Bauhaus-Universität Weimar
Faculty of Media
Degree Programme Digital Engineering

Chart Retrieval for Arguments

Master's Thesis

Shashi Sharma

1. Referee: Prof. Dr. Benno Stein
2. Referee: Prof. Dr. Volker Rodehorst

Submission date: October 2, 2024

Declaration

Unless otherwise indicated in the text or references, this thesis is entirely the product of my own scholarly work.

Weimar, October 2, 2024

Shashi Sharma

. . . .
Shashi Sharma

Abstract

This thesis introduces the task of chart retrieval for arguments and a first approach to it based on Retrieval-Augmented Generation (RAG). Given a user query, the task is to retrieve relevant charts and generate supporting arguments to address it. This thesis analyzes the effectiveness of the RAG system in providing users with relevant charts and generated answers to their queries. The presented system retrieves relevant charts and their descriptions related to a given query using different retrieval approaches, both sparse and dense. These retrieval approaches are manually evaluated using the normalized discounted cumulative gain (NDCG) metric to determine their effectiveness. The result of the approach yielding the highest NDCG score is then inserted into a prompt template and submitted to the Large Language Model (LLM), precisely tailoring the information to answer the given query.

Contents

1 Introduction **1**

2 Background **3**
2.1 Retrieval-Augmented Generation 3
2.1.1 Conceptual Overview 3
2.1.2 Evolution of RAG 5
2.1.3 Evaluation of RAG 7
2.2 Multi-modal Data Retrieval 9
2.3 Chart Data Retrieval 12
2.4 Evidence-Based Question-Answering 15
2.4.1 Similarities and Differences with RAG 16

3 Methodology **19**
3.1 Chart Retrieval Framework 19
3.1.1 Overview 20
3.1.2 Application Back-end 21
3.1.3 Front-end Interface 21
3.2 Large Language Models 22
3.2.1 E5-Mistral-7B-Instruct 24
3.2.2 gte-Qwen2-7B-instruct 25
3.2.3 Contrastive Language-Image Pre-training 26
3.2.4 GPT-4o 26
3.2.5 Meta-Llama-3.1-70B-Instruct 27
3.2.6 Quantization 28
3.3 Indexing process 28
3.3.1 ElasticSearch 29
3.3.2 Handling of Data 29
3.3.3 Indexing 31
3.4 Retrieval Process 34
3.4.1 Sparse Retrieval 34
3.4.2 Dense Retrieval 35

3.5 Generation Process 36
3.5.1 Llama 3.1-Based Generation 37
3.5.2 GPT-4o-Based Generation 38
3.6 Dataset 41
3.7 Query Selection and Pre-processing 44
3.7.1 Touché Queries 44
3.7.2 Manual Queries 45
3.7.3 Random Queries 45

4 Evaluation **51**
4.1 Evaluation of Retrieval Process 51
4.1.1 Human Evaluation of Retrieval Results 51
4.1.2 NDCG Metric for Measuring Effectiveness 53
4.1.3 Results and Analysis 54
4.2 Evaluation of Generation Process 57
4.2.1 Human Evaluation of Generated Response 57
4.2.2 Results and Analysis 59

5 Conclusion **67**

A Evaluation Interface **70**

Bibliography **73**

Acknowledgements

I would like to express my gratitude to the 13 annotators who played a pivotal role in evaluating the retrieval and generation processes. Their dedication and commitment were invaluable in shaping the results of this thesis. Special thanks to Dr. Johannes Kiesel for his countless hours of consultation, insightful guidance, and unwavering support throughout this thesis.

List of Figures

2.1 Overview of the Three Paradigms of RAG. https://github.com/Tongji-KGLLM/RAG-Survey/tree/main/images/Survey 5

3.1 The chart retrieval pipeline proposed in this thesis comprises three processes: indexing, retrieval, and generation. 20
3.2 Payload sample for Meta-Llama 3.1 at NDCG@1 37
3.3 Payload sample for Meta-Llama 3.1 at NDCG@2 38
3.4 Payload sample for GPT-4o at NDCG@1. 39
3.5 Payload sample for GPT-4o at NDCG@2. 40
3.6 Topic Distribution in the Pew Dataset: analyzing proportional representation across domains. 42
3.7 Topic Distribution in the Statista Dataset: analyzing proportional representation across domains. 43
3.8 Empirical Cumulative Distribution Function of Title+Caption length (in tokens) and LLaVA-generated caption length. 44
3.9 Prompt for generating Touché like queries. 46

4.1 Number of times the generated answers for each query received a faithfulness score of 3 across 60 queries, evaluated across NDCG levels for $k \in [1, 3]$. 60

A.1 An interface for displaying retrieval results and facilitating human evaluation. 71
A.2 An interface displaying top retrieved documents at NDCG@3 with LLM-generated answers, facilitating human evaluation. . . 72

List of Tables

3.1 Indexed Database Structure 33
3.2 Sample Representation of Titles, Captions, and Charts from the Statista and Pew Research Dataset. 47
3.3 Chart Type Distribution. 48
3.4 Token Statistics for the combined dataset (Pew and Statista). . 48
3.5 Touché Queries with their corresponding topics. 48
3.6 Manual Queries with their corresponding topics. 49
3.7 Ramdom Queries with their corresponding topics. 50

4.1 Number of times each retrieval method is the best according to NDCG@k for all 60 queries. In case of a tie, the count is split evenly so that each column sums up to a total of 60. 55
4.2 Mean NDCG@k of each retrieval method for k$\in [1, 3]$. 56
4.3 Human Evaluation of LLM-generated answers for the top document retrieved at NDCG@1 for the Touché query shown at the top. 62
4.4 Human Evaluation of LLM-generated answers for the top document retrieved at NDCG@1 for the manual query shown at the top. 63
4.5 Human Evaluation of LLM-generated answers for the top document retrieved at NDCG@1 for the random query shown at the top. 64
4.6 Example of an LLM-generated answer, factually correct but not directly inferred from the provided content, verified via external web searches. 65
4.7 Mean Relevance and Faithfulness of the responses generated by each model across NDCG levels for $k \in [1, 3]$. 66

Chapter 1

Introduction

Chart retrieval has gained importance as digital content has become more multi-modal. While the need for retrieving charts and images has existed for decades, only with the advancement of web-based data and information retrieval (IR) systems are we now able to tackle this problem effectively. For instance, in today's data-driven environments, industries such as finance, marketing, and operations frequently rely on charts to access critical statistical insights that support strategic decision-making. For example, a financial analyst might query, *Show me the trends in stock prices over the last five years*, or an e-commerce marketing manager might ask, *What were the top-selling products during Black Friday last year?*. Therefore, the process of chart retrieval is crucial in modern information retrieval systems.

Recent advancements in pre-trained language models have significantly improved the ability of IR systems by providing a better understanding of the semantics and context of natural language queries and documents.[Hambarde and Proença, 2023]. Additionally, research into multi-modal large language models (MM-LLMs) has opened new possibilities for integrating diverse types of data, including text, images, and audio, allowing for more comprehensive and context-aware retrieval systems [Zhang et al., 2024].

Despite significant advancements in natural language processing and information retrieval, existing systems still face challenges in effectively retrieving and interpreting charts, in response to user queries. Access to this data is often critical, as charts provide a visual representation of trends and relationships that might not be available in textual information alone. This gap highlights a need for improved systems that can effectively retrieve charts along with textual information.

In response to this, this thesis proposes a Retrieval-Augmented Generation (RAG)-based framework capable of retrieving and generating responses using the charts and their descriptions. The RAG approach, a specific method of generative retrieval, involves two key steps: retrieval of relevant documents and generation of tailored answers [Gienapp et al., 2024]. Different retrieval approaches, including both sparse and dense methods, were implemented and evaluated through human judgment. The top retrieved documents from the best retrieval method were then combined with a prompt template and sent to the large language model (LLM) to generate contextually tailored answers. To ensure the reliability of the generated responses, a comprehensive evaluation through human judgment was conducted again. This evaluation assessed the overall effectiveness and potential limitations, focusing on how well the generated answer met the user's information needs.

This thesis is guided by the following research questions:

1. How effective are the retrieval methods in addressing a given query?

2. How reliable are the generated answers for a given query?

The structure of this thesis is organized to build a coherent narrative around these objectives:

- **Chapter 2:** This chapter reviews the evolution and advancements of RAG and multi-modal data retrieval. It further explores specialized approaches such as Chart Data Retrieval and Evidence-Based Question Answering, emphasizing the critical role of external knowledge integration in enhancing natural language processing performance.

- **Chapter 3:** This chapter presents the design and implementation of the Chart Retrieval Framework, integrating Elasticsearch and various LLMs for the effective retrieval of multi-modal content and the generation of reliable answers to given queries. It details the indexing, retrieval, and generation processes, the dataset utilized, and the query selection and pre-processing methodologies.

- **Chapter 4:** This chapter evaluates the Chart Retrieval Framework through human assessments to measure the effectiveness of its retrieval and generation processes. It analyzes performance across different query sets, identifies superior retrieval methods, and assesses the reliability of answers from LLMs, thereby validating the system's ability to generate relevant and faithful answers.

- **Chapter 5:** This chapter synthesizes the thesis findings, acknowledges its limitations, and outlines future research directions to enhance multi-modal chart retrieval and answer generation capabilities.

Chapter 2

Background

This chapter provides an overview of key advancements in Retrieval-Augmented Generation (RAG) and multi-modal data retrieval, emphasizing their evolution, methodologies, and applications. Section 2.1 introduces the conceptual framework of RAG, detailing the data indexing process and the two main phases of RAG—retrieval and generation. The section outlines the evolution of RAG, from Naive RAG to more advanced modular approaches, each addressing the limitations of earlier models. Additionally, it discusses the evaluation methods used to assess the effectiveness of RAG systems. The chapter in section 2.2 highlights recent innovations in multi-modal data retrieval, which integrate various data types (text, images, video) to improve accuracy and relevance in retrieval tasks. Section 2.3 further examines the progress in chart data retrieval, including techniques like ChartSense, Chart Decoder, and LineFormer, which utilize deep learning, computer vision, and hybrid models to extract data from various chart types (e.g., line, bar, and pie charts).

The section 2.4 concludes with a discussion on Evidence-Based Question-Answering, presenting it as a specific version of RAG that emphasizes source attribution and task specificity. Together, these topics demonstrate the growing importance of external knowledge integration in enhancing the performance of natural language processing systems across diverse applications.

2.1 Retrieval-Augmented Generation

2.1.1 Conceptual Overview

Retrieval-Augmented Generation (RAG) is an approach in natural language processing (NLP) that enhances the accuracy and contextual relevance of responses by integrating external information with the pre-existing knowledge of large language models (LLMs).

Prior to the effective operation of RAG, the system relies on a data indexing process to organize the external information that will be retrieved during inference.

Data Indexing Process: The data indexing process is essential for organizing large volumes of information in a manner that can be efficiently searched and retrieved. This preparation allows the system to respond to queries with precision and speed. The key steps include:

1. **Data Loading:** All documents or information intended for use are collected, which may include various formats such as PDFs, HTML, Word documents, etc. This step also involves data cleaning to ensure consistency.

2. **Data Chunking:** To improve retrieval accuracy and prevent information overload, large documents are divided into smaller, digestible chunks.

3. **Data Embedding:** The chunked data is then transformed into vector representations using embedding models. These vector embeddings capture the semantic meaning of the text, and the choice of embedding model depends on the specific use case, as different models excel in different scenarios. This step is optional and is not typically performed for sparse retrieval methods.

4. **Data Storing:** The vector embeddings are stored in a vector database, which is optimized for fast, accurate retrieval. This enables the system to quickly identify relevant pieces of information when answering a user's query [Gao et al., 2024].

With the data indexed, the RAG system is ready to operate through its two main phases: retrieval and generation [Gienapp et al., 2024].

1. Retrieval Phase: When a user submits a query, the system initiates the retrieval phase. It converts the user's input into a vector (query vector) and searches the vector database to identify the most relevant data chunks. The retrieval methods utilize advanced neural retrieval techniques like Dense Retrieval, which enables a deeper understanding of the query and the documents by encoding them as vectors.

2. Generation Phase: After retrieving the relevant information, the system moves into the generation phase. Various generative models can be employed to process the user's query along with the retrieved data, generating a coherent, factually grounded answer. This approach allows RAG to produce more accurate and contextually aware answers compared to models that rely solely on pre-trained knowledge.

The combination of a well-structured data indexing process and an effective retrieval approach enables RAG to overcome traditional limitations of large language models, such as hallucinations, outdated knowledge, and non-transparent or untraceable reasoning processes, in generating accurate and contextually relevant answers. By integrating external knowledge from dynamic, up-to-date sources, RAG becomes a powerful tool for addressing complex queries that require credible answers [Gao et al., 2024].

2.1.2 Evolution of RAG

The evolution of RAG has been a significant advancement in the field of NLP, addressing key limitations of LLMs, such as hallucinations and generating outdated information in classical NLP tasks like question answering and summarization [Béchard and Ayala, 2024]. RAG's development evolved through several distinct phases, each stage refining the methodology and addressing earlier limitations:

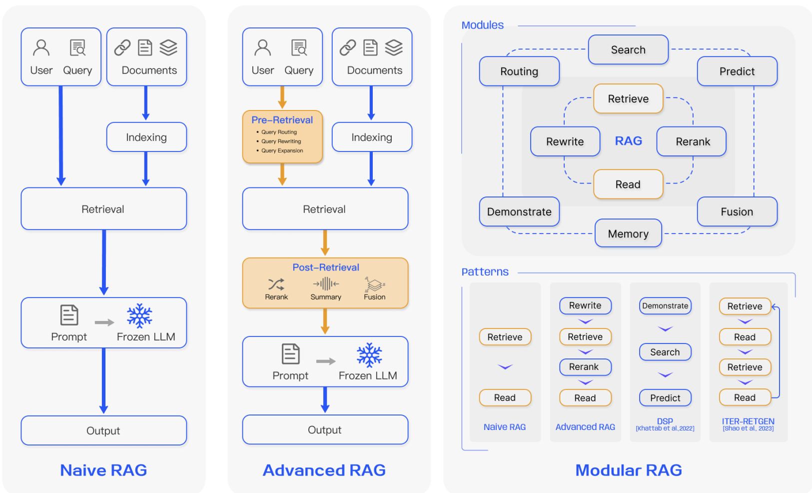


Figure 2.1: Overview of the Three Paradigms of RAG. `https://github.com/Tongji-KGLLM/RAG-Survey/tree/main/images/Survey`

- **Naive RAG** represents the earliest phase of the RAG methodology, following a simple Retrieve-Read framework. It involves three key steps: indexing, retrieval, and generation. During indexing, data from various formats like PDF and HTML is cleaned, chunked, and converted into vector representations stored in a vector database. When a user query is received, the system retrieves the most similar chunks based on vector similarity. The selected chunks are combined with the query in a prompt and provided to the LLM to generate an answer.

However, Naive RAG faces several limitations. The retrieval phase often struggles with selecting relevant chunks, leading to incomplete or irrelevant information being retrieved. The generation phase can suffer from hallucinations, producing unsupported or biased content. Additionally, integrating retrieved data can be challenging, resulting in redundancy or incoherent outputs. Naive RAG's simplicity limits its effectiveness, especially for complex queries that require deeper context and nuanced responses [Gao et al., 2024].

- **Advanced RAG** introduces specific enhancements to address the limitations of Naive RAG, focusing on improving both retrieval and generation quality. It incorporates pre-retrieval and post-retrieval strategies to refine the indexing process and optimize queries. Key improvements include the use of sliding window techniques (breaking large texts into overlapping segments), fine-grained segmentation (dividing the text into very precise units), and metadata integration to enhance data granularity and indexing precision.

  In the pre-retrieval phase, query optimization methods such as query rewriting and expansion are used to make user queries clearer and more effective for retrieval tasks. In the post-retrieval phase, retrieved information is re-ranked to prioritize the most relevant content, while context compression techniques are applied to streamline the data provided to the model, reducing information overload and improving response coherence. These refinements make Advanced RAG more effective in delivering accurate and contextually relevant outputs [Gao et al., 2024].

- **Modular RAG** builds on the foundations of Naive and Advanced RAG, offering enhanced adaptability and versatility through specialized modules that refine retrieval and processing. For example, the Search module allows direct searches across different data sources, while the Memory module creates a memory system that improves with use. RAG-Fusion helps expand user queries into multiple perspectives to find more relevant information, and the Predict module helps eliminate unnecessary or repetitive information to ensure the response stays accurate.

  What sets Modular RAG apart is its flexibility. It allows for swapping or reconfiguring modules based on the specific needs of the task. This flexibility goes beyond the fixed process of earlier RAG versions. New strategies, such as the Rewrite-Retrieve-Read model, refine retrieval by improving the initial query using feedback from the LLM. Hybrid approaches that combine keyword, semantic, and vector searches help improve the relevance of the retrieved information. The dynamic setup of modules in frameworks such as the Demonstrate-Search-Predict (DSP) and the iterative Retrieve-Read-Retrieve-Read flow (ITER-RETGEN) allows for more adaptive and responsive retrieval based on the situation. Additionally, Modular RAG works well with other technologies like fine-tuning and reinforcement learning, improving both the retrieval and generation processes for a wide range of tasks [Gao et al., 2024].

