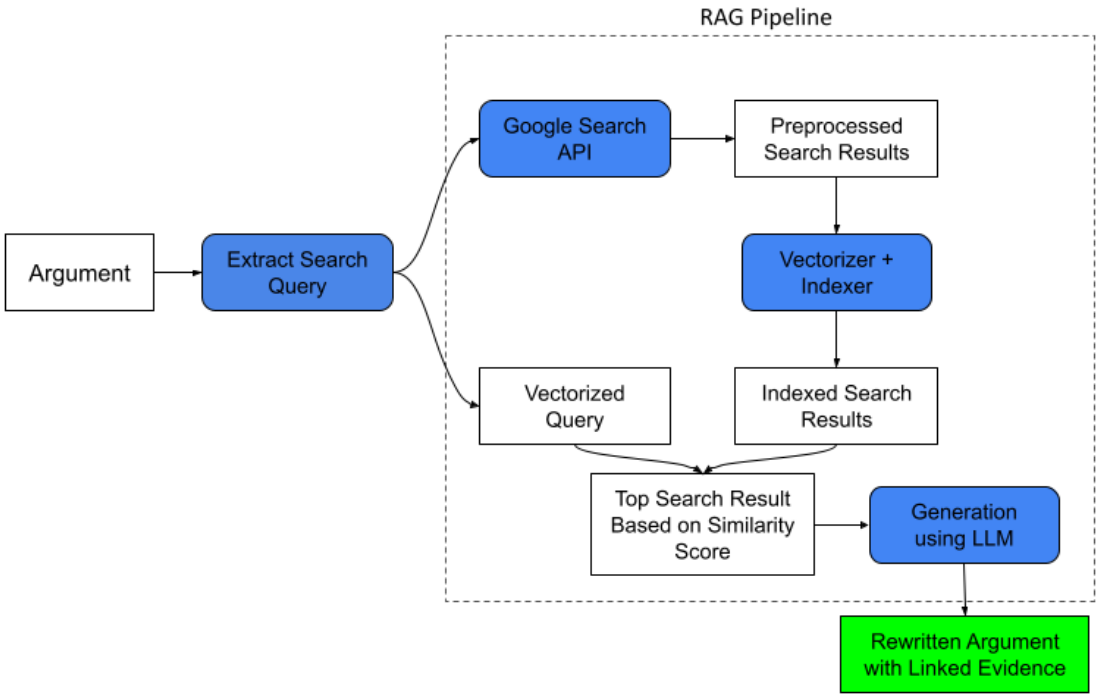


Figure 4.11: Evidence Retrieval Pipeline based on Retrieval Augmented Generation.

4.3 Implementation Challenges and Solutions

We encountered several challenges during the development process. The main issues and their respective solutions have been highlighted below.

(a) **Context Window Length of LLMs**: One of the drawbacks of working with Large Language Models stems from their limitation of being able to process a limited number of words at one time, and this limit is termed as the Context Window (Talamadupula (2023)). This issue arose when the context of the existing arguments presented in the debate was passed to an LLM agent to generate the subsequent arguments. This issue was

tackled by using a BERT Extractive Summarizer model (Miller (2019)) to summarize the context before passing it on to the LLM agent.

(b) **Stochasticity of outputs**: The other prominent issue with LLM powered applications is that they have a tendency to produce dissimilar outputs for the same prompt, owing to the stochastic nature of the models themselves. Inconsistency in the outputs can lead to challenges in evaluating the system, and more generally affects the stability of the system itself in terms of results. While there is no exact solution for this issue, we have tried to address it by controlling the temperature hyperparameter of the models used which controls the randomness and creativity of the generated text, as well as by making the prompts as instructionally exhaustive as possible to reduce possibilities of out-of-scope outputs.

(c) **Processing time for each argument**: The latency or processing time associated with generation of outputs was a key implementation hurdle for us. While employing LLM-based pipelines and working with an RAG framework with the native CPU processing of a local machine, the average processing time for a generated result in a debate was 63.4 seconds. In order to bring this time down, we chose to host our web-app via AWS Sagemaker which enabled us to utilize faster compute instances such as the AWS G4dn.xLarge, which is powered by a GPU unit with 16 GB RAM, enabling us to bring down the processing time to 42.1 seconds.

(d) **In-line citations as part of RAG**: One of the main features we wanted to implement as part of the RAG module was the in-line citations to show linked evidence supporting the perspective. This would mean that the sources used to generate the output would be cited in line within the output itself, along with the presentation of references at the end. This feature implementation proved to be a significant challenge since typical RAG pipelines only output the generated text along with the references

and no citations are provided as such. Cohere's RAG implementation was deemed to be a solution for this challenge, as its instruction model as part of the implementation has been trained to output citations along with the token windows in which the citation is present.

