Online Shopping Assistant
Team 2

Amruth Amruth[2043415], Jahanvi Panchal[1939439], Marmee Pandya[1963521], Manasi Patil[2034414], and Rutuja Gaikwad[1960118]

University of Mannheim, Mannheim, Germany

1 What is the problem you are solving?

Online shopping can be overwhelming and time-consuming, often requiring users to go through numerous websites, compare multiple products, and navigate technical product descriptions. The challenge becomes even more daunting when users need to match their specific requirements with relevant product information, including reviews and specifications from various sources.

To address these challenges, we propose developing a multi-agent shopping assistant powered by a Large Language Model (LLM). This assistant will automate browsing, intelligently search for products matching user-defined criteria, analyze product information, and compare specifications and reviews across multiple platforms. By delivering concise, relevant insights, our solution will make online shopping more intuitive, efficient, and user-centered, helping users make confident and informed purchase decisions with minimal effort.

2 What data will you use?

We will use product data that includes details like product specifications, prices, features, and reviews. This data will help us provide the most relevant and up-to-date information for the user's shopping needs.

2.1 Where will you get it?

We will gather product data primarily from the 'Shop' section of Google. This section provides a wide range of product listings along with detailed specifications and purchasing options.

2.2 How will you gather it?

To gather the data, we will use SERPapi and Tavily, which will help us automate the process of retrieving product information from the web. These tools will allow us to search for products based on user requirements, extract relevant details, and present them in a clear and organized way. We will also explore other tools and methods that might help enhance our data collection and improve the quality of results.

3 How will you solve the problem?

To create an efficient online shopping assistant, we propose a multi-agent framework using LangGraph for workflow management and Streamlit for a user-friendly interface. We will utilize large language models to handle tasks like retrieving, validating, ranking, and summarizing product listings, streamlining the shopping experience.

The Search Agent will process user queries and retrieve relevant products, while the Validator Agent ensures quality by filtering irrelevant results. The Ranker Agent will prioritize products based on user preferences, and the Summarizer Agent will generate concise summaries to help users make informed decisions.

3.1 What LLMs do you plan to use?

We plan to use LLaMA or DeepSeek v3 models for different agents in our system:

- **Search Agent:** LLaMA will refine user queries and enhance search retrieval by structuring queries for better web search results.
- **Summarizer Agent:** LLaMA will generate concise, structured summaries, highlighting key product details and ranking explanations.
- **Validator Agent:** DeepSeek v3 will validate and clean product listings by filtering out irrelevant, duplicate, or incomplete results.
- **Ranker Agent:** DeepSeek v3 will assist in ranking products based on multi-criteria decision-making (MCDM), ensuring rankings are data-driven rather than solely relying on LLM reasoning.

If the performance of LLaMA or DeepSeek v3 does not meet expectations, we will explore integrating OpenAI models for specific tasks where enhanced reasoning or better accuracy is required. This flexible approach ensures that the system operates optimally while balancing efficiency and robustness.

3.2 Which methods do you plan to apply?

Each agent employs specific methods to optimize system accuracy and relevance.

Product Searching Methodology (Search Agent)

- **Structured Query Processing:** Constructs search queries dynamically using extracted attributes (category, brand, features, price).
- **Feature-Based Filtering:** Ensures only relevant products are retrieved (e.g., “gaming laptop with RTX 4070”).
- **Multi-Source Aggregation:** Uses Tavily and SerpAPI to enhance result diversity and accuracy.
- **Optimized Product Retrieval:** Filters results based on user-defined constraints (e.g., price range, must-have features).

Query Validation and Filtering Methodology (Validator Agent)

- **Keyword and Attribute Matching:** Identifies key product features (brand, specifications, price) to ensure accurate filtering.
- **Query Refinement:** Cleanses and restructures user queries for better alignment with search APIs.

Product Ranking Methodology (Ranker Agent)

- **Advanced Product Ranking:** Uses multi-criteria decision-making (MCDM) to rank products based on:
  - User intent (budget vs. premium)
  - Price-to-performance ratio
  - Reviews, ratings, and brand trustworthiness
- **NDCG-Based Evaluation:** Ensures ranking relevance by comparing ranked results to ideal rankings.

Summary Generation Methodology (Summarizer Agent)

- **Key Feature Extraction:** Extracts essential product details using LangGraph for concise summaries.
- **Best Pick and Budget Pick Recommendations:** Highlights the top product and best budget-friendly option.
- **Structured Summary Format:** Presents product insights in a table format, making decision-making easier.

3.3 What is your idea for a multi-agent workflow for your task?

User Query Processing Agent: Refines complex queries by extracting key elements such as product category, brand, features, price range, and purchase intent. Uses an LLM-based approach to restructure ambiguous queries into clear, structured components for accurate processing.

Search Agent: Retrieves product listings from multiple e-commerce platforms. Leverages web search APIs (SERPApi, Tavily) to fetch diverse, relevant results.

Validator Agent: Ensures search results align with user-specified criteria. The system applies keyword and attribute matching to extract essential product attributes such as brand, price, and features from the user's input. It also uses feature-based filtering to eliminate irrelevant or incomplete listings, retaining only the most relevant products.

Ranker Agent: Ranks retrieved products based on multiple weighted criteria. The system employs a multi-criteria decision-making (MCDM) approach, considering factors such as user intent (budget vs. premium preference), price-to-performance ratio, reviews, ratings, and brand trustworthiness. Additionally, an NDCG-based evaluation ensures the relevance of the product rankings.

Summarizer Agent: Generates concise, structured summaries to assist in decision-making. Using LangGraph, the system extracts essential product details and recommends the top-ranked products based on objective criteria. The summaries are presented in a clear table format, simplifying product comparison for the user.

User Interface (Streamlit): Displays results in an interactive UI, allowing users to refine searches, compare products, and make informed decisions.

4 How will you measure success?

The evaluation will focus on assessing the system's ability to retrieve and rank products that align with the user's specified preferences. It will measure how accurately the system delivers products based on key attributes such as price, performance, user intent (budget vs. premium), brand trustworthiness, and other requested features.

Evaluation Metrics:

The evaluation will assess how well the system retrieves and ranks products based on user preferences, considering attributes such as price, performance, user intent, and brand trustworthiness.

Metrics

- **Precision:** Measures the ratio of relevant products retrieved to the total returned, assessing how well the system meets the user's preferences.
- **NDCG (Normalized Discounted Cumulative Gain):** Evaluates product rankings by considering both relevance and position, ensuring top-ranked products are the most relevant.
- **Coverage:** Measures how well the system incorporates requested features in the retrieved products.

Evaluation Methodology

- **Test Set Creation:** A benchmark set of predefined queries with known correct matches.
- **System Query Execution:** The system processes queries, retrieving products, which will be evaluated based on precision and NDCG.
- **Comparison with Ground Truth:** Analyzes the retrieved products against expected results.
- **Human Evaluation:** Human reviewers ensure the system meets user expectations.