As RAG evolves, its impact on NLP becomes increasingly profound, bridging the gap between static pre-trained models and the need for real-time, accurate data retrieval. By continuously refining its methodologies and introducing new modular strategies, RAG not only enhances the performance of LLMs in knowledge-intensive tasks but also paves the way for more adaptive and intelligent AI systems across diverse domains, and multi-modal tasks that integrate images, videos, and code.

2.1.3 Evaluation of RAG

Evaluating Retrieval-Augmented Generation (RAG) systems can be challenging due to the integration of both retrieval and generation processes. Traditional metrics like BLEU and ROUGE do a decent job at measuring text generation quality but often miss the mark when it comes to capturing the nuances of the retrieval component.

1. **BLEU Score:** The Bilingual Evaluation Understudy, measures the quality of machine-generated text by comparing n-gram overlaps with reference texts. Scores range from 0 to 1, with higher values indicating better alignment.[1]

2. **ROUGE Score:** The Recall-Oriented Understudy for Gisting Evaluation, evaluates machine-generated summaries by comparing them to reference summaries, focusing on recall to determine how much reference content is captured. It's widely used for summarization tasks.[1]

Recognizing this gap, Yu et al. [2024] introduced a comprehensive framework called Auepora (A Unified Evaluation Process of RAG). Auepora breaks down the evaluation into three main modules: Target, Dataset, and Metric, focusing on three key questions: What to Evaluate? How to Evaluate? How to Measure?

[1] `https://www.elastic.co/search-labs/blog/evaluating-rag-metrics`

The Target module identifies key evaluation objectives for both the retrieval and generation aspects of RAG systems. It assesses the relevance and accuracy of retrieved information, as well as the system's ability to handle noisy or misleading data. Additionally, it evaluates whether the generated content faithfully reflects the retrieved data and aligns with the user's query. The Dataset component focuses on selecting datasets that represent various real-world scenarios and information sources, enabling a comprehensive evaluation of RAG systems across diverse contexts. Finally, the Metric component introduces specific measures tailored to the defined targets and datasets. These metrics include precision, recall, and relevance for retrieval, as well as faithfulness, coherence, and response relevance for generation. Additional metrics, such as latency, response diversity, and robustness, ensure that the evaluation covers not just accuracy but also the efficiency and reliability of RAG systems. Overall, Auepora offers a structured and adaptable methodology for thoroughly evaluating both retrieval and generation performance in RAG systems across real-world applications.

The advancements in RAG evaluation don't end with Auepora. Other frameworks like RAGAs (Retrieval Augmented Generation Assessment) and ARES (Automated RAG Evaluation System) have also made significant contributions. RAGAs offer a reference-free evaluation approach, examining dimensions such as passage relevance and content faithfulness, enabling faster iteration cycles crucial for LLM adoption [Es et al., 2023]. ARES, on the other hand, employs synthetic training data and human annotations to judge context relevance, answer faithfulness, and relevance across various domains [Saad-Falcon et al., 2023]. This is particularly important because, as studies have shown, even minor issues like typos can disrupt RAG systems, making them less reliable in practical settings [Cho et al., 2024].

One of the more challenging aspects of RAG evaluation is dealing with multiple languages. The NoMIRACL (a human-annotated dataset for evaluating LLM robustness) highlights this by testing RAG systems across 18 different languages, revealing that most models struggle to balance between minimizing hallucinations and maintaining low error rates. For example, models like LLAMA-2 and FLAN-T5 had high hallucination rates, while Mistral managed fewer hallucinations but at the cost of higher error rates [Thakur et al., 2023]. This underscores the ongoing challenge of creating RAG systems that perform well across different languages and contexts.

Language-specific evaluations provide additional insights. For instance, research on Arabic Retrieval-Augmented Generation (ARAG) has shown that certain models, like Microsoft's E5 sentence embeddings, are particularly effective for Arabic, achieving high recall rates on datasets like ARCD(Arabic reading comprehension dataset) [Abdelazim et al., 2023]. Similarly, the Retrieval-

Augmented Generation Benchmark (RGB), a new corpus for RAG evaluation in both English and Chinese, looks at key aspects like noise robustness and how well the system rejects irrelevant information [Chen et al., 2023]. These studies emphasize the need for language-specific optimizations to ensure RAG systems are effective in different linguistic contexts.

Ongoing research in RAG evaluation points towards several key areas for improvement. Enhancing multilingual robustness remains a priority, as current models struggle to maintain consistent performance across diverse languages [Thakur et al., 2023].

Future work should focus on developing more sophisticated evaluation metrics that can accurately assess RAG systems' ability to handle nuanced linguistic differences. There's also a need to address the vulnerabilities exposed by minor errors like typos, which can significantly impact the system's performance [Cho et al., 2024]. Researchers are also exploring ways to optimize the balance between hallucination prevention and error reduction, as exemplified by GPT-4's promising performance in this regard [Thakur et al., 2023]. These advancements will be instrumental in creating more reliable and versatile RAG systems for a wide range of applications.

This thesis contributes to these efforts by streamlining the data indexing process, embedding documents from structured datasets without relying on chunking, while preserving their original structure. Furthermore, an evaluation framework has been designed, combining quantitative metrics with comprehensive human assessments, to ensure a more robust and nuanced evaluation of the RAG system compared to existing methodologies.

2.2 Multi-modal Data Retrieval

Multi-modal data retrieval refers to the process of retrieving and integrating information from multiple data modalities (such as text, images, video, and audio) into a unified feature space to respond to user queries that might also involve multiple modalities [Rafailidis et al., 2013]. This approach enables a more comprehensive and accurate retrieval of information, but it also introduces challenges due to the complexity and heterogeneity of the data involved.

Significant progress has been made in addressing the challenges of multimodal data retrieval. Zhu et al. [2023] highlights the limitations of traditional retrieval methods when dealing with heterogeneous data types like text, images, and videos. The development of cross-modal retrieval techniques marks a pivotal advancement, enhancing user interaction and retrieval performance across different modalities. The study provides a comprehensive review of these methods, categorizing them into five main types: unsupervised real-value retrieval, supervised real-value retrieval, unsupervised hashing retrieval[2], supervised hashing retrieval, and cross-modal retrieval in special scenarios. It also highlights the evolution of these techniques from shallow statistical methods to advanced deep learning models, particularly through shared representation learning, has significantly bridged the gap between different data types, underscoring the importance of these techniques in modern retrieval systems.

The study *Deep Multi-modal Learning for Information Retrieval* by Ji et al. [2023] further explores the integration of multiple data modalities, emphasizing the impact of deep multi-modal learning on information retrieval systems. By leveraging advanced models like CLIP and other large language models, the study demonstrates how deep learning can significantly enhance the ability of retrieval systems to process and understand diverse data types. The focus on real-world applications, such as image-text and video-text retrieval, highlights the practical implications of this research, particularly in improving the accuracy and scope of retrieval tasks in complex environments.

To address specific limitations in traditional methods, the study by Zhu et al. [2024] titled *Generalized Contrastive Learning for Multi-Modal Retrieval and Ranking* introduces an approach to contrastive learning, which typically relies on binary relevance scores. The researchers curated a large-scale dataset of around 10 million query-document pairs, each annotated with fine-grained relevance scores. This dataset enables more detailed evaluations, especially in cold-start scenarios, where the model is evaluated on completely unseen query-document pairs, which were previously underexplored. The proposed Generalized Contrastive Learning (GCL) framework extends traditional methods by incorporating ranking information into the learning process, rather than solely relying on binary relevance. This framework uses a weighted cross-entropy loss, where weights are derived from relevance scores, allowing the model to prioritize higher-ranking pairs during training. Additionally, GCL supports multi-field training, combining various document attributes (e.g., titles and images) into a unified embedding, which enhances its adaptability to real-world search tasks. Results demonstrated significant improvements, with GCL outperforming existing models by a 94.5% increase in NDCG@10 for in-domain evaluations and 26.3-48.8% improvements in cold-start scenarios. These findings highlight the potential of GCL in applications such as e-commerce and retrieval-augmented generation, showcasing the framework's ability to handle more complex and nuanced retrieval tasks.

Expanding on the integration of multi-modal data retrieval with generative models, Zhao et al. [2023] provides a comprehensive overview of methods that combine various modalities, such as images, code, structured knowledge,

[2]Hashing refers to the process of mapping high-dimensional data into compact binary codes to enable efficient retrieval.

audio, and video, into the generative process. These methods involve retrieving relevant information from different modalities to provide enhanced context or content for generative models. For instance, image retrieval is used to improve text generation in tasks like visual question answering, image captioning, and visually grounded dialogue. The survey emphasizes the relevance of these retrieval-augmented methods in improving the performance and reliability of generative models, particularly in handling diverse and complex information formats, and addresses challenges related to hallucinations and interpretability in large language models.

In the context of domain-specific applications, Wu et al. [2024] introduces a scientific domain-specific multi-modal information retrieval (SciMMIR) benchmark to address the gap in evaluating MMIR models specifically in the scientific domain. The benchmark is constructed using a large dataset of 530K image-text pairs extracted from scientific figures and tables with detailed captions from open-access research papers. The study evaluates the performance of various visual language models (VLMs), including CLIP, BLIP, and BLIP-2, in zero-shot and fine-tuned settings. The results highlight significant challenges in applying general VLMs to scientific data, showing marked improvements only after fine-tuning with domain-specific data. Notably, incorporating Optical Character Recognition (OCR) text from images substantially enhances model performance, particularly in retrieving tabular data. This study emphasizes the need for domain-specific adaptation in MMIR tasks and the importance of OCR in improving VLM capabilities within the scientific context, providing valuable insights for future research in scientific MMIR.

On a specific application of multi-modal retrieval, the study by Fang et al. [2021] introduces the CLIP2Video network, which leverages the pre-trained CLIP model to enhance retrieval accuracy. The methodology focuses on adapting the image-language model for video applications by dividing the task into two key components: a Temporal Difference Block (TDB) to capture motion changes between frames and a Temporal Alignment Block (TAB) to align video clips with textual descriptions. The results show significant improvements in retrieval performance, setting new benchmarks for accuracy in text-to-video and video-to-text retrieval tasks. This approach highlights the importance of temporal dynamics in video retrieval and demonstrates the potential of using pre-trained image-language models in video applications.

Addressing challenges in text-image retrieval, Nguyen et al., 2024 explored the application of learned sparse retrieval (LSR) to text-image retrieval tasks. The study introduced a novel approach that transforms dense vectors from pre-trained dense models (BLIP and ALBEF) into sparse lexical vectors using a lightweight projection layer. This method addresses issues of high dimension co-activation and semantic deviation commonly encountered in multi-modal retrieval. The researchers developed a probabilistic expansion control algorithm using Bernoulli random variables, which effectively mitigated these issues, resulting in models that outperformed state-of-the-art multi-modal LSR models in both efficiency and accuracy while requiring less computational resources. This work contributes to the broader field by offering a scalable and interpretable solution for multi-modal retrieval tasks, with potential applications in reducing the computational footprint of deep learning models.

Focusing on complex document retrieval, Ding et al. [2024] introduce the PDF-MVQA dataset in their paper titled *PDF-MVQA: A Dataset for Multimodal Information Retrieval in PDF-based Visual Question Answering*, specifically designed for research journal articles that span multiple pages and contain various multi-modal elements such as paragraphs, tables, and figures. The study leverages retrieval-based models to enhance document understanding, focusing on the logical and layout relationships across multiple pages. The methodology involves developing frameworks that integrate Vision-Language Pre-trained Models (VLPMs) with multi-modal entity retrieval, employing both Region-of-Interest (RoI)-based and Image Patch-based frameworks. Experimental results demonstrate that Image Patch-based models like ViLT perform better in Exact Matching(EM) and Partial Matching(PM) metrics compared to RoI-based models. The introduction of a joint-grained retriever further improves model performance by incorporating fine-grained textual information, significantly enhancing retrieval accuracy in complex document sections. Their study addresses the challenges of multi-modal data retrieval in visually rich, text-dense documents, offering insights into the development of more robust models for real-world applications.

Building upon these advancements in multi-modal data retrieval, this thesis employs multi-modal data retrieval by integrating text- and image-based retrieval methods within a unified framework. Using a multi-dimensional indexing strategy, both text embeddings and image embeddings are indexed in a structured format, enabling cross-modal retrieval where relevant text and associated images are retrieved together, regardless of the modality of the retrieval method. This is achieved through the joint indexing of embeddings and raw data, ensuring the preservation of relationships between the modalities.

2.3 Chart Data Retrieval

Chart Data Retrieval refers to the process of extracting underlying numerical data from visual representations, such as charts, in digital documents. In recent years, Chart Data Retrieval has evolved significantly, moving from foundational methods focused on basic data extraction to advanced techniques that integrate deep learning, hybrid models, and multi-modal approaches. These advancements have enhanced accuracy, efficiency, and the ability to handle increasingly complex chart types and tasks.

Among these advancements is ChartSense, introduced by Jung et al. [2017], which is a semi-automatic system designed to classify chart types and extract underlying data. Using a convolutional neural network (CNN) model, specifically GoogLeNet, ChartSense improves classification accuracy compared to previous systems like ReVision (automatic chart data extraction tool). After classifying the chart type, ChartSense employs interactive algorithms optimized for each chart type, allowing users to refine the results. This approach proved to outperform WebPlotDigitizer(a computer vision assisted software that helps extract numerical data from images of a variety of data visualizations), particularly in task completion time, error rate, and user satisfaction across various chart types, including bar, line, area, pie, and radar charts. The study emphasizes the importance of mixed-initiative interaction, integrating deep learning with user inputs to enhance both efficiency and accuracy in data extraction from static chart images.

Following this, Dai et al. [2018] presented Chart Decoder, a system that further automates the recovery of textual and numerical information from chart images. Chart Decoder leverages deep learning, computer vision, and text recognition to classify chart types into five categories—bar, pie, line, scatter, and radar—with classification accuracy exceeding 99%. The system extracts and identifies textual elements such as titles and axis labels and focuses primarily on bar charts for graphical data recovery. Chart Decoder demonstrated higher accuracy rates, particularly 77% accuracy for web-collected bar charts and 89% for script-generated ones, significantly surpassing the performance of semi-automatic tools like ChartSense by relying on a larger dataset and more advanced methodologies.

Expanding on these techniques, the study by Luo et al. [2021] titled *ChartOCR: Data Extraction from Charts Images via a Deep Hybrid Framework* introduces a hybrid approach that addresses the limitations of purely rule-based or deep learning methods. ChartOCR combines deep learning and rule-based methods for semantic richness, enhancing accuracy and flexibility in data extraction across different chart types. The framework involves first extracting key points and classifying the chart type, then applying type-specific rules to assemble chart components into structured data. The study demonstrated the effectiveness of this approach across charts and outperformed previous methods in both accuracy and processing speed, demonstrating the benefits of hybrid approaches in overcoming the limitations of single-method systems.

Specific methods targeting particular chart types have also seen significant advancements. One such method is LineFormer, introduced by Lal et al.

[2023], which addresses the challenge of extracting data from complex line charts, particularly those with multiple intersecting lines. Traditional methods based on keypoint detection and low-level feature aggregation often struggle with accuracy due to the visual and structural variations in line charts. LineFormer redefines the problem as an instance segmentation task, treating each line as a distinct object to be segmented pixel-wise. The study employs a transformer-based encoder-decoder architecture to predict binary masks for each line instance, improving robustness in scenarios involving line crossings, occlusions, and crowding. Trained on both synthetic and real datasets, LineFormer achieves state-of-the-art performance on benchmark datasets like UB-PMC, significantly outperforming existing methods like ChartOCR. This approach offers a more reliable framework for line chart data extraction, particularly in complex cases where traditional methods fail[3].

Moving into more advanced and multi-modal approaches, the study by Nowak et al. [2024] explores the challenge of retrieving image-based charts using textual queries by comparing four methodologies:

- **OCR → Text Retrieval**: Converts the chart image into text using an OCR model.
- **Chart DeRendering → Table Retrieval**: DEPLOT, a chart derendering model, converts the chart image into a table.
- **VLM Retrieval**: Uses a vision-language model (e.g., PALI-3) for direct image understanding[4].
- **Chart DeRendering → VLM Retrieval**: Combines DEPLOT and PALI-3 for enhanced retrieval.

The study introduces TAB-GTR, a text retrieval model enhanced with table structure embeddings, which set a new benchmark with a 48.88% R@1 on the NQ-TABLES dataset [Herzig et al., 2021]. The findings reveal that while the DEPLOT approach is more efficient and performs better on simpler data, it struggles with more complex charts. However, the combined approach delivers the best overall results, especially when using multi-task training. This underscores the value of a hybrid strategy that combines both table and image-based information for effective retrieval.

RAG has further enhanced the capabilities of Chart Data Retrieval. The study by He et al. [2024] introduces G-Retriever, a first RAG approach for general textual graphs that integrates LLMs with graph neural networks to

[3] `https://chartinfo.github.io/toolsanddata.html`
[4] `https://huggingface.co/papers/2310.09199`

interact with textual graphs. G-Retriever utilizes Prize-Collecting Steiner Tree (PCST) optimization to efficiently retrieve relevant subgraphs by balancing node relevance and edge minimization. This method reduces hallucination by grounding responses in actual graph data, significantly improving accuracy and efficiency.