Chapter 5: Discussion

The development of the system outlined in this thesis represents a significant step towards integrating the interactive and contextually nuanced capabilities of Large Language Models (LLMs) into the realm of information retrieval and presentation, for multi-perspective search scenarios. The system has been developed as a front-end interface integrated with a back-end architecture based on prompting of Large Language Models.

The developed system aligns with the outlined design goals in 3.2, with a key focus on serving the multi-perspective search use case in a conversational style flow, without prescribing any particular perspective, and providing the user with an interactive interface with easy to understand features.

The back-end architecture is supported by a combination of prompt engineering of Large Language Models, Retrieval Augmented Generation facilitated by search results from the web as well as some specialized models such as the BERT Extractive Summarizer to summarize and shorten context wherever needed.

5.1 Potential Applications

The application scenario discussed so far for our tool has been that of a broad multi-perspective search engine that can enable the users to simulate diverse perspectives for a given topic of interest, and potentially provide a more holistic way of information retrieval. This tool can also have applications in some specific scenarios:

- **Personal Decision Making Tool**: When it comes to decision making regarding aspects of career or lifestyle choices, our tool can be very helpful to simulate diverse opinions backed by reliable evidence. For example,

consider the case of an undergraduate college student who wants to decide whether they should pursue a job after graduation or go on to do a PhD in their discipline. Our tool can help break this complex decision down into a debate between personas who can represent the two scenarios and the student can make an informed decision by studying the debate - a much more holistic solution than what a traditional search engine can provide.

- **Public Opinion Simulator**: A lot of industries are concerned about the effects of their actions on the public perception of a certain product or entity. Public Policy Departments and the fields of Marketing and Advertising are two such examples, who can benefit from our tool by simulating opinions of the important stakeholders for a particular topic of interest that concerns their future actions.
- **Opinion-checker for Social Media**: Social media platforms often struggle with the emergence of echo chambers - environments that amplify or reinforce users' preexisting beliefs by communication and repetition with disregard for beliefs or opinions different from their own. This leads to alleviated confirmation biases in people, and a heightened bias in their information consumption behavior. Our tool can potentially be used as an integration for social media platforms, providing users with different perspectives related to the topic they are currently viewing on the platform, and help them understand opinions different from their own and as a process help them in identifying their bias and gaining a more holistic view on the topic at hand.

5.2 Limitations

- **Verbosity of Results**: Our proposed system deals with the trade-off of providing enough context in order to be unbiased towards any one perspective versus the attention span and cognitive abilities of users to process

longform text. The response length for each result has been set to 500 words to maintain the quality of results, but this can potentially lead to 2000+ words per debate round given four personas. This can be a significant burden on the user, and the effectiveness of the tool can be limited because of this. Potential solutions can involve using a medium other than text to design the system, such as audio or video to convey the results.

- **Speed of Generation**: Our system takes an average of 42.1 seconds per generated result, which is significantly higher than the few miliseconds that traditional search engines take to generate search results. This makes for a poorer user experience and might lead to drop-off in user engagement as well. Future work can be focussed on improving the efficiency of the back-end architecture through parallelization process and other optimizations.
- **Lack of Contrarian Perspectives** For certain topics of interest, the personas that the system comes up with might all be on the same side of the argument - this might lead to a one-sided debate on the topic of interest and thus result in a biased presentation of perspectives on the topic. The prompt being used to generate the personas can be further engineered to avoid this scenario, and the stance of personas can be explicitly delineated in the persona sidebar interface to better inform the user.
- **Knowledge Cut-off Date of LLMs** Large Language Models such as ChatGPT and GPT-4 are trained on massive amounts of data borrowed from the internet, but once training is completed there is very little scope to update their fundamental knowledge base. This can leave them susceptible to knowledge gaps when considering world events or phenomena that occured after their training was completed. This issue can be mitigated partially by the use of Retrieval Augmented Generation - which our system employs - but since our pipeline is still solely dependent on LLMs for the original argument creation, we are never able to fully tackle this issue.

While there are no clear and obvious solutions to completely mitigate this issue, it is important to address it as part of the design process, as it is a caveat of dealing with LLMs in general.