Building upon the advancements introduced by the G-Retriever framework, the study by Peng et al. [2024] provides a comprehensive overview of Graph Retrieval-Augmented Generation (GraphRAG), a novel approach that enhances RAG by incorporating graph-structured data. It formalizes the GraphRAG workflow into three stages:

- **Graph-Based Indexing**: Constructs and organizes graph databases for efficient retrieval.
- **Graph-Guided Retrieval**: Extracts relevant subgraphs based on the query.
- **Graph-Enhanced Generation**: Synthesizes responses using the retrieved graph elements.

The survey categorizes existing GraphRAG techniques based on their application to various downstream tasks, such as question answering, information extraction, and fact verification. Through an extensive comparison with related methods, the study highlights how GraphRAG can significantly improve the contextual depth and precision of language model outputs across various domains.

Following an in-depth review of advancements in Chart Data Retrieval, the indexing and retrieval approach in this thesis draws inspiration from the GraphRAG framework, which is also based on structured data. In this thesis, both the charts and their textual descriptions are indexed and then retrieved simultaneously using different retrieval methods. Chart Data Retrieval in our system is based on the VLM Retrieval method, where we utilize the state-of-the-art CLIP model.

2.4 Evidence-Based Question-Answering

Evidence-Based Question-Answering (EBQA) is a specialized form of RAG approach where LLMs generate answers to queries by directly referencing and relying on external, verifiable sources of information. The primary goal of this approach is to ensure that the generated answers are accurate, trustworthy, and traceable to reliable sources. This process is especially critical in contexts where the accuracy and reliability of information is paramount, such as in legal, medical, and scientific domains [Schimanski et al., 2024].

In this method, models are designed or fine-tuned to prioritize citing credible sources and ensuring the information in the generated answers is directly attributable to these sources. The process involves three steps:

1. **Retrieval of Relevant Evidence:** The model first retrieves relevant documents or information snippets from a database or corpus that is likely to contain the needed evidence [Zheng et al., 2024].

2. **Selection of Valuable Information:** The model then selects the most relevant and accurate information from the retrieved documents, filtering out noise or irrelevant content [Zheng et al., 2024].

3. **Answer Synthesis:** The selected evidence is used to generate a coherent and well-supported answer, with citations or references to the original sources [Schimanski et al., 2024].

This approach addresses several challenges inherent to LLMs, such as the tendency to hallucinate or produce information that sounds reasonable but is incorrect or fabricated. By grounding answers in verifiable sources, EBQA reduces the likelihood of such errors and enhances the trustworthiness of the outputs [Schimanski et al., 2024].

2.4.1 Similarities and Differences with RAG

While EBQA and RAG both enhance LLM-generated answers by incorporating external information, EBQA can be seen as a more specific application of RAG with a strong focus on verifiability and attribution. Below are the key similarities and differences between the two:

Similarities

- **Use of External Knowledge:** Both approaches rely on retrieving external information to augment the LLM's internal knowledge, which is crucial for generating more accurate and contextually appropriate answers [Schimanski et al., 2024, Zheng et al., 2024].

- **Reduction of Hallucinations:** By grounding the generated text in external data, both methods aim to mitigate the risk of hallucinations, improving the reliability of the generated answer [Huo et al., 2023, Zheng et al., 2024].

Differences

- **Focus on Evidence and Attribution:** EBQA places a stronger emphasis on ensuring that the generated content is directly attributable to specific, cited sources, which is critical in domains that require verifiability, such as legal or scientific question answering. RAG, while it also relies on retrieval, may not always emphasize strict attribution and can integrate retrieved information more loosely [Pride et al., 2023, Schimanski et al., 2024].

- **Handling of Retrieved Information:** In EBQA, the system goes through a more rigorous process of filtering and validating the retrieved information before using it in the final answer. RAG integrates the retrieved content more directly into the answer generation process, without rigorous process of filtering [Huo et al., 2023, Zheng et al., 2024].

In a targeted effort to enhance the faithfulness and traceability of Large Language Models (LLMs) in Evidence-Based Question Answering (EBQA), Schimanski et al. [2024] address key challenges such as citing the correct sources (source quality) and accurately representing the information from those sources (answer attributability). The study introduces a novel data generation pipeline with automated quality filters, enabling the synthesis of diversified, high-quality datasets for training and testing at scale. By focusing on improving data quality rather than quantity, the researchers demonstrate that fine-tuning LLMs on this curated synthetic data leads to significant performance improvements in both in-distribution (when testing data closely matches training data) and out-of-distribution scenarios (where the model is tested on unfamiliar data). To benchmark the robustness of these fine-tuned models, four test sets were introduced, and extensive evaluations show that the proposed pipeline and quality filters play a crucial role in enhancing EBQA performance. This scalable approach effectively addresses current limitations, improving the reliability and applicability of LLMs in evidence-based systems across diverse real-world applications.

In conclusion, this chapter reviewed key advancements in RAG, Multi-Modal Data Retrieval, Chart Data Extraction techniques, and Evidence-Based Question Answering. This thesis contributes by streamlining the data indexing process, embedding structured datasets without chunking, and using a multi-dimensional indexing strategy. A multi-modal retrieval framework is employed, integrating text and image embeddings capable of simultaneous cross-modal retrieval. Additionally, an evaluation framework was designed, combining quantitative metrics and human assessments to ensure a more robust evaluation of the system. Finally, advanced generative models are used to generate answers from cited sources, aligning with EBQA principles without requiring extensive fine-tuning.

Chapter 3

Methodology

This chapter presents the design and implementation of the Chart Retrieval Framework, a multi-modal Retrieval-Augmented Generation (RAG) system. Section 3.1 introduces the Chart Retrieval Framework, outlining its client-server architecture and the integration of Large Language Models (LLMs) and Elasticsearch to support both sparse and dense retrieval methods. Section 3.2 discusses the LLMs, which are used to create text and image embeddings and generate answers. In Section 3.3, the Indexing Process is briefly covered, focusing on how the dataset is structured for effective retrieval using raw and vector-based indexing methods. Section 3.4 provides an overview of the Retrieval Process, which employs both sparse and dense retrieval to match user queries with the most relevant charts and their descriptions. Section 3.5 introduces the Generation Process, where LLMs generate responses based on the retrieved documents. The dataset is discussed in Section 3.6, which includes a range of charts from Statista and Pew Research, enhanced with available descriptions and machine-generated captions for multi-modal retrieval. Finally, Section 3.7 presents the Query Selection and Pre-processing, covering queries sourced from the Touché dataset, manually created queries, and randomly generated ones used to evaluate the system's performance. Together, these sections offer a structured overview of the system, highlighting the integration of multi-modal chart retrieval and LLMs to generate relevant and faithful answers to the given query.

3.1 Chart Retrieval Framework

The Chart Retrieval Framework is a custom-designed multi-modal RAG system that retrieves charts with their descriptions through various retrieval methods and generates answers to user queries. This section outlines the system's design, the roles of different components, and their interactions. The framework's pipeline, as illustrated in Figure 3.1, comprises three key processes: indexing, retrieval, and generation. Each process plays a crucial role in ensuring that the system can efficiently retrieve and generate relevant answers to user queries based on multi-modal content.

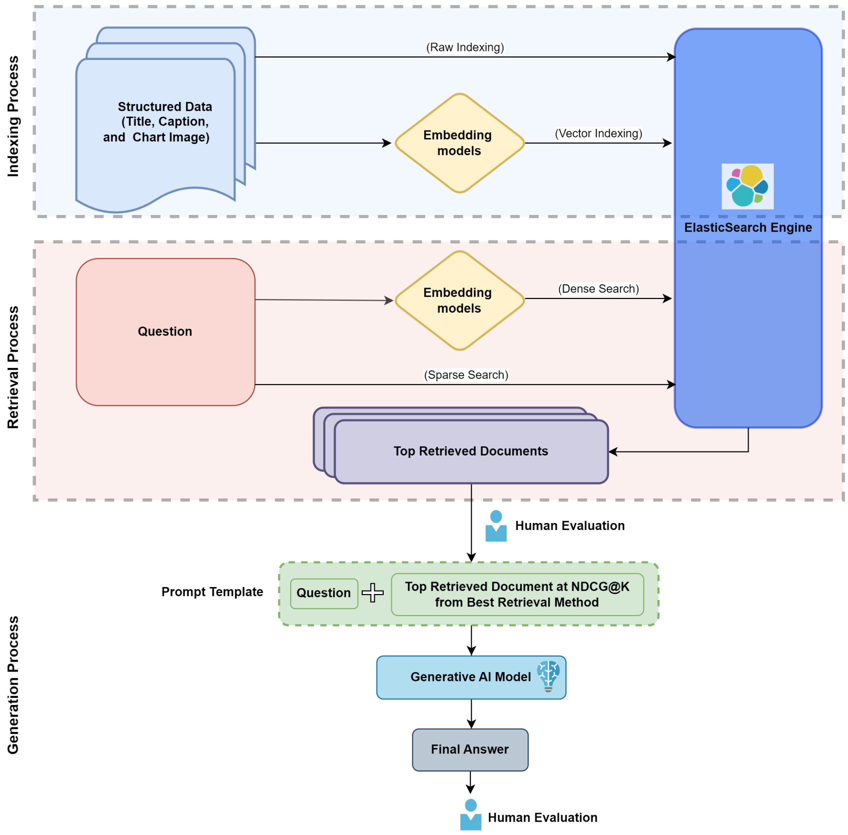


Figure 3.1: The chart retrieval pipeline proposed in this thesis comprises three processes: indexing, retrieval, and generation.

3.1.1 Overview

The system operates on a client-server architecture where the front-end interacts with the server to perform various operations, such as writing queries, selecting predefined queries, generating embeddings, searching, retrieving documents, conducting evaluations, and generating answers from large language models (LLMs). The back-end is built using the Flask web framework, with Elasticsearch serving as both the database and the search engine for document retrieval. The system integrates multiple models, including text-based and vision models, which contribute to the retrieval process by generating embeddings for similarity searches against corresponding pre-indexed embeddings in Elasticsearch. For the generation process, the system utilizes advanced generative models to produce detailed and relevant answers to the given query

3.1.2 Application Back-end

The back-end of the system is built using Flask, a lightweight web framework in Python. It manages the interactions between the user interface, models, Elasticsearch, and external APIs for LLMs. It serves as the communication bridge, ensuring that user requests are processed and the appropriate responses are returned. The back-end communicates with various services through REST endpoints, which are specific URLs that define how different components of the system can interact via HTTP requests.

Key responsibilities of the back-end:

- **Model Integration:** Integrates the embedding models.
- **Query Processing:** Processes user queries by either using them directly for search or converting them into embeddings via the integrated embedding models.
- **Search and Retrieval:** Executes search queries in Elasticsearch and retrieves the most relevant documents using different retrieval methods.
- **Metric-Based Evaluation:** Utilizes manually evaluated scores from the front-end to determine the most effective retrieval method.
- **LLM Integration:** Communicates with external APIs to generate answers based on the given query and documents retrieved from the best retrieval method.

3.1.3 Front-end Interface

The front-end of the system is built using HTML, CSS, and JavaScript. It provides an interactive interface for users to input queries, select queries, view results,and evaluate the retrieval results and generated answers. The user interface is designed to make the search and retrieval process intuitive.

Key responsibilities of the front-end:

- **Query Selection:** Users can select from predefined queries or enter custom ones manually.

- **Result Display:** Displays all retrieved documents on a web page after a query is submitted for search. The retrieval methods are hidden to ensure unbiased human evaluation. Each document includes a title, content, and chart.
- **Human Evaluation:** Users evaluate the displayed results against the query based on predefined metrics.
- **Top Result Display:** After evaluating all retrieved documents, the top retrieved document from the best retrieval method is shown, which is provided to the LLMs along with a prompt.
- **LLM Answer Display:** Displays the answers generated by the LLMs, with the LLM name hidden.
- **Answer Evaluation:** Users assess the quality of the LLM-generated answers against predefined metrics, based on the information from the top retrieved document provided to the LLMs.

Query Selection, Top Result Display, LLM Answer Display, and Answer Evaluation occur on the main web page (see Figure A.2), while Result Display and Human Evaluation occur on a second web page (see Figure A.1).

3.2 Large Language Models

Large Language Models (LLMs) have significantly advanced the field of natural language processing (NLP) by enabling machines to understand, generate, and manipulate human language with remarkable accuracy and depth. These models, characterized by their vast number of parameters often reaching into billions are trained on extensive datasets, allowing them to capture intricate linguistic nuances, contextual information, and semantic relationships between words [Naveed et al., 2024] [Raiaan et al., 2024].

The architecture of LLMs is primarily based on the Transformer model, introduced by Vaswani et al. [2017]. The Transformer architecture utilizes a self-attention mechanism that enables the model to weigh the importance of different words in a sentence, capturing long-range dependencies and contextual relationships more effectively than previous models like Recurrent Neural Networks (RNNs) and Long Short-Term Memory networks (LSTMs) [Zhang and Shafiq, 2024].

Key Components

- **Tokenization:** This process involves breaking down text into smaller units or tokens, which are then used as inputs to the model. Advanced tokenization techniques like Byte Pair Encoding (BPE), UnigramLM, and WordPiece are often employed to handle a diverse vocabulary efficiently [Zhang and Shafiq, 2024].

- **Embeddings:** These are continuous vector representations of tokens that capture semantic information, allowing the model to understand subtle contextual nuances [Zhang and Shafiq, 2024].

- **Attention Mechanism:** The self-attention mechanism is crucial for LLMs, enabling them to process and generate text with high contextual awareness by considering the relationships between all tokens in a sequence [Raiaan et al., 2024].

- **Pre-training and Fine-tuning:** LLMs are typically pre-trained on large corpora to learn general linguistic patterns and world knowledge. They are then fine-tuned on specific tasks to enhance their performance in particular applications [Naveed et al., 2024].

Capabilities and Applications

LLMs have demonstrated impressive capabilities across a wide range of NLP tasks:

- **Text Generation:** They can generate coherent and contextually relevant text, making them useful for content creation, including articles, stories, and product descriptions [Naveed et al., 2024].

- **Language Translation:** LLMs can translate text between languages with high accuracy, facilitating communication across linguistic barriers [Raiaan et al., 2024].

- **Summarization and Classification:** These models can efficiently summarize lengthy documents and classify text based on predefined categories, aiding in tasks like sentiment analysis and content moderation [Raiaan et al., 2024].

- **Conversational Agents:** LLMs power virtual assistants and chatbots, enabling them to understand user queries and engage in natural language conversations [Naveed et al., 2024].

In this thesis, LLMs are employed in the RAG system for two key tasks: embedding generation for indexing and retrieval, and text generation for the final answer. Specifically, we have selected *E5-Mistral-7B-Instruct* and *gte-Qwen2-7B-Instruct* to create high-quality text embeddings, while *CLIP* to create image embeddings. For answer generation, *GPT-4o* and *Meta-Llama-3.1-70B-Instruct* are selected. The following subsections (3.2.1, 3.2.2, 3.2.3, 3.2.4, 3.2.5) provide technical details and the motivation behind the selection of these LLMs.

3.2.1 E5-Mistral-7B-Instruct

The *E5-Mistral-7B-Instruct* is a sophisticated model designed to generate high-quality text embeddings. Fine-tuned on a diverse set of multilingual datasets, this model enhances its capacity to provide single-vector text representations for a wide range of NLP tasks, including retrieval, clustering, classification, and semantic textual similarity. This model is the extend of the EmbEddings from bidirEctional Encoder rEpresentations(E5) family of models, which are known for their effectiveness in producing embeddings through contrastive learning techniques that helps them generate effective text representations, making them suitable for zero-shot or fine-tuned applications without the need for extensive labeled data [Wang et al., 2023]. It is particularly well-suited for tasks that require efficient text similarity and retrieval.

Technical Details of the Model

The *E5-Mistral-7B-Instruct* model is built on the Mistral-7B-v0.1 architecture, consisting of 32 layers with an embedding size of 4096. It can process input sequences up to 4096 tokens, making it ideal for embedding longer documents in retrieval tasks. The training utilizes contrastive learning and the Information Noise Contrastive Estimation(InfoNCE) loss function to effectively distinguish between relevant and irrelevant text pairs by measuring cosine similarity [Wang et al., 2022][Jiang et al., 2023].

The architecture uses pooling techniques to derive embeddings from the last token in a sequence, which are then normalized for consistency across tasks. Additionally, it uses task-specific prompts during query encoding, a process known as instruction tuning, to customize embeddings for specific tasks. This approach ensures high-quality embeddings tailored to diverse applications, such as search queries and semantic similarity tasks[1].

[1] `https://huggingface.co/intfloat/e5-mistral-7b-instruct`

Motivation for Model Selection

The model was selected for generating text embeddings due to its strong performance across benchmarks like Benchmarking Information Retrieval (BEIR) and Massive Text Embedding Benchmark (MTEB), where it outperforms many of its competitors [Wang et al., 2022][Muennighoff et al., 2022]. Additionally, its ability to handle long input sequences (up to 4096 tokens) makes it especially suitable for generating embeddings of large textual contents.

3.2.2 gte-Qwen2-7B-instruct

The *gte-Qwen2-7B-instruct* is a cutting-edge text embedding model from the GTE (General Text Embedding) family. Released in 2024, this model is based on the Qwen2-7B model and excels in generating high-quality embeddings across both English and Chinese tasks. It ranks among the top 5 models on the MTEB, outperforming its predecessor *gte-Qwen1.5-7B-instruct*. The model incorporates advancements from the Qwen2 series, designed to improve contextual understanding and effectiveness in handling diverse tasks[2].

Technical Details of the Model

Built upon Qwen2-7B, this model contains 7 billion parameters and offers an embedding dimension of 3584, supporting up to 32,000 input tokens. This large capacity allows it to process substantial amounts of text efficiently. A core feature of the model is its bidirectional attention mechanism, which enhances the depth of contextual comprehension by analyzing relationships between tokens across entire texts [Li et al., 2023]. This capability is especially beneficial for tasks requiring complex text representations.

Similarly to the *E5-Mistral-7B-Instruct*, this model also employs pooling and instruction tuning techniques. The instruction tuning is focused solely on queries, streamlining the process for retrieval tasks without sacrificing performance.

Motivation for Model Selection

This model was selected for its exceptional performance across multiple benchmarks, ranking among the top models on MTEB [Li et al., 2023]. Its ability to handle long input sequences, up to 32,000 tokens, makes it ideal for generating embeddings of large textual content without the need for chunking.

[2] `https://huggingface.co/Alibaba-NLP/gte-Qwen2-7B-instruct`

3.2.3 Contrastive Language-Image Pre-training

The *Contrastive Language-Image Pre-training (CLIP)* model, developed by OpenAI, offers a significant advancement in learning image representations by leveraging natural language. Unlike traditional computer vision models that rely on manually labeled datasets with predefined categories. Instead, *CLIP* is pre-trained on image-text pairs, allowing it to generalize across tasks through zero-shot learning. This enables *CLIP* to predict new categories it hasn't encountered during training, making it highly adaptable for tasks such as image classification, retrieval, and fine-grained object recognition [Radford et al., 2021].

Technical Details of the Model

In this thesis, the *CLIP-ViT-Large-Patch14* variant is used, which employs a Vision Transformer (ViT) as the image encoder. This architecture uses 14x14 input patch sizes, improving spatial understanding of images by capturing long-range dependencies and complex patterns, surpassing the performance of traditional convolutional networks like ResNet. The model processes images by breaking them into patches and learning their relationships through self-attention mechanisms. On the text side, a masked self-attention Transformer encodes textual descriptions. Both image and text encoders are trained jointly using a contrastive loss function, aligning image and text embeddings in a shared latent space. This training, conducted on a dataset of 400 million image-text pairs, contributes to *CLIP's* ability to generalize across diverse tasks [Radford et al., 2021].

Motivation for Model Selection

The model was selected for its ability to generate semantically rich image embeddings, which are essential for tasks requiring fine-grained visual representation. Its Vision Transformer architecture processes high-resolution images effectively, capturing intricate details. *CLIP's* unique alignment of visual and textual information further enhances its ability to retrieve images based on natural language queries.

3.2.4 GPT-4o

GPT-4o, also known as ChatGPT-4omni, developed by OpenAI is designed to process text, audio, and visual data simultaneously. This multi-modal capability allows for more natural human-computer interaction, enhancing communication and data privacy across various applications [Shahriar et al., 2024].

Technical Details of the Model

GPT-4o builds upon a transformer-based architecture, excelling in language, vision, and speech processing. By processing all inputs: text, audio, images, and video within the same neural network, *GPT-4o* overcomes limitations of prior models that used separate pipelines for different modalities. This integrated approach enables the model to capture nuances such as tone and context across modalities, while its proficiency in few-shot learning allows it to adapt to new tasks quickly with minimal data. High performance is demonstrated across tasks like language translation, image classification, and object recognition [Shahriar et al., 2024]. A standout feature is Temporary Chat, which enhances privacy during interactions, making it particularly suited for sensitive applications [Temsah et al., 2024].

Motivation for Model Selection

GPT-4o's multi-modal capabilities enable it to generate accurate and contextually appropriate responses across various domains. In this study, text and image inputs need to be processed efficiently, ensuring quick responses with low latency. The model is optimized for cost-effective API usage, providing a practical, plug-and-play solution that offers seamless integration without significant overhead. Furthermore, it features a context window of 128,000 tokens and a maximum output of 4,096 tokens, allowing for the handling of large documents[3].

3.2.5 Meta-Llama-3.1-70B-Instruct

The *Meta-Llama-3.1-70B-Instruct*[4] is part of Meta's Llama 3 series, designed to handle a wide range of AI tasks such as multilingual processing, coding, and reasoning. This 70 billion-parameter model is specifically optimized for instruction-following, showcasing Meta's dedication to advancing AI while ensuring alignment with human values [Dubey et al., 2024].

Technical Details of the Model

With a dense Transformer architecture featuring 80 layers, 8192 model dimensions, and 64 attention heads, the model supports a context window of up to 128K tokens, enabling it to manage extensive text sequences. It has been pre-trained on a diverse corpus of 15 trillion tokens, significantly enhancing its ability to understand and generate language.

[3] `https://platform.openai.com/docs/models/gpt-4o`
[4] `https://huggingface.co/meta-llama/Meta-Llama-3.1-70B-Instruct`

The model undergoes both pre-training, which focuses on learning language structures through next-token prediction, and post-training, which finetunes its instruction-following abilities using methods like supervised finetuning (SFT) and Direct Preference Optimization (DPO). This two-stage training approach ensures the model not only grasps language but also responds effectively to human instructions [Dubey et al., 2024].

Motivation for Model Selection

This model was selected for response generation due to its advanced capabilities in understanding and generating human-like text. The model's instruction-following capabilities are enhanced through rigorous post-training processes, making it adept at generating responses that align with user prompts and expectations. Like *GPT-4o*, the model is optimized for cost-effective API usage, offering seamless, plug-and-play integration with minimal overhead.

Moreover, the integration of safety measures during post-training ensures that the model can generate helpful and harmless responses, which is crucial for maintaining user trust and satisfaction in AI-driven interactions [Dubey et al., 2024].

3.2.6 Quantization

Quantization is a technique used to represent data with fewer bits, which reduces memory usage and accelerates inference, particularly in LLMs. This process is essential when dealing with models that have billions of parameters, as it enables efficient use of computational resources[5]. In this thesis, quantization was applied to *E5-Mistral-7B-Instruct* and *gte-Qwen2-7B-instruct* because the size of the model was large and required optimization for both memory and speed.

The BitsAndBytes library was used to quantize the model to 4-bit precision[6]. This approach maintained a balance between reducing the memory footprint and preserving the model's performance.

3.3 Indexing process

This section discussed the in-depth understanding of the indexing process for structured tabular data, particularly in applications requiring multi-dimensional information retrieval. The process integrates Elasticsearch as a key vector database, enabling fast and efficient retrieval in response to complex queries.

[5] `https://huggingface.co/docs/peft/en/developer_guides/quantization`
[6] `https://github.com/bitsandbytes-foundation/bitsandbytes`

3.3.1 ElasticSearch

ElasticSearch is a robust, open-source search and analytics engine designed to handle large volumes of data in real-time. Built on Apache Lucene libraries, it is widely used for full-text search, event data analysis, and as a general-purpose data store. ElasticSearch's primary purpose is to provide fast and scalable search capabilities, which are crucial for indexing and searching structured datasets. Its distributed architecture allows for horizontal scaling, making it suitable for handling large datasets and complex queries efficiently [Majumdar, 2022].

Why ElasticSearch?

- **Vector Search Capabilities**: ElasticSearch enables advanced search functionality beyond traditional keyword matching, supporting semantic search through vector embeddings. This allows for more context-aware and nuanced data retrieval, ideal for complex queries across structured and unstructured datasets [Ni et al., 2024].

- **Customizable Mappings**: ElasticSearch offers flexibility in defining custom mappings and schemas, which optimize the storage and querying of structured tabular data. This feature enhances performance by tailoring indexing processes to the specific characteristics of the data [Ni et al., 2024].

- **Real-Time Data Handling**: ElasticSearch efficiently manages real-time data ingestion and querying, ensuring that indexed data remains up-do-date. This capability is crucial for research involving frequently updated datasets, such as the Statista or Pew datasets [Ni et al., 2024].

- **Scalability and Performance**: It is designed to handle large volumes of data, with effective indexing and search algorithms that ensure fast and responsive performance, making it suitable for complex queries [Majumdar, 2022].

3.3.2 Handling of Data

Data Cleaning

Before cleaning the data, the Pew and Statista datasets were filtered to retain only the title, caption, and image columns. These filtered datasets were then combined into a single dataset to facilitate easier comparison and analysis across both sources. Additionally, the Llava-generated captions were integrated into this combined dataset, ensuring they were aligned with the corresponding title, caption, and chart entries.

Text cleaning procedures were applied to resolve common issues in large text datasets, such as replacing newline characters with spaces for improved readability. During this process, some encoding issues were also identified in the text due to characters encoded using `ISO-8859-1` (Latin-1). When the data was read in `UTF-8`, certain characters were displayed incorrectly, resulting in misinterpreted symbols (e.g., â€“ instead of -). To resolve this, a systematic approach was used to identify and correct these misinterpreted characters by replacing them with their proper equivalents.

Example:

- **Before correction:**
  - The report states that â€œdata integrity is essentialâ€.
  - This issue occurred in 2020â€“2021.
- **After correction:**
  - The report states that "data integrity is essential".
  - This issue occurred in 2020-2021.

This approach ensured the text was accurately cleaned and formatted for further analysis.

No Chunking Approach

The decision to forgo chunking in this thesis was a deliberate choice based on the structured, multi-dimensional nature of the dataset. Traditional embedding techniques typically involve dividing non-tabular data, such as documents, into smaller chunks, generating embeddings for each chunk, and indexing them in a vector database. However, this method is not well-suited for structured, tabular data, where rows represent distinct units of information that can vary in format and data type [Khanna and Subedi, 2024].

Given the structured nature of our dataset, chunking would introduce several challenges. First, for large structured datasets, chunking would create unnecessary redundancies and increase the complexity of indexing and querying without adding meaningful value to the analysis. Additionally, sending these chunks to an LLM during RAG would quickly exceed the model's context window, limiting its ability to effectively analyze relevant information [Khanna and Subedi, 2024].

By preserving the data in its original structure and applying targeted cleaning and transformation methods, we avoided the inefficiencies associated with chunking, ensuring that the data remained organized, clean, and accessible for further analysis.

3.3.3 Indexing

The indexing process in this thesis involves transforming the data into a format that can be efficiently retrieved and queried in Elasticsearch. It follows a multi-dimensional indexing strategy, with two distinct approaches: raw indexing and vector-based indexing. Raw indexing directly stores the textual and image data ensuring that the original content is preserved. On the other hand, vector-based indexing leverages specific LLMs to enhance retrieval accuracy and efficiency by generating embeddings. The structure of the indexing process ensures that both textual and visual data are represented in a searchable, semantically rich format.

Raw Indexing

In raw indexing, the primary elements of each document - title, caption, Llava-generated caption, and associated chart are directly indexed without transformation into vector representations. The process begins by concatenating the title and caption into a single, coherent text string. This combined text serves two purposes: first, it uniquely identifies the document, and second, to provide the basis for further indexing and querying operations. To ensure efficient retrieval and prevent duplicates, a unique identifier for each document is generated using the SHA-256 hash function[7]. The chart associated with the document is encoded in base64 format, allowing it to be stored compactly alongside its textual metadata.

When querying this raw indexed data, sparse retrieval methods are applied. These methods rely on keyword-based matching, where the query must contain exact terms or phrases present in the indexed document.

The following components are indexed using the raw indexing approach:

- **Title**: The name or main heading of the document or image.
- **Caption**: A brief description associated with the image.
- **Llava-generated Caption**: A machine-generated caption of the image that provides additional context.

[7] `https://www.elastic.co/guide/en/ecs/current/ecs-hash.html#field-hash-sha256`

- **Image data**: The base64-encoded version of the chart.

This initial phase ensures that each document in the database is comprehensively indexed with both human-provided and machine-generated captions, along with the associated chart.

Vector-based Indexing

Vector indexing enhances search capability by converting textual and visual information into embeddings, a dense numerical representation using advanced models. This process enables semantically rich querying, allowing documents to be retrieved based on the meaning and context of the input query, rather than relying on exact keyword matches.

The following components are indexed using a vector-based indexing approach, leveraging different models for text and image embeddings:

- **Combined-text-mistral**: The combined text (title + caption) is embedded using the *E5-Mistral-7B-Instruct* model. The input text length is capped at 1,250 tokens (max_length=1250) to ensure efficient processing and consistency across inputs. The same unique identifier, generated using the SHA-256 hash function (as in raw indexing), is used to maintain consistency across both raw and vector-based indexing. The resulting embedding is then indexed alongside the raw indexed data.

- **Combined-text-gte**: Similarly, *GTE-Qwen2-7B-Instruct* model is employed to generate embeddings for the same combined text (title + caption). The same input text length limit of 1,250 tokens is used, with the same unique identifier ensuring consistency.

- **Llava-generated-text-mistral**:The combined text (title + Llava-generated caption) is embedded using the *E5-Mistral-7B-Instruct* model. The input text length is capped at 3,070 tokens, as the Llava-generated caption tends to be longer. The same unique identifier, generated using the SHA-256 hash function, is used to ensure consistency across indexing.

- **Image-embedding-clip**: Each chart is embedded using the *CLIP* model. The same unique identifier is used to ensure consistency across indexing. The resulting image embedding is then indexed alongside the other indexed data in Elasticsearch.

Indexed Database Structure

The table 3.1 provides a detailed breakdown of the fields and their associated configurations.

Field	Type	Dimensions	Analyzer/Index Options
Title	text	-	custom_english_analyzer
Caption	text	-	custom_english_analyzer
Llava-generated Caption	text	-	custom_english_analyzer
Image data	text	-	-
Combined-text-mistral	dense_vector	4096	cosine (int8_hnsw, m=16, ef_construction=100)
Combined-text-gte	dense_vector	3584	cosine (int8_hnsw, m=16, ef_construction=100)
Llava-generated-text-mistral	dense_vector	4096	cosine (int8_hnsw, m=16, ef_construction=100)
Image-embedding-clip	dense_vector	768	cosine (int8_hnsw, m=16, ef_construction=100)

Table 3.1: Indexed Database Structure

To better understand the indexing and search methodologies used, the following key components are outlined:

- **custom_english_analyzer**: The custom_english_analyzer in Elasticsearch is a custom-configured analyzer designed for processing English text. It uses the standard tokenizer[8] to break text into terms based on word boundaries. It applies two filters: lowercase to make searches case-insensitive, and english_stop[9] to remove common English stop words using Elasticsearch's built-in _english_ stopwords list. This setup enhances search accuracy and performance by normalizing text and excluding unnecessary words.

- **int8_hnsw**: The int8_hnsw refers to an implementation of the Hierarchical Navigable Small World (HNSW) algorithm for efficient approximate nearest neighbor search, that optimizes memory usage by compressing the vectors into 8-bit integers (int8). This method significantly reduces the memory footprint without compromising the accuracy of vector-based searches, making it highly efficient for large-scale databases [Malkov and Yashunin, 2018].

- **m=16**: In the context of HNSW, m refers to the number of bi-directional edges that each node maintains in the graph[10].A higher value for m

[8] `https://www.elastic.co/guide/en/elasticsearch/reference/current/analysis-standard-tokenizer.html`

[9] `https://www.elastic.co/guide/en/elasticsearch/reference/current/analysis-stop-analyzer.html`

[10] `https://www.elastic.co/guide/en/elasticsearch/reference/current/dense-vector.html`

results in more connections between nodes, which can improve accuracy but may also slow down query processing [Malkov and Yashunin, 2018].

- **ef_construction=100**: The *ef_construction* parameter controls the number of neighbors explored during the insertion phase of the HNSW graph[10]. Higher values improve search accuracy by ensuring better connectivity and recall but increase graph construction time. It balances search quality and construction efficiency [Malkov and Yashunin, 2018].In ElasticSearch, the default value for *ef_construction* is 100 for the hnsw, int8_hnsw, and int4_hnsw index types.

3.4 Retrieval Process

In this thesis, two retrieval approaches are employed: Sparse Retrieval and Dense Retrieval, both implemented using Elasticsearch as the core search engine. Each approach is carefully designed to optimize document retrieval based on the characteristics of the query and the indexed fields. Cross-modal retrieval is enabled, allowing relevant text and associated chart to be retrieved simultaneously, irrespective of the retrieval method's modality. Depending on the retrieval approach, specific fields such as the *title*, *caption*, *Image data*, or *Combined-text-mistral* are targeted for keyword-based searches or similarity searches. Each retrieved document is then displayed with its title, content, and associated chart.

3.4.1 Sparse Retrieval

Sparse Retrieval refers to the traditional retrieval approach that primarily focuses on keyword matching within a document corpus. One of the most common algorithms used in sparse retrieval is Best Matching 25 (BM25), an extension of the probabilistic information retrieval model[11]. BM25 ranks documents based on the term frequency (TF) and inverse document frequency (IDF) of query terms within the documents, while adjusting for document length. This approach enables BM25 to prioritize documents that contain more frequent and rare query terms, effectively balancing relevance and comprehensiveness.