5.3 Future Work

- **Robust Evaluation** The efficacy and accuracy of the current system needs to evaluated to determine how effective it is in multi-perspective search scenarios. Future work can focus on three dimensions of evaluation:
  - **Usability of the Tool**: The evaluation of the design elements of the front-end interface would require a user-study based evaluation to analyze the ease-of-use and the functionality of these features. In-depth user interviews can also be conducted to explore newer features that might improve the system and align it better with the core motivation of the study.
  - **Coverage of Different Perspectives**: The fundamental purpose of the tool is to provide a holistic result encompassing all the different perspectives on a topic-of-interest. Evaluating the coverage of perspectives presented thus becomes a key component of the overall evaluation. The PERSPECTRUM dataset by Chen et al. (2019) can prove to be a useful resource in this scenario.
  - **Efficacy of Evidence Retrieval**: RAG systems, due to their multi-component nature, can be very difficult to evaluate in terms of relevance and accuracy of generated results. However, context-agnostic techniques for evaluation of RAG systems such as RAGAS by Es et al. (2023) can be a helpful starting point here. These techniques help evaluate the different components (retrieval model, embedding model, generative model) of the RAG system independently, offering a deep insight into how each component is performing for the given task.

- **Recommendation Features as part of the System**: The proposed system had a design goal of staying netural in recommendation, and not prescribing a certain perspective as part of the result. A contrary solution that can be explored is a system which has features which points the user towards certain perspectives or results. The application of such a tool might involve nudging the user towards a desired outcome by presenting evidence-backed perspectives. One way this can potentially be done is by introducing an AI agent that summarizes the debate, and also points to the more apt perspectives presented in the debate by evaluating their arguments and linked evidence. This can be done either in the form of a quantitative rating based evaluation, or in the form of a qualitative, verbose solution.

Chapter 6: Conclusion

The work presented as part of this thesis has aimed to put forth a novel system of multi-perspective information retrieval by employing the natural language capabilities of Large Language Models. The system was developed in the form of a conversational interface, to enable users to search for, and obtain information similar to search engines, but with an added layer of conversational flow.

Future research should aim to address the identified limitations, exploring more sophisticated methods for persona development, debate generation, and factual verification to ensure the tool's reliability and applicability across various contexts. The potential integration of the system as a plugin or its application in diverse fields like social media, public policy, and marketing points to its broad utility and the significant impact it could have on decision-making processes and information consumption.

In conclusion, while the journey to refine and expand this system's capabilities is ongoing, the groundwork laid by this thesis contributes to the evolving landscape of AI-driven information tools, paving the way for more informed, balanced, and constructive engagements with complex topics in our increasingly information-driven world.

Appendix

Code and Documentation

The code and documentation for the back-end architecture and the front-end integration can be found here

Demonstration Video

A short video introducing the tool, its various features and the interface flow can be found here

Works Cited

AWS G4dn.xLarge. Aws instance types. URL https://aws.amazon.com/ec2/instance-types/.

BARD. An important next step on our AI journey. URL https://blog.google/technology/ai/bard-google-ai-search-updates/.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? . In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623, Virtual Event Canada, March 2021. ACM. ISBN 978-1-4503-8309-7. doi: 10.1145/3442188.3445922. URL https://dl.acm.org/doi/10.1145/3442188.3445922.

Bing Chat. Bing Chat | Microsoft Edge. URL https://www.microsoft.com/en-us/edge/features/bing-chat.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. Language (Technology) is Power: A Critical Survey of "Bias" in NLP. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.485. URL https://aclanthology.org/2020.acl-main.485.

Claire Cardie, Janyce Wiebe, Theresa Wilson, and Diane J. Litman. Combining low-level and summary representations of opinions for multi-perspective question answering. In *New Directions in Question Answering*, 2003. URL https://api.semanticscholar.org/CorpusID:1317793.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. ChatEval: Towards Better LLM-based Evaluators through Multi-Agent Debate, August 2023. URL http://arxiv.org/abs/2308.07201. arXiv:2308.07201 [cs].

Tyler A. Chang, Katrin Tomanek, Jessica Hoffmann, Nithum Thain, Erin van Liemt, Kathleen Meier-Hellstern, and Lucas Dixon. Detecting Hallucination and Coverage Errors in Retrieval Augmented Generation for Controversial Topics, March 2024. URL http://arxiv.org/abs/2403.08904. arXiv:2403.08904 [cs].

ChatGPT. Introducing ChatGPT. URL https://openai.com/blog/chatgpt.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. Benchmarking Large Language Models in Retrieval-Augmented Generation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17754–17762, March 2024. ISSN 2374-3468. doi: 10.1609/aaai.v38i16.29728. URL https://ojs.aaai.org/index.php/AAAI/article/view/29728. Number: 16.

Sihao Chen, Daniel Khashabi, Wenpeng Yin, Chris Callison-Burch, and Dan Roth. Seeing Things from a Different Angle:Discovering Diverse Perspectives about Claims. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 542–557, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1053. URL https://aclanthology.org/N19-1053.

Sihao Chen, Siyi Liu, Xander Uyttendaele, Yi Zhang, William Bruno, and Dan Roth. Design Challenges for a Multi-Perspective Search Engine. In Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz, editors, *Findings of the Association for Computational Linguistics: NAACL 2022*,

pages 293–303, Seattle, United States, July 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.22. URL https://aclanthology.org/2022.findings-naacl.22.