BM25 was applied in two different methods:

- The first method, **BM25 (Title+Caption)**, applies BM25 scoring to

[11] `https://www.elastic.co/blog/practical-bm25-part-2-the-bm25-algorithm-and-its-variables`

both the *title* and *caption* fields using Elasticsearch's multi_match query[12]. It retrieves the top three most relevant documents by evaluating how well the user's query matches terms in either field. Using best_fields mode, the highest scoring field primarily determines the document ranking, while a tie-breaker ensures that if both fields match, the second-best field contributes 30% to the overall score. This approach prioritizes documents with strong relevance in both fields while also accounting for documents where only one field matches effectively.

- The second method, **BM25 (Title+LLaVA Caption)**, applies the BM25 scoring approach to the *title* and *LLaVA-generated caption* fields. The top three most relevant documents are retrieved in the same manner as the first method, based on how well the user's query matches terms in these specific fields.

These BM25-based retrieval methods provide a baseline for retrieval by prioritizing documents that explicitly match the query terms. However, this retrieval approach struggles with semantic similarity, which is where dense retrieval provides a complementary solution.

3.4.2 Dense Retrieval

In the Dense Retrieval approach, transformer-based models discussed in section 3.2 were employed to generate query embeddings, facilitating similarity searches within pre-indexed data stored in Elasticsearch. The query embeddings were compared against indexed embeddings, which were also generated using the same models, ensuring consistency within the semantic space. In this thesis, three distinct models were used in four dense retrieval methods, each employing a unique approach to generate and compare embeddings.

- The first method, **e5-Mistral (Title+Caption)**, involved using the *e5-mistral-7b-instruct* model to create embeddings of the query. These embeddings were then matched against the indexed field *Combined-text-mistral*, which consisted of combined embeddings of titles and captions, also generated by the same model. The system retrieved the top three documents based on this similarity search.
- In the second method, **e5-Mistral (Title+LLaVA Caption)**, the same *e5-mistral-7b-instruct* model was used to embed the query. However,

[12] `https://www.elastic.co/guide/en/elasticsearch/reference/current/query-dsl-multi-match-query.html`

the similarity search was performed against the indexed field *Llava-generated-text-mistral*, which comprised titles combined with captions generated by the Llava model. Once again, the search retrieved the top three documents.

- The third method, **GTE-Qwen2 (Title+Caption)**, involved the *gte-Qwen2-7B-instruct* model, which was used to generate embeddings for the queries. These were then compared with embeddings in the indexed field *Combined-text-gte*, a combination of title and caption embeddings generated using the same *gte* model. This method also retrieved the top three relevant documents.

- Lastly, the fourth method, **CLIP (Image Embedding)**, employed the *CLIP* model to generate query embeddings, focusing on image embeddings stored in the *Image-embedding-clip* field. In this case, the model compared the query embeddings with the indexed image embeddings, retrieving the top three relevant documents.

By utilizing these methods, the retrieval process ensured that both textual and visual content could be matched and retrieved based on the corresponding embeddings generated for each query. All these retrieved documents are displayed in a dedicated web page, with the retrieval methods hidden for the human evaluation.

3.5 Generation Process

The generation process involved two distinct methods, each utilizing a different language model. This process was preceded by document retrieval, where the system searched for relevant documents based on the query inputs. The retrieved documents, which included the title, caption, and chart, were evaluated through human judgment. The effectiveness of the retrieval methods was measured using NDCG scores at different levels (NDCG@1, NDCG@2, and NDCG@3).

For each NDCG level, the top retrieved documents from the best-performing retrieval method were selected and structured as part of the input payload. This payload included the query, a prompt, and the retrieved documents, which were passed to the respective models—*GPT-4o* and *Llama 3.1* - via an API call. The models then generated answers based on the provided input, delivering comprehensive and relevant insights aligned with the given query.

3.5.1 Llama 3.1-Based Generation

The first method employs the *Meta-Llama-3.1-70B-Instruct* model, which processes the text-only data (titles and captions) of the retrieved documents without incorporating the associated chart. A key aspect of this approach is how the structured payload is formulated. It includes the prompt, query, and relevant text-only documents, with the evaluation metric determining the number of documents to include. For instance, the top retrieved document is used for NDCG@1, the top two documents for NDCG@2, and the top three for NDCG@3. This allows the model to process varying amounts of input data based on the ranking criteria, ensuring that the analysis adapts to the importance or relevance of each document. Samples of the structured payloads used for NDCG@1 and NDCG@2 are presented in Figures 3.2 and 3.3, respectively.

```
{
    "model": "meta-llama/Meta-Llama-3.1-70B-Instruct",
    "messages": [
        {
            "role": "system",
            "content": "You are an expert statistical analyst. Answer the
                ↪ given query with a detailed and comprehensive statistical
                ↪ insight from the following title and content. Format the
                ↪ response in the following structure with 3 paragraphs,
                ↪ without paragraph title:
                1. Start the response with a clear classification or a
                    ↪ straightforward answer to the query.
                2. Provide supporting findings and detailed analysis,
                    ↪ including relevant statistical data.
                3. Summarize the final conclusion briefly."
        },
        {
            "role": "user",
            "content": "Query: {query}
                        Title1: {title}
                        Content1: {caption}"
        }
    ],
    "max_tokens": 1000
}
```

Figure 3.2: Payload sample for Meta-Llama 3.1 at NDCG@1

In the *Llama 3.1* based method, the system role provides the model with specific instructions on how to generate the response. The system prompt establishes the context by designating the model as an expert statistical analyst and provides a structured prompt for the response. The model is instructed to deliver the answer in three paragraphs:

- The first offers a classification or direct response to the query.
- The second provides detailed analysis with supporting data.

```
{
    "model": "meta-llama/Meta-Llama-3.1-70B-Instruct",
    "messages": [
        {
            "role": "system",
            "content": "You are an expert statistical analyst. Answer the
                ↪ given query with a detailed and comprehensive statistical
                ↪ insight from the following title and content. Format the
                ↪ response in the following structure with 3 paragraphs,
                ↪ without paragraph title:
                1. Start the response with a clear classification or a
                    ↪ straightforward answer to the query.
                2. Provide supporting findings and detailed analysis,
                    ↪ including relevant statistical data.
                3. Summarize the final conclusion briefly."
        },
        {
            "role": "user",
            "content": "Query: {query}
                        Title1: {title}
                        Content1: {caption}
                        Title2: {title2}
                        Content2: {caption2}"
        }
    ],
    "max_tokens": 1000
}
```


Figure 3.3: Payload sample for Meta-Llama 3.1 at NDCG@2

- The third offers a brief conclusion[13].

The user role provides the model with the query and the text-only data (titles and captions). This clear division of roles ensures that the model follows a well-defined path in generating its insights. By focusing solely on text-only inputs, this method is ideal for scenarios where visual data does not add value. The structured design of this method ensures that the statistical analysis is precise and easy to follow.

3.5.2 GPT-4o-Based Generation

In contrast to the text-only approach of *Llama 3.1*, the second method employs the *GPT-4o* model, which integrates both text and visual data to provide a more comprehensive analysis. This method utilizes all components of the

[13]Note: For the first 15 queries, a slightly different prompt was used for the third paragraph: "Summarize the final conclusion briefly. If the query does not specify a country, provide a global perspective in the conclusion based on the provided content." However, this additional instruction felt redundant, especially when the prompt was sufficiently specific. As a result, this part was removed for subsequent queries.

retrieved documents, including the title, caption, and chart, to fully understand the context of the query.

The structured payload for this method includes a combination of the user query, document title, caption, and chart encoded in base64 format. As with the *Llama 3.1* method, the evaluation metric dictates how many documents are included in the process, with the top one retrieved document used for NDCG@1 and increasing numbers of documents used for higher NDCG metrics. However, *GPT-4o* is distinct in its ability to process both the textual and visual aspects of the data, enabling a more thorough exploration of the documents' content. Samples of the structured payloads for NDCG@1 and NDCG@2 are presented in Figures 3.4 and 3.5, respectively.


```
{
    "model": "gpt-4o",
    "messages": [
        {
            "role": "user",
            "content": [
                {
                    "text": "You are an expert statistical analyst. Answer the
                        ↪  given query with a detailed and comprehensive
                        ↪ statistical insight from the following title,
                        ↪ content, and provided image data. Query: {query}
                          Title: {title}
                          Content: {caption}
                          Format the response in the following structure
                              ↪ with 3 paragraphs, without paragraph title:
                          1. Start the response with a clear classification
                              ↪ or straightforward answer to the query.
                          2. Provide supporting findings and detailed
                              ↪ analysis, including relevant statistical
                              ↪ data.
                          3. Summarize the final conclusion briefly.",
                    "type": "text"
                },
                {
                    "image_url": {
                        "url":"data:image/jpeg;base64,{image_data}"
                    },
                    "type": "image_url"
                }
            ],
        }
    ],
    "max_tokens": 1000
}
```


Figure 3.4: Payload sample for GPT-4o at NDCG@1.

In this approach, the user role includes both the instructions for the model and the data it needs to process. This data consists of the query, text-based content (such as titles and captions), and base64-encoded charts. The absence of a separate system role means that both the instructions on how to respond

```
{
    "model": "gpt-4o",
    "messages": [
        {
            "role": "user",
            "content": [
                {
                    "text": "You are an expert statistical analyst. Answer the
                        ↪ given query with a detailed and comprehensive
                        ↪ statistical insight from the following title,
                        ↪ content, and provided image data.
                        Query: {query}
                        Title: {title}
                        Content: {caption}
                        Title1: {title1}
                        Content2: {caption2}
                        Format the response in the following structure
                            ↪ with 3 paragraphs, without paragraph title:
                        1. Start the response with a clear classification
                            ↪ or a straightforward answer to the query.
                        2. Provide supporting findings and detailed
                            ↪ analysis, including relevant statistical
                            ↪ data.
                        3. Summarize the final conclusion briefly.",
                    "type": "text"
                },
                {
                    "image_url": {
                        "url": "data:image/jpeg;base64,{image_data}"
                    },
                    "type": "image_url"
                },
                {
                    "image_url": {
                        "url": "data:image/jpeg;base64,{image_data}"
                    },
                    "type": "image_url"
                }
            ],
        }
    ],
    "max_tokens": 1000
}
```

Figure 3.5: Payload sample for GPT-4o at NDCG@2.

and the actual query with the document data are embedded within the same user message. The prompt provided to *GPT-4o* also designates the model as an expert statistical analyst and instructs it to generate a response following a similar three-paragraph structure, as done in the first method.

Both methods are structured to provide relevant and insightful responses. *Llama 3.1* is tailored for processing text-only data, with the system role guiding the model and the user providing the data. In contrast, *GPT-4o* integrates both text and visual data within a single user role, making it more suitable for complex queries requiring visual context. The strengths of each model allow for effective analysis based on the specific nature of the input data. Additionally, these responses are further evaluated manually, a process that will be discussed in more detail in the subsequent chapter.

3.6 Dataset

The dataset used in this research are sourced from the paper *Chart-to-Text: A Large-Scale Benchmark for Chart Summarization* by [Kantharaj et al., 2022]. The dataset was derived from two prominent sources: Statista[14] and Pew Research[15]. Statista is an online platform widely recognized for publishing charts on a diverse array of topics, including economics, market research, and public opinion. Pew Research, on the other hand, is a reputable organization that focuses on social issues, public opinion, and demographic trends.

The combined dataset (Statista and Pew) comprises 29,354 charts with their descriptions, covering a broad range of topics and chart types. The construction of the dataset involved scraping publicly accessible charts and their accompanying descriptions from the Statista and Pew websites. Each chart in the dataset includes a screenshot of the chart image, the underlying data table(when available), the chart title, axis labels, and human-written captions extracted from the surrounding content on the respective web pages. Table 3.2 provides an example of the dataset format, illustrating the titles, captions, and charts for two sample entries.

The combination of varied chart types, diverse topics, and detailed captions ensures that this dataset is comprehensive and well-suited to the goals of this thesis.

Chart Type Distribution

The following Table 3.3 shows the distribution of different chart types within the Statista and Pew datasets. It is evident that bar charts dominate both sources, particularly within the Statista dataset. Line charts and pie charts are also frequently present, with a smaller number of area charts and tables.

Topic Distribution

The dataset includes charts on a variety of topics. Figures 3.6 and 3.7 present the topic distributions for the Pew and Statista Datasets, respectively. These

[14] `https://www.statista.com/`
[15] `https://www.pewresearch.org/`

distributions highlight the diversity of topics covered in the dataset, offering a wide range of domains for our research into chart-based argument retrieval.

The topic distribution was already present in the Pew Dataset. However, since the Statista Dataset lacked this information, the same list of topics from the Pew Dataset was used to classify the Statista charts. To achieve this, the *Meta-Llama-3.1-70B-Instruct* model was employed, utilizing the title and caption of each chart for classification.

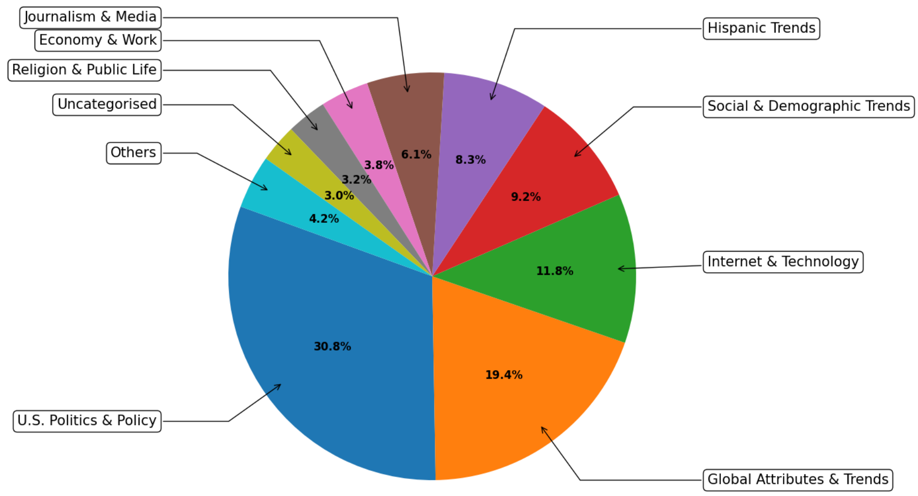


Figure 3.6: Topic Distribution in the Pew Dataset: analyzing proportional representation across domains.

Token Statistics

Table 3.4 and Figure 3.8 provide descriptive statistics for the token of the combined dataset (Statista and Pew). This information is crucial as it highlights the variability in the token size of chart titles, captions, and LLaVA-generated captions which can impact both the quality of embeddings and model efficiency by ensuring that the input token size stays within the model's token limit. The token information was obtained using a tokenizer from the longformer-base-4096 model[16], which tokenized the combined title and caption for each chart to determine the token length.

[16] `https://huggingface.co/allenai/longformer-base-4096`

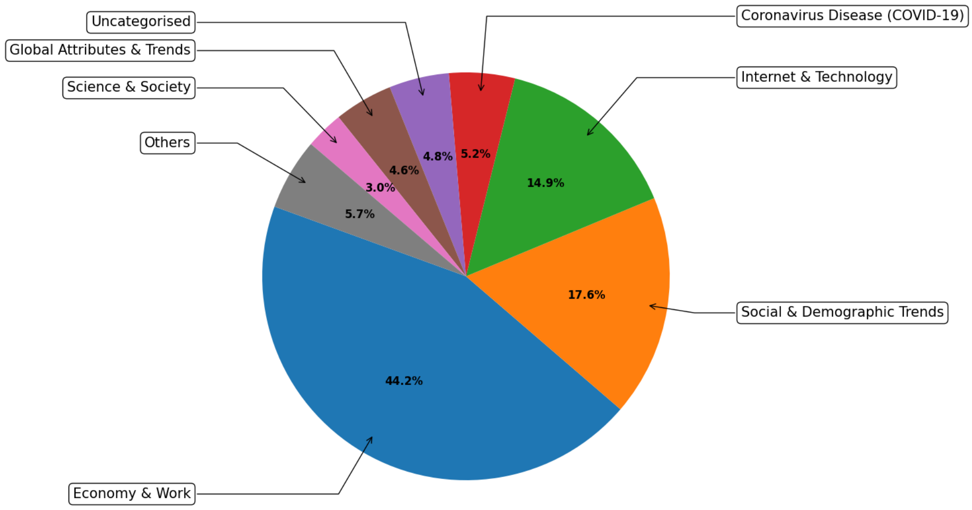


Figure 3.7: Topic Distribution in the Statista Dataset: analyzing proportional representation across domains.

LLaVA-generated Caption

LLaVA (Large Language and Vision Assistant) is a multi-modal model designed to integrate both visual and language processing capabilities. It combines the CLIP visual encoder with the Vicuna language model, enabling it to generate responses based on visual inputs. This integration allows LLaVA to interpret and describe images, including charts, while following specific language-based instructions [Liu et al., 2023].

The LLaVA-generated captions are machine-generated descriptions of chart images. The LLaVA-13B model was employed to generate these captions based on the specific prompt: *Can you provide a statistical summary and analysis of the chart?*

This prompt directed the model to provide a descriptive and analytical summary of the visual data presented in each chart. LLaVA acted as a replacement, generating captions that substituted the original captions with more descriptive summaries aligned with the statistical focus. The token statistics for the LLaVA-generated captions, reflecting the length of the descriptions, are presented in Table 3.4.

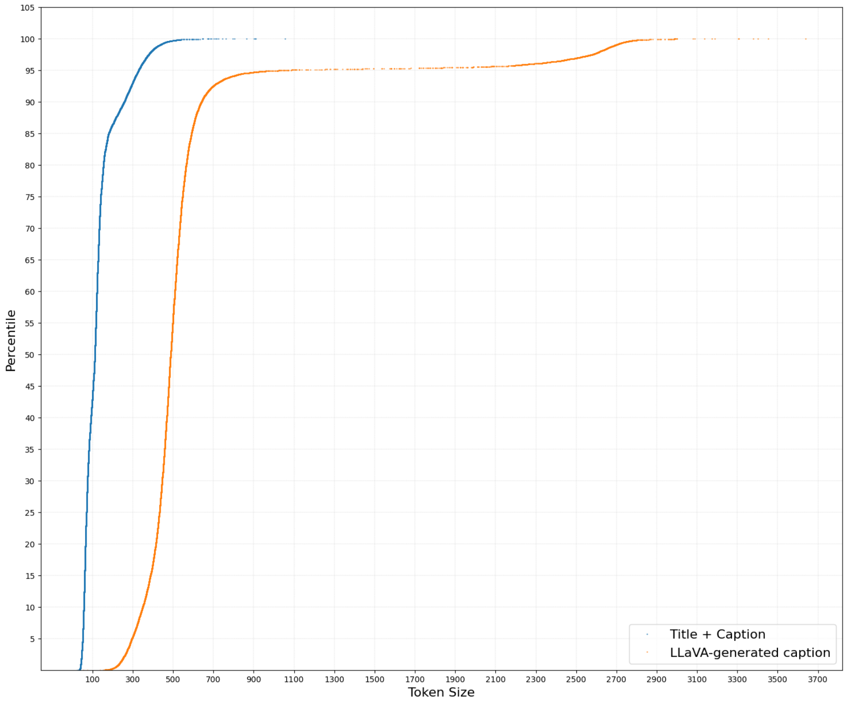


Figure 3.8: Empirical Cumulative Distribution Function of Title+Caption length (in tokens) and LLaVA-generated caption length.

3.7 Query Selection and Pre-processing

This section outlines the process of query selection and pre-processing utilized in the thesis. Three distinct types of queries were used, with each type detailed in subsections 3.7.1, 3.7.2, and 3.7.3.

3.7.1 Touché Queries

The Touché queries used in this research were sourced from the Touché dataset by the Webis Group, a well-known resource for information retrieval and argumentation mining [Bondarenko et al., 2021]. A list of 100 queries was taken from this source. The queries were converted into embeddings using the quantized *e5-mistral-7b-instruct* model and compared with the pre-indexed *Combined-text-mistral* field in the Elasticsearch engine, which contains embeddings generated by the same model. By computing the cosine similarity between the embedded queries and the indexed content, a ranking of queries based on relevance was established. The total similarity score for each query, calculated by summing the individual scores of the retrieved documents, provided a list of the top 20 queries. These top-ranked queries were selected for further analysis, ensuring a high degree of relevance for subsequent retrieval tasks.

To classify the queries into a relevant topic, we employed the *Meta-Llama-3.1-70B-Instruct* model via an API. This model processes each query and returns a classification of the query into a topic in a single word, providing insights into the topic distribution of queries across various domains. The list of manually generated queries is provided in Table 3.5.

3.7.2 Manual Queries

These queries were constructed using keywords such as *globalization*, *gas prices*, *tourism*, and *climate change* to retrieve relevant documents from the chart retrieval framework. They were then refined through an in-depth analysis of the retrieved documents to enhance the overall relevance of the retrieval process. Like the Touché queries, the manual queries were also classified into topics using the Meta-Llama-3.1-70B-Instruct model, ensuring a consistent approach to topic classification across all queries. The list of manually generated queries is provided in Table 3.6.

3.7.3 Random Queries

The queries were generated using the Meta-Llama-3.1-70B-Instruct model, which processes the combined text (titles and captions) of the dataset. The model was given a structured prompt (see listing 3.9) to produce queries like those in Touché, focusing on clarity and engagement. The length of each response was kept short to ensure the queries were concise. After generating the queries, the model classified them into different topic categories, using a one-word label for each. A total of 12,061 queries were generated, with each one based on a unique combination of text from the dataset.

For chart retrieval, 20 queries were randomly selected at a time, providing users with a diverse set of questions to choose from. This random selection prevented bias and allowed users to explore different topics by picking one query from each batch over multiple rounds.The list of manually generated queries is provided in Table 3.7.

```
{
    "model": "meta-llama/Meta-Llama-3.1-70B-Instruct",
    "messages": [
        {
            "role": "system",
            "content": (
                "You are an AI assistant. Generate a 1 line query based on the
                    ↪  following information."
                "Here are some example queries to illustrate the desired query
                    ↪  format:"
                "Are gas prices too high?"
                "Is a college education worth it?"
                "Is capitalism the best form of economy?"
                "Do we need cash?"
                "Should education be free?"
                "Does lowering the federal corporate income tax rate create
                    ↪ jobs?"
                "Should everyone get a universal basic income?"
                "Should the press be subsidized?"
                "Does legal prostitution increase human trafficking?"
                "Should the penny stay in circulation?"
                "Should government spending be reduced?"
                "Should blood donations be financially compensated?"
                "Should prescription drugs be advertised directly to consumers
                    ↪ ?"
                "Does poverty cause crime?"
            )
        },
        {
            "role": "user",
            "content": content
        }
    ],
    "max_tokens": 20,
    "temperature": 0.5
}
```

Figure 3.9: Prompt for generating Touché like queries.

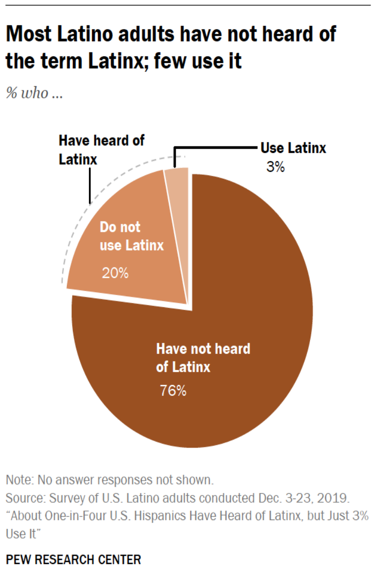
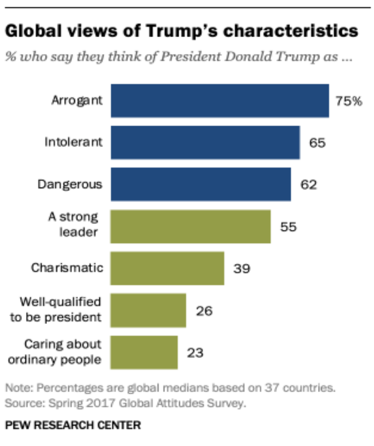
<table>
<tr><th>Title</th><th>Caption</th><th>Chart</th></tr>
<tr><td>Most Latino adults have not heard of the term Latinx; few use it</td><td>Only 23% of U.S. adults who self-identify as Hispanic or Latino have heard of the term Latinx, and just 3% say they use it to describe themselves, according to a bilingual survey of U.S. Hispanic adults the Center conducted in December 2019. Awareness and use vary across subgroups, with young Hispanics ages 18 to 29 among the most likely to have heard of the term – 42% say they have heard of it, compared with 7% of those ages 65 or older. Use is among the highest for Hispanic women ages 18 to 29 – 14% say they use it, compared with 1% of Hispanic men in the same age group who say they use it.</td><td>
</td></tr>
<tr><td>Global views of Trump 's characteristics % who say they think of President Donald Trump as Arrogant</td><td>What have people around the world not liked about Trump? Our 37-nation survey in 2017 found that many did not like his personal characteristics or leadership style. Majorities said he was arrogant, intolerant and dangerous. Few considered him well-qualified or believed that he cares about ordinary people.</td><td>
</td></tr>
</table>

Table 3.2: Sample Representation of Titles, Captions, and Charts from the Statista and Pew Research Dataset.

Chart type	Statista	Pew
Bar	24591	807
Line	2646	325
Area	0	29
Pie	408	325
Table	223	0
Total	27868	1486

Table 3.3: Chart Type Distribution.

Token Length	Title	Caption	LLaVA-generated Caption
Max	139	990	3639
Min	3	15	54
Mean	42.60	88.93	580.61
Median	25	57	488

Table 3.4: Token Statistics for the combined dataset (Pew and Statista).

Queries	Topic
Are gas prices too high?	Economy
Should the government allow illegal immigrants to become citizens?	Politics
Are social media platforms doing enough to prevent cyberbullying?	Technology
Is a college education worth it?	Education
Should abortion be legal?	Abortion
Are social networking sites good for our society?	Technology
Should recreational marijuana be legal?	Politics
Does poverty cause crime?	Sociology
Should gay marriage be legal?	Politics
Should government spending be reduced?	Economy
Should social networks be banned?	Technology
Should education be free?	Education
Should holders of public offices resign on bad approval ratings?	Politics
Should the federal minimum wage be increased?	Economy
Should Turkey join the EU?	Politics
Should any vaccines be required for children?	Healthcare
Should we imprison fewer people?	Justice
Can alternative energy effectively replace fossil fuels?	Energy
Do we need sex education in schools?	Education
Is feminism still needed?	Sociology

Table 3.5: Touché Queries with their corresponding topics.

Queries	Topic
Did COVID-19 affect global gasoline prices?	Gas Price
Do international conflicts impact fuel prices?	Gas Price
Do Americans support spending on policing?	Economy
What is the government expenditure in the US?	Economy
Is globalization beneficial for economic growth?	Economy
Should the US reform its gun laws?	Policy
Has the internet influenced US election campaigns?	Politics
Is a low inflation rate beneficial to the economy?	Economy
Is Twitter usage growing globally?	Technology
Is Facebook still the most popular social network?	Technology
Are property prices in European cities too high?	Real Estate
Are household electricity prices in Europe too high?	Energy
Is an aging population a concern for Europe's future?	Demographics
Do people prefer online shopping over in-store shopping?	E-commerce
Is China the world's leading economic power?	Economy
Does tourism significantly impact Europe's GDP?	Tourism
Is Islamophobia on the rise in the world?	Religion
Do Hispanics face discrimination in the US?	Racism
Is climate change a major global threat?	Environment
Can online banking replace traditional banking?	Finance

Table 3.6: Manual Queries with their corresponding topics.

Queries	**Topic**
Should online piracy of TV shows be punishable by law?	Piracy
Does India see Pakistan as a threat?	Politics
Should pets be considered a necessary part of a household?	Pets
Should immigrants who commit crimes be deported?	Politics
Should social media be more transparent about data collection?	Technology
Is the decline in marriage rates a cause for concern?	Sociology
Should the internet be considered a daily necessity?	Technology
Is nuclear energy safer than coal energy?	Energy
Are Football League ticket prices too expensive?	Sports
Are women more likely to use Instagram than men?	Social
Is the UK's defense spending increasing?	Defence
Should health and fitness apps be free to download?	Health
Is air travel becoming less safe?	Aviation
Do tattoos negatively impact job opportunities?	Sociology
Does reducing police funding increase crime?	Crime
Has organ donation been increasing over time in the UK?	Health
Is society more accepting of transgender people?	Sociology
Do Russians regret the breakup of the Soviet Union?	Politics
Should newspapers shift entirely to digital formats?	Media
Is anti-Semitism on the rise globally?	Politics

Table 3.7: Ramdom Queries with their corresponding topics.

Chapter 4

Evaluation

This chapter outlines a comprehensive assessment of both the retrieval and generation processes employed in the system. Section 4.1 discusses the evaluation of retrieval methods using human evaluations and the Normalized Discounted Cumulative Gain (NDCG) metric, focusing on the *relevance* and *completeness* of retrieved results. This section also includes the corresponding results and analysis of the retrieval methods' performance. Section 4.2 covers the evaluation generation process, incorporating human evaluations of the *relevance* and *faithfulness* of LLM-generated answers, and provides the associated results and analysis for the generation model's performance.

A total of 13 users participated in evaluating both retrieval and generation processes for 60 queries. Each user annotated an average of 5 queries, with each query taking approximately 1.3 hours to evaluate, leading to an estimated 80 hours to complete the human evaluation process.

4.1 Evaluation of Retrieval Process

This section provides a detailed evaluation of the retrieval process used in the system, covering both human evaluation and quantitative assessment using the NDCG metric. These evaluations focused on assessing the relevance and completeness of the documents retrieved in response to each query. The aim was to assess the retrieval's performance without revealing the retrieval methods to the users, ensuring unbiased scoring.

4.1.1 Human Evaluation of Retrieval Results

Human evaluation was employed in this thesis to ensure a detailed assessment of the retrieval results. While algorithmic evaluations or re-ranking approaches may efficiently rank documents based on predefined metrics, they often overlook complex, subjective factors such as context, relevance, and user personalization [Aliannejadi et al., 2024]. Due to the statistical details of these tasks, human evaluation was chosen over automated techniques, as it provides a deeper, more context-aware assessment of the retrieved results in relation to specific queries.

In this thesis, the results for each query, retrieved through various methods, were presented in a front-end interface without disclosing the names of the retrieval methods (see Appendix A for details). Evaluators were instructed to carefully examine each result, which comprised the document title, caption, and chart, and to assess its quality in relation to the query based on two predefined metrics: *relevance* and *completeness*. These two dimensions were critical to understanding how well the retrieved results matched the information needs of the query. Clear definitions of both metrics were provided to the evaluators, with specific guidance on how to score each document on a graded scale. This interface ensured consistency in the evaluation process, offering evaluators detailed descriptions of what constituted high or low scores for both relevance and completeness.

Relevance refers to how well the response (i.e., retrieved result) addressed the query. Scores were assigned on a scale from 0 to 3.

- **0 - Not Relevant**: The response seems to be completely random to the query.
- **1 - Partially Relevant**: The response is partially off-topic; may be vaguely related, but too divergent from the query.
- **2 - Relevant**: Response answers the query, though it might lack full detail or depth.
- **3 - Highly Relevant**: The response fully and clearly answers the query with detailed information.

Completeness evaluates whether the response provides a thorough and comprehensive answer to the query. Scores were similarly assigned on a scale from 0 to 3.

- **0 - No**: The response does not address the query or is completely unrelated.
- **1 - Somewhat**: The response addresses the query but misses significant details or only covers part of the topic.
- **2 - Mostly**: The response covers most aspects of the query but may miss minor details.

- **3 - Yes**: The response fully and thoroughly addresses the query, leaving no aspect untouched.

The sum of the relevance and completeness scores constituted the projected score for each document. This combined score provided a holistic evaluation of the quality of each retrieved result, ensuring that both the relevance and the depth of the response were accounted for. The motivation behind the dual evaluation of relevance and completeness was drawn from the principles outlined in the study by Aliannejadi et al. [2024], which provided key insights into evaluating retrieval results in complex, personalized contexts. This approach ensured that the retrieval results were not solely measured by algorithmic precision but were aligned with actual user needs and expectations through human interpretation.

4.1.2 NDCG Metric for Measuring Effectiveness

The Normalized Discounted Cumulative Gain (NDCG) is a widely-used evaluation metric in information retrieval systems to assess the quality of ranking algorithms. It is especially popular in contexts where search results or recommendations must be ranked in order of relevance to a user query. NDCG evaluates the ranking performance by considering the position of relevant items and applying a discount factor to lower-ranked items, reflecting the reality that users are more likely to interact with higher-ranked results. This property makes NDCG highly suitable for applications like search engines and recommendation systems [Wang et al., 2013].

NDCG is derived from the Discounted Cumulative Gain (DCG), which measures how relevant the items in a list are, while giving more weight to higher-ranked items. The further down an item appears in the ranking, the less valuable it is, and this diminishing value is captured using a logarithmic scale.

The formula for DCG at a specific position k is:

$$DCG@k = \sum_{i=1}^{k} \frac{rel_i}{\log_2(i+1)}$$

where,

- rel_i represents the relevance score of the item at position i. This score indicates how important or relevant an item is.
- The term $\log_2(i+1)$ reduces the influence of lower-ranked items, reflecting the fact that items further down the list contribute less to the overall value.

NDCG is the normalized version of DCG, ensuring that the score falls between 0 and 1. To normalize, DCG is divided by the Ideal DCG (IDCG), which is the maximum possible DCG that could be achieved if all items were ranked perfectly, meaning the most relevant items appear at the top of the list.

Thus, NDCG is defined as:

$$NDCG@k = \frac{DCG@k}{IDCG@k}$$

- A value of 1 indicates a perfect ranking where items are arranged in the best possible order.
- A value close to 0 indicates a poorly ranked list.

4.1.3 Results and Analysis

The performance of different retrieval methods is analyzed across various query sets, using NDCG at different k values. Table 4.1 presents the frequency of best-performing retrieval methods across NDCG@k levels, while table 4.2 provides the mean NDCG values for each method at different k levels.