Claude. Claude. URL https://www.anthropic.com/claude.

Cohere's RAG. Retrieval augmented generation. URL https://docs.cohere.com/docs/retrieval-augmented-generation-rag.

Copilot. Introducing Microsoft 365 Copilot – your copilot for work - The Official Microsoft Blog. URL https://blogs.microsoft.com/blog/2023/03/16/introducing-microsoft-365-copilot-your-copilot-for-work/.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. Improving Factuality and Reasoning in Language Models through Multiagent Debate, May 2023. URL http://arxiv.org/abs/2305.14325. arXiv:2305.14325 [cs].

Embed-English-v3.0. Introducing embed v3. URL https://cohere.com/blog/introducing-embed-v3.

Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. Ragas: Automated evaluation of retrieval augmented generation, 2023.

Google Search API. Custom search json api. URL https://developers.google.com/custom-search/v1/overview.

GPT Series. Models overview. URL https://platform.openai.com/docs/models/overview.

Hnwslib. Hnswlib - fast approximate nearest neighbor search. URL https://github.com/nmslib/hnswlib.

Langchain's Document Loader. Document loaders. URL https://python.langchain.com/docs/modules/data_connection/document_loaders/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate, May 2023. URL http://arxiv.org/abs/2305.19118. arXiv:2305.19118 [cs].

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. Generated Knowledge Prompting for Commonsense Reasoning. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.225. URL https://aclanthology.org/2022.acl-long.225.

LLaMa2. Llama 2. URL https://llama.meta.com/llama2/.

Robert Logan IV, Ivana Balazevic, Eric Wallace, Fabio Petroni, Sameer Singh, and Sebastian Riedel. Cutting Down on Prompts and Parameters: Simple Few-Shot Learning with Language Models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2824–2835, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.222. URL https://aclanthology.org/2022.findings-acl.222.

Donald Metzler, Yi Tay, Dara Bahri, and Marc Najork. Rethinking Search: Making Domain Experts out of Dilettantes. *ACM SIGIR Forum*, 55(1):1–27, June 2021. ISSN 0163-5840. doi: 10.1145/3476415.3476428. URL http://arxiv.org/abs/2105.02274. arXiv:2105.02274 [cs].

Derek Miller. Leveraging BERT for extractive text summarization on lectures. *CoRR*, abs/1906.04165, 2019. URL http://arxiv.org/abs/1906.04165.

Perplexity. Perplexity Blog. URL https://www.perplexity.ai/hub.

Sumedh Rasal. LLM Harmony: Multi-Agent Communication for Problem Solving, January 2024. URL http://arxiv.org/abs/2401.01312. arXiv:2401.01312 [cs].

Murray Shanahan, Kyle McDonell, and Laria Reynolds. Role-Play with Large Language Models, May 2023. URL http://arxiv.org/abs/2305.16367. arXiv:2305.16367 [cs].

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval Augmentation Reduces Hallucination in Conversation, April 2021. URL http://arxiv.org/abs/2104.07567. arXiv:2104.07567 [cs].

Karthik Sreedhar and Lydia Chilton. Simulating Human Strategic Behavior: Comparing Single and Multi-agent LLMs, February 2024. URL http://arxiv.org/abs/2402.08189. arXiv:2402.08189 [cs].

Kartik Talamadupula. Guide to Context in LLMs, December 2023. URL https://symbl.ai/developers/blog/guide-to-context-in-llms/.

Unstructured. Unstructured-io. URL https://github.com/Unstructured-IO.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You

Need, August 2023. URL http://arxiv.org/abs/1706.03762. arXiv:1706.03762 [cs].

Zhenhailong Wang, Shaoguang Mao, Wenshan Wu, Tao Ge, Furu Wei, and Heng Ji. Unleashing the Emergent Cognitive Synergy in Large Language Models: A Task-Solving Agent through Multi-Persona Self-Collaboration, January 2024. URL http://arxiv.org/abs/2307.05300. arXiv:2307.05300 [cs].

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, January 2023. URL http://arxiv.org/abs/2201.11903. arXiv:2201.11903 [cs].

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. Tree of Thoughts: Deliberate Problem Solving with Large Language Models, December 2023. URL http://arxiv.org/abs/2305.10601. arXiv:2305.10601 [cs].

Vita

Utkarsh Mujumdar is going to graduate soon!

Address: utkarsh.mujumdar@utexas.edu

This thesis was typeset with LaTeX[†] by the author.

[†]LaTeX is a document preparation system developed by Leslie Lamport as a special version of Donald Knuth's TeX Program.