The results highlight a distinct difference between sparse and dense retrieval methods. Sparse retrieval, represented by BM25, consistently under-performed across all NDCG@k levels compared to dense retrieval methods represented by *E5-Mistral-7B-Instruct*, *gte-Qwen2-7B-instruct*, and *CLIP*. However, BM25 (Title+Caption) showed competitive performance when compared to the e5-Mistral (Title+LLaVA Caption) method, indicating that, in certain scenarios, sparse methods can be comparable to dense methods. This under-performance of sparse retrieval may be attributed to the query selection process employed in this thesis, wherein queries were embedded and ranked using the *E5-Mistral-7B-Instruct* model, inherently favoring dense retrieval methods.

GTE-Qwen2 (Title+Caption) emerged with the highest frequency as the best retrieval method across all query sets, particularly at NDCG@2 and NDCG@3. GTE-Qwen2 (Title+Caption) method consistently improved with higher NDCG@k values, excelling in larger-scale retrieval tasks. For instance, in the Touché dataset, its frequency increased significantly, from 7.28 at NDCG@1 to 13.00 at NDCG@3, highlighting its ability to retrieve relevant documents in larger sets. e5-Mistral (Title+Caption) method also delivered good performance but was consistently outperformed by the GTE-Qwen2 based method.

On the other hand, CLIP (Image Embedding) also showed competitive performance when compared to BM25-based methods and e5-Mistral based methods at NDCG@1 across query sets. However, its performance diminished

Retrieval Method	**NDCG@1**	**NDCG@2**	**NDCG@3**
Touché			
BM25 (Title+Caption)	2.53	3.00	1.50
BM25 (Title+LLaVA Caption)	0.70	0.67	0.00
e5-Mistral (Title+Caption)	4.78	1.92	3.50
e5-Mistral (Title+LLaVA Caption)	2.78	2.42	2.00
GTE-Qwen2 (Title+Caption)	**7.28**	**11.25**	**13.00**
CLIP (Image Embedding)	1.92	0.75	0.00
Manual			
BM25 (Title+Caption)	1.87	3.53	3.25
BM25 (Title+LLaVA Caption)	1.03	0.53	0.00
e5-Mistral (Title+Caption)	2.37	4.70	5.75
e5-Mistral (Title+LLaVA Caption)	3.53	1.70	0.25
GTE-Qwen2 (Title+Caption)	**8.53**	**8.87**	**10.75**
CLIP (Image Embedding)	2.67	0.67	0.00
Random			
BM25 (Title+Caption)	1.70	1.08	0.00
BM25 (Title+LLaVA Caption)	1.67	1.00	0.00
e5-Mistral (Title+Caption)	**5.95**	4.08	4.50
e5-Mistral (Title+LLaVA Caption)	2.78	1.75	1.00
GTE-Qwen2 (Title+Caption)	5.62	**10.08**	**12.50**
CLIP (Image Embedding)	2.28	2.00	2.00
Total	**60**	**60**	**60**

Table 4.1: Number of times each retrieval method is the best according to NDCG@k for all 60 queries. In case of a tie, the count is split evenly so that each column sums up to a total of 60.

at NDCG@2 and NDCG@3, possibly due to the limitations of the image data (chart) utilized, which lack the depth of information and contextual detail found in the textual data employed.

Impact of LLaVA-generated Captions

The replacement of LLaVA-generated captions did not significantly improve retrieval performance for either sparse or dense retrieval methods. This was especially evident in the Touché dataset, where both BM25 (Title+LLaVA Caption) and e5-Mistral (Title+LLaVA Caption) showed considerable drops in frequency and mean NDCG scores compared to their performance using standard captions. Similar trends were observed in the Manual and Random datasets, indicating that the replaced LLaVA captions introduced complexity

Retrieval Method	**NDCG@1**	**NDCG@2**	**NDCG@3**
Touché			
BM25 (Title+Caption)	0.48	0.43	0.43
BM25 (Title+LLaVA Caption)	0.18	0.19	0.21
e5-Mistral (Title+Caption)	0.67	0.62	0.66
e5-Mistral (Title+LLaVA Caption)	0.50	0.47	0.48
GTE-Qwen2 (Title+Caption)	**0.77**	**0.78**	**0.81**
CLIP (Image Embedding)	0.37	0.36	0.37
Manual			
BM25 (Title+Caption)	0.51	0.54	0.52
BM25 (Title+LLaVA Caption)	0.30	0.29	0.26
e5-Mistral (Title+Caption)	0.67	0.75	0.80
e5-Mistral (Title+LLaVA Caption)	0.51	0.52	0.51
GTE-Qwen2 (Title+Caption)	**0.84**	**0.86**	**0.87**
CLIP (Image Embedding)	0.36	0.36	0.36
Random			
BM25 (Title+Caption)	0.54	0.54	0.52
BM25 (Title+LLaVA Caption)	0.33	0.32	0.33
e5-Mistral (Title+Caption)	0.79	0.78	0.77
e5-Mistral (Title+LLaVA Caption)	0.60	0.65	0.60
GTE-Qwen2 (Title+Caption)	**0.86**	**0.86**	**0.87**
CLIP (Image Embedding)	0.41	0.37	0.36

Table 4.2: Mean NDCG@k of each retrieval method for k$\in [1, 3]$.

and redundancy without enhancing retrieval effectiveness for either method.

Performance Variation Across Query Sets

GTE-Qwen2 (Title+Caption) consistently performs well across all query sets, with notable variation in retrieval performance across different sets. This variation is observed across all retrieval methods, except for the CLIP (Image Embedding). The mean NDCG scores from the table 4.2 reveal that retrieval models perform best on the Random query set, followed closely by Manual, while Touché demonstrates the lowest performance. This difference can be attributed to the nature of the queries within each set. The Touché queries, sourced from a different dataset, tend to be more generic, making it more challenging for retrieval models to infer relevant documents. In contrast, the Manual and Random queries are more contextually aligned with the indexed content, leading to better retrieval results.

4.2 Evaluation of Generation Process

This section presents a detailed evaluation of the generation process in Retrieval-Augmented Generation (RAG) systems, focusing on human assessments of the generated responses. The evaluation is conducted using two primary metrics: *relevance* and *faithfulness*, to rigorously assess the alignment of generated content with the user queries and the underlying factual data. The answers generated by the language models are systematically evaluated to determine the models' effectiveness in producing accurate and contextually appropriate answers.

4.2.1 Human Evaluation of Generated Response

Human evaluation plays an important role in assessing the generation component of RAG systems due to the subjective nature of the tasks. These tasks often involve variability in defining what constitutes a correct or high-quality response, making it difficult to rely on automated metrics for evaluation [Yu et al., 2024]. While automated methods provide efficient ways to assess LLM performance across multiple dimensions, they still require validation against human judgment to ensure accuracy and reliability [Lin and Chen, 2023].

The models employed for generating responses in this thesis include *GPT-4o* and *Meta-Llama-3.1-70B-Instruct*, both of which represent the most advanced, state-of-the-art LLMs. Given their superior capabilities, the use of alternative LLMs were not considered for this evaluation. Therefore, human evaluation was conducted to assess the generated responses.

The evaluation of the LLM-generated answers in this thesis followed a methodology similar to that used for human evaluation of retrieval results. As shown in Appendix A, the answers were displayed in the front-end interface without disclosing the name of the LLM. Evaluators were instructed to carefully read and analyze the answers and assess them based on two defined metrics: *relevance* and *faithfulness*. These metrics were critical in determining how well the responses aligned with the query and were factually supported by the provided content (top retrieved document at NDCG@k).

Relevance, as in the previous section, measured how well the generated response (answer) addressed the query. Scores were assigned on a scale from 0 to 3.

- **0 - Not Relevant:** The response seems to be completely random to the query.
- **1 - Partially Relevant:** The response is partially off-topic; it may be vaguely related, but too divergent from the query.

- **2 - Relevant:** The response answers the query, though it might lack full detail or depth.
- **3 - Highly Relevant:** The response fully and clearly answers the query with detailed information.

Faithfulness was introduced to determine whether the generated response (answer) contained claims that could be directly inferred from the provided content. This was critical to ensure that the model did not generate hallucinated or fabricated content unsupported by the provided data. Scores were similarly assigned on a scale from 0 to 3.

- **0 - No:** The response cannot be inferred from the given context or is unrelated to it.
- **1 - Somewhat:** The response includes some claims that can be inferred from the given context but also has significant claims that cannot be inferred.
- **2 - Mostly:** The response is mostly inferable from the given context with minor inconsistencies.
- **3 - Yes:** All claims in the response can be directly inferred from the given context.

Tables 4.3, 4.4, and 4.5 illustrate examples of generated answers from the two employed models (*GPT-4o* and *Meta-Llama-3.1-70B-Instruct*) alongside the top retrieved document at NDCG@1 for each query from different query sets. Evaluators assessed each answer by assigning *relevance* and *faithfulness* scores on a scale from 0 to 3. In addition to the scores, the comments provided by the evaluators were added to further clarify their reasoning, identifying specific strengths or weaknesses in the answers. This approach ensures a transparent and comprehensive evaluation of the generated answers, offering detailed insight into how human judgment was applied during the assessment process.

Several observations were noted from the LLM-generated answers. In some instances, the answers provided accurate interpretations of the content, demonstrating a deep understanding of both the query and the retrieved document. Notably, there were cases where the LLMs offered moral suggestions or additional reasoning in a reasonable manner, particularly in response to questions related to education and public welfare. However, there was also an instance (see table 4.6) where the answer, although not directly inferred from the provided content, was factually correct when verified through external web searches.

One challenge encountered was that the datasets used in this thesis are primarily aligned with specific countries, whereas many queries were posed from a more general or global perspective. Consequently, the generated answers occasionally lacked a broader, global context, which reduced their overall relevance. Furthermore, many answers were found to be slightly misinterpreted or unfaithful to the original content. These findings emphasize the critical role of human evaluation in identifying such nuances and ensuring the accuracy and reliability of the generated answers.

4.2.2 Results and Analysis

The figure 4.1 and table 4.7 presents a comparative analysis of answers generated by two models, *GPT-4o* and *Llama-3.1*, evaluated across three query sets - Touché, Manual, and Random. The evaluation was conducted using two key metrics: relevance and faithfulness. These metrics were assessed at varying levels of NDCG@k, where $k \in [1, 3]$, to examine the models' ability in generating both relevant and factual answers.

Relevance Analysis

The mean relevance scores across NDCG@k levels for all query sets, as presented in Table 4.7, exhibit a consistent upward trend for both models. This indicates that as the provided content (top retrieved documents) increases from NDCG@1 to NDCG@3, the models generate answers that address the query more effectively. *GPT-4o* consistently achieves higher relevance scores across all query sets, with its highest scores observed at NDCG@3. *Llama-3.1* follows a similar trend, though its scores are slightly lower compared to *GPT-4o* at corresponding NDCG levels. For example, in the Manual query set, *Llama-3.1* reaches 2.95 at NDCG@3, just below GPT-4o's 3.00. These results suggest that while both models improve in relevance as more content is provided, *GPT-4o* consistently demonstrates superior performance compared to *Llama-3.1*.

Faithfulness Analysis

The mean faithfulness scores shown in table 4.7, exhibit a more complex pattern compared to mean relevance. While mean relevance scores consistently increase with higher NDCG@k levels, faithfulness displays variability, particularly in the Touche and Random query sets. For *GPT-4o*, mean faithfulness starts at 2.75 for NDCG@1 in the Touche query set, decreases to 2.55 at NDCG@2, and then rises slightly to 2.70 at NDCG@3. This fluctuation suggests that while the LLM-generated answers may become increasingly relevant, their alignment with factual effectiveness does not always improve linearly as the amount of content increases. *Llama-3.1* also shows fluctuations in mean faithfulness scores across the query sets, following a similar pattern to *GPT-4o*.

Figure 4.1 presents the number of fully faithful LLM-generated answers. Out of 60 queries, *GPT-4o* delivered fully faithful answers to 45 queries at NDCG@1, achieving the highest faithfulness accuracy of 75% for the system. Notably, *GPT-4o* generated more fully faithful answers compared to *Llama-3.1*, underscoring its stronger performance in delivering factually accurate answers.

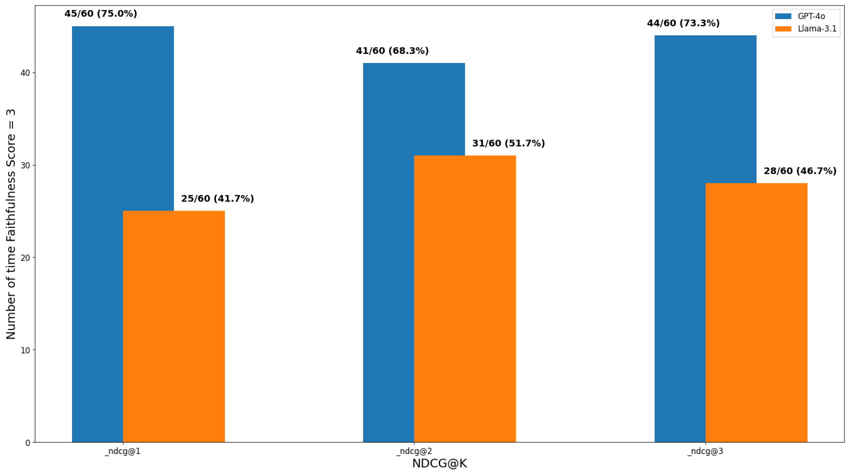


Figure 4.1: Number of times the generated answers for each query received a faithfulness score of 3 across 60 queries, evaluated across NDCG levels for $k \in [1, 3]$.

Performance Variation Across Query Sets

Both models demonstrated superior performance on the Manual and Random query sets compared to the Touché set. This result aligns with expectations, as the Manual and Random queries are more closely related to the indexed documents, enabling the retrieval of more relevant results provided to the models for answer generation. This alignment enhances the models' ability to produce answers with higher relevance and, in many cases, improved faithfulness.

In conclusion, a comparative analysis of the two models reveals that *GPT-4o* consistently outperforms *Llama-3.1* in terms of both relevance and faithfulness across most query sets and NDCG levels. It was also observed that the responses generated by *GPT-4o* contained more detailed statistical information compared to those from *Llama-3.1*. A key factor contributing to *GPT-4o's* superior performance is its multi-modal capability. Unlike the text-based

Llama-3.1, which received text-only descriptions (title and caption), *GPT-4o* was provided with both textual descriptions and chart images. This ability to incorporate visual data enhances *GPT-4o's* relevance and faithfulness, particularly in cases where charts provided critical information.

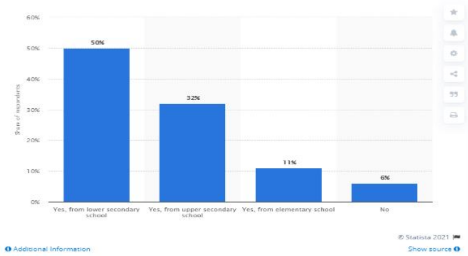
Query	**Do we need sex education in schools?**
Top retrieved document at NDCG@1	 *Title*: According to you, should the school provide information about sexuality and reproduction? *Caption*: This statistic depicts the results of a survey conducted among secondary school students asking if, in their opinion, the school should provide information about sexuality and reproduction, in Italy in 2018. According to data, the largest group of respondents (50 percent) believed that school should guarantee information on these topics starting from the lower secondary school, whereas 32 percent of students considered appropriate to provide information about sexuality and reproduction in the upper secondary school.
Gpt-4o answer	Based on the survey results, there is a clear consensus among secondary school students in Italy that sex education is needed in schools. A significant majority of students believe that information about sexuality and reproduction should be provided within the educational framework. Analyzing the data in detail, 50 percent of respondents think that sex education should start from lower secondary school. This reflects a strong inclination towards beginning this form of education earlier rather than later, highlighting a recognition of its importance during the formative years of adolescence. Additionally, 32 percent of the students believe that such information should be provided in upper secondary school, suggesting that a substantial portion of students sees value in continued sex education throughout their secondary education. Furthermore, 11 percent of respondents advocate for starting sex education as early as elementary school, emphasizing the need for early awareness. On the other hand, only 6 percent of the students believe that schools should not provide information on sexuality and reproduction, indicating minimal opposition to sex education. In conclusion, the overwhelming majority of secondary school students support the inclusion of sex education in the school curriculum at various educational stages. While preferences exist regarding the specific timing, the overall consensus is clear and strong in favor of providing this essential education to young individuals.
Eval	Relevance=2; It focuses only on Italy, missing the broader question of whether sex education is needed in schools globally. Faithfulness=2; The 32% want sex education to start in upper secondary school, not continue from earlier, so the interpretation is misleading..
Llama-3.1 answer	Based on the survey results, it is evident that the majority of secondary school students in Italy believe that sex education should be provided in schools, with a significant proportion thinking it should start from the lower secondary school level. A closer analysis of the data reveals that 50% of respondents, which is the largest group, advocate for introducing sex education in lower secondary school, suggesting that they recognize the importance of acquiring knowledge about sexuality and reproduction at an early age. This is a statistically significant proportion, indicating a strong consensus among students on the need for sex education in schools. Furthermore, the cumulative percentage of students who believe that sex education should be provided in schools, either in lower or upper secondary school, is 82% (50% + 32%), highlighting a strong demand for comprehensive sex education among students. In conclusion, the survey results provide strong evidence to support the inclusion of sex education in school curricula, with a significant majority of students believing it is essential to provide information about sexuality and reproduction in schools, starting from an early age. This finding suggests that schools have a critical role to play in providing students with accurate and comprehensive information about sex education, which can have long-term benefits for their physical, emotional, and mental well-being.
Eval	Relevance=2; Like Response 1, it focuses on Italy but misses the broader question about the global need for sex education in schools. Faithfulness=3; The interpretation is accurate, with the added long-term benefits being a reasonable conclusion.

Table 4.3: Human Evaluation of LLM-generated answers for the top document retrieved at NDCG@1 for the Touché query shown at the top.

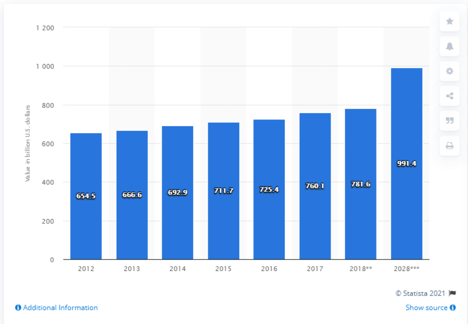
Query	**Does tourism significantly impact Europe's GDP?**
Top retrieved document at NDCG@1	 *Title*: Direct contribution of travel and tourism to GDP in Europe from 2012 to 2028*(in billion U.S. dollars). *Caption*: This statistic presents the direct contribution of travel and tourism to GDP in Europe from 2012 to 2018, with an additional forecast for 2028. Over this period, the direct contribution of the travel and tourism industry to GDP in Europe increased, reaching around 760 billion U.S. dollars in 2017.
Gpt-4o answer	Tourism does provide a significant impact on Europe's GDP. The data depicting the direct contribution of travel and tourism to Europe's GDP from 2012 to 2018, along with a forecast for 2028, demonstrates a clear and consistent increase in tourism's financial contribution, which points to its significant role in the economic framework of Europe. The direct contribution of travel and tourism to GDP in Europe has shown a steady upward trajectory over the years. Starting at 654.5 billion U.S. dollars in 2012, it rose to 781.6 billion U.S. dollars by 2018. Furthermore, the forecast for 2028 predicts this figure to surge to 991.4 billion U.S. dollars. This continuous growth highlights not only the resilience of the tourism sector but also its growing importance. The increment from 654.5 billion U.S. dollars in 2012 to 760.1 billion U.S. dollars in 2017 alone indicates a substantial growth of approximately 16.1% over five years, showcasing the dynamic expansion and the sector's intrinsic value to Europe's economy. In conclusion, the direct contribution of travel and tourism to Europe's GDP over the examined period reveals a significant impact, with a clear increasing trend that is forecasted to continue into the future. The growth data supports the assertion that tourism plays a crucial role in bolstering Europe's economic prosperity, cementing its status as a vital sector within the regional GDP structure.
Eval	Relevance=2; The answer gives good details on tourism's contribution to Europe's GDP but lacks the percentage of tourism's impact on the total GDP. Faithfulness=3; All the information provided in the answer is inferred from the given content, and the interpretation is good.
Llama-3.1 answer	Yes, tourism significantly impacts Europe's GDP. The travel and tourism industry is a substantial contributor to Europe's economy, with a notable direct contribution to GDP. A closer examination of the data reveals that the direct contribution of travel and tourism to GDP in Europe has consistently increased over the years, from approximately 540 billion U.S. dollars in 2012 to around 760 billion U.S. dollars in 2017. This represents a growth rate of about 40% over the six-year period, indicating a steady and significant impact on Europe's economy. Furthermore, the forecast for 2028 suggests that this trend is expected to continue, with the industry's direct contribution to GDP projected to reach even higher levels. The statistical significance of this impact is underscored by the substantial monetary value of the industry's contribution, which is likely to have a multiplier effect on the broader economy, generating additional economic activity and job creation. In conclusion, the data clearly indicate that tourism has a substantial and growing impact on Europe's GDP. The industry's direct contribution to GDP is significant, and its steady growth over the years underscores its importance to the European economy. As such, it is likely that tourism will continue to play a vital role in shaping Europe's economic landscape in the years to come.
Eval	Relevance=2; The answer gives good details on tourism's contribution to Europe's GDP but lacks information on its impact on the total GDP, similar to the first answer. Faithfulness=1; The answer includes incorrect data, citing 540 billion in 2012 and a 40% growth rate, neither of which are inferred from the given content.

Table 4.4: Human Evaluation of LLM-generated answers for the top document retrieved at NDCG@1 for the manual query shown at the top.

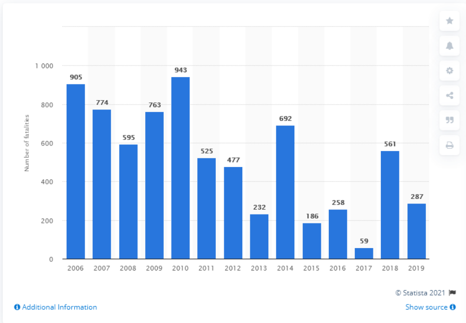
Query	**Is air travel becoming less safe?**
Top retrieved document at NDCG@1	 *Title*: Number of worldwide air traffic fatalities from 2006 to 2019. *Caption*: Air travel fatalities have been recorded in each of the last 12 years, with a total of 287 deaths in 2019 due to air crashes. However, despite some pronounced year-to-year differences, the overall trend has been for a reduction in the number of fatalities ...a trend confirmed when looking at a slightly longer or much longer time frame. Fatal airline accidents According to the Convention on International Civil Aviation, air traffic fatalities refer to an incident where a person is fatally injured due to an occurrence associated with the operation of the aircraft. This definition covers any time from when the first person starts boarding to when the final person disembarks the plane. Corporate jet and military transport accidents are generally excluded. Overall safety of air travel The overall trend downward in air travel fatalities is notable given that the volume of passenger air traffic has increased by more than 66 percent since 2004. Indeed, when considered in terms of the number of accidents per distance travelled, air travel is statistically the safest form of transport. For example, in both the United States and the United Kingdom, air travel is many thousands of times safer than the most dangerous form of travel ...motorcycle riding.
Gpt-4o answer	Based on the provided data and analysis, air travel is becoming increasingly safer over time. Despite fluctuations in the number of air traffic fatalities from year to year, the overall trend from 2006 to 2019 indicates a general decline in fatal incidents, which suggests improvements in air travel safety protocols and technologies. Supporting this conclusion, the number of fatalities in 2019 was 287, a significant decline from the peak of 943 fatalities in 2010. The chart displays annual variations, with some years experiencing higher fatalities (e.g., 692 in 2014 and 561 in 2018), and others showing remarkable reductions (e.g., just 59 in 2017). This fluctuating trend underscores the unpredictable nature of air accidents but does not negate the overall decreasing trend in fatalities. Furthermore, relative to the considerable increase in air traffic volume—over 66% since 2004—this decline in fatalities per distance traveled highlights significant advancements in aviation safety. In conclusion, while individual years may show peaks in fatalities, the long-term trajectory illustrates that air travel is becoming safer. Despite the inherent risks associated with flying, the continuous reduction in fatalities, especially when considering increased passenger volumes, reaffirms air travel as a statistically safer mode of transportation.
Eval	Relevance=3; Clearly addresses the query with details. Faithfulness=3; All the claims made in the answer can be inferred from the provided content with good and reasonable interpretation.
Llama-3.1 answer	Air travel is becoming safer, despite some year-to-year fluctuations in the number of fatalities. The data suggests that the overall trend is a reduction in the number of air traffic fatalities over the past 12 years, with a total of 287 deaths in 2019 due to air crashes. A closer examination of the data reveals that the decline in air travel fatalities is even more impressive when considered in the context of the significant increase in passenger air traffic over the same period. Since 2004, the volume of passenger air traffic has increased by more than 66 percent, yet the number of fatalities has not followed a similar upward trend. In fact, when measured in terms of the number of accidents per distance travelled, air travel is statistically the safest form of transport. This is evident in the comparison with other modes of transportation, such as motorcycle riding, which is many thousands of times more dangerous than air travel in both the United States and the United Kingdom. For example, if we consider the number of fatalities per mile travelled, air travel has a fatality rate of approximately 0.01 per 100 million passengers, compared to 212.57 per 100 million passengers for motorcycle riding in the United States. In conclusion, the data suggests that air travel is becoming safer over time, despite some minor fluctuations in the number of fatalities. The significant increase in passenger air traffic over the past decade and a half has not led to a corresponding increase in fatalities, and air travel remains the safest form of transportation when measured in terms of the number of accidents per distance travelled.
Eval	Relevance=3; This answer directly addresses the query of air travel safety and includes relevant details. Faithfulness=2; Most claims in the answer can be inferred from the provided content, but the specific fatality rate comparison (e.g., 0.01 for air travel vs. 212.57 for motorcycles) isn't mentioned in the given content.

Table 4.5: Human Evaluation of LLM-generated answers for the top document retrieved at NDCG@1 for the random query shown at the top.

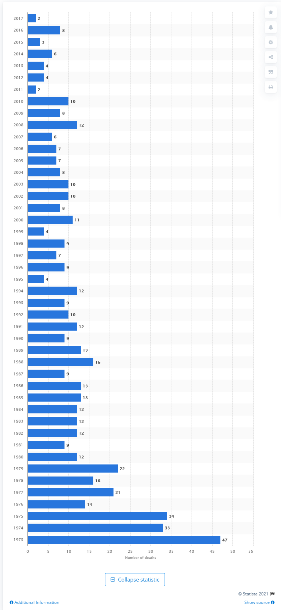
Query	**Should abortion be legal?**
Top retrieved document at NDCG@1	 *Title*: Number of abortion-related deaths reported in the U.S. from 1973 to 2017. *Caption*: The number of abortion-related deaths in the U.S. has decreased dramatically since 1973. In 1973, the number of deaths related to abortions was 47. As of 2017, the number of reported deaths related to abortions had decreased to just 2. Abortion is the act of ending a pregnancy so that it does not result in the birth of a baby. Abortions in the U.S. Abortions can be performed in a surgical setting or a medical setting (the pill). The number of legal abortions reported in the U.S. has been on the decline in recent years. The most frequently performed kind of abortion in the U.S. in 2016 were surgical abortions, which includes procedures such as curettage or vacuum aspiration. Abortion and the legality and morality of the procedure has been a publicly debated topic in recent years. Public opinions on abortion In the first half of 2019 at least 8 states had passed laws to limit or ban abortions. In general, most U.S. women believe that legislation that protects access to abortions (Roe vs. Wade) should not be overturned. Opinions on abortions can be divided into two campaigns. Pro-choice is the belief that women have the right to decide when they want to become pregnant and if they want to terminate the pregnancy through an abortion. Pro-life, is the belief that women should not be able to choose to have an abortion in any situation. The percentage of the U.S. population that is pro-choice has remained stable over the years. However, a recent survey has indicated that more people in the U.S. thought abortions were morally wrong as opposed to morally acceptable.
Llama-3.1 answer	Based on the provided data and trends, it can be inferred that making abortion legal leads to a significant reduction in abortion-related deaths. The dramatic decrease in abortion-related deaths in the U.S. from 47 in 1973 to 2 in 2017 suggests that the legalization of abortion has improved the safety of the procedure. A closer examination of the data reveals that the number of abortion-related deaths has consistently declined over the years, indicating a positive correlation between the legalization of abortion and the reduction in mortality rates. Additionally, the fact that the number of surgical abortions, which is the most frequently performed kind of abortion, has been on the decline in recent years, suggests that the availability of safe and legal abortion services has reduced the need for more invasive and potentially life-threatening procedures. Furthermore, public opinion polls indicate that a majority of women in the U.S. believe that legislation protecting access to abortions should not be overturned, suggesting that there is a significant demand for safe and legal abortion services. In conclusion, from a global perspective, the data from the U.S. suggests that making abortion legal can lead to a significant reduction in abortion-related deaths. The World Health Organization (WHO) estimates that approximately 25 million unsafe abortions take place globally each year , resulting in 7 million hospitalizations and 22,800 deaths. Therefore, it is essential to consider the legalization of abortion as a public health measure to reduce maternal mortality rates and ensure that women have access to safe and legal reproductive health services.
Eval	Relevance=3; The answer addresses the query by including U.S. data and global statistics, which enhances the argument and offers a broader perspective. Faithfulness=2; The reference to global data from WHO is not supported by the provided content. However, it was found to be accurate through a web search (see https://www.who.int/news/item/28-09-2017-worldwide-an-estimated-25-million-unsafe-abortions-occur-each-year, https://www.who.int/news-room/fact-sheets/detail/abortion).

Table 4.6: Example of an LLM-generated answer, factually correct but not directly inferred from the provided content, verified via external web searches.

Model	Metric	NDCG@1	NDCG@2	NDCG@3
Touche				
GPT-4o	Relevance	2.75	2.80	2.85
	Faithfulness	2.75	2.55	2.70
Llama-3.1	Relevance	2.70	2.70	2.75
	Faithfulness	2.40	2.35	2.60
Manual				
GPT-4o	Relevance	2.80	2.90	3.00
	Faithfulness	2.60	2.75	2.80
Llama-3.1	Relevance	2.75	2.85	2.95
	Faithfulness	2.35	2.50	2.25
Random				
GPT-4o	Relevance	2.80	3.00	2.90
	Faithfulness	2.65	2.50	2.70
Llama-3.1	Relevance	2.80	3.00	3.00
	Faithfulness	2.05	2.40	2.30

Table 4.7: Mean Relevance and Faithfulness of the responses generated by each model across NDCG levels for $k \in [1, 3]$.

Chapter 5

Conclusion

This thesis has addressed the challenge of retrieving and interpreting chart data within the retrieval and generation system. By proposing and evaluating a RAG framework specifically designed for the retrieval of charts and their textual descriptions, we have advanced the capabilities of retrieval and generation systems in handling complex multi-modal data essential for industries that rely on statistics. For example, an e-commerce marketing manager might query, *What were the top-selling products during Black Friday last year?* to access critical statistical insights for strategic decision-making. The system would efficiently retrieve relevant sales charts and generate an answer, enabling the manager to quickly interpret the data and make informed decisions.

This thesis was guided by two primary questions: *How effective are the retrieval methods in addressing a given query?* and *How reliable are the generated answers for a given query?*. The evaluation involved both human judgment and quantitative metric, providing a robust assessment of the processes.

Key findings

1. **Superior performance of dense retrieval methods:** Dense retrieval models, such as *GTE-Qwen2-7B-instruct*, significantly outperform sparse approaches by effectively capturing semantic similarities, leading to higher NDCG scores.

2. **Advantages of multi-modal generative models:** Models capable of processing both text and images, like *GPT-4o*, generate more relevant and faithful answers by integrating visual context from charts, surpassing text-only models.

3. **Enhanced relevance with more provided Content:** The relevance of the generated answers improved when more content (i.e., top retrieved

documents) was provided to the model.

4. **Variation across query sets:** Retrieval and generation perform better on manually constructed or document-inferred queries compared to the Touché queries. Aligning datasets and queries within the same domain is crucial for optimizing system capabilities, leading to enhanced retrieval accuracy and more faithful generated responses. This finding has significant industry-specific applications, as such alignment can substantially improve system performance across various sectors.

The evaluation of these findings involved both human judgment and quantitative metrics, providing a robust assessment of the processes. The findings highlight the potential of integrating advanced dense retrieval techniques and multi-modal generative models to enhance the retrieval and generation systems handling charts. Such integration can help in decision-making processes in data-driven industries. This thesis contributes valuable insights into the development of a RAG-based system capable of managing complex multi-modal content.

Limitations

Despite the promising results demonstrated by the retrieval and generative models, several limitations must be acknowledged. Firstly, the research is constrained by the size and diversity of the dataset used. With only 29,354 chart entries, the dataset may not adequately represent the extensive range of queries encountered in this work, potentially affecting the system's ability to retrieve relevant results and generate effective answers. Secondly, although both generative models demonstrate proficiency in generating responses, they are not exempt from occasional misinterpretations and misinformation. Consequently, the human evaluation of the generated responses proved to be time-consuming.

Future Directions

In this thesis, we used retrieval methods based separately on textual embeddings and image embeddings. For future work, integrating a multi-modal model capable of processing both long textual descriptions and charts into unified embeddings could enhance retrieval performance by capturing the combined information. Addressing misinterpretations by generative models is also crucial for reliability; therefore, future research could focus on implementing automated verification steps within the generation process or integrating self-correcting feedback mechanisms to improve model accuracy without relying on time-consuming human intervention. Additionally, implementing seamless real-time updates to the Elasticsearch database would be beneficial. As time-based statistical content changes, updates would ensure the system remains current and more useful to users.

Appendix A

Evaluation Interface

Evaluation Interface for Retrieval Results

The interface in Figure A.1 illustrates how retrieval results are presented for human evaluation. Users assess the relevance and completeness of the retrieved documents in relation to the query

Evaluation Interface for LLM-generated Answer

Figure A.2 illustrates the interface used for displaying both top retrieved documents (at NDCG@k) and LLM-generated answers, designed to facilitate human evaluation. Users evaluate the relevance of the generated answer in relation to the query and assess its faithfulness by analyzing the content of the top retrieved documents. Additionally, the chart of the document can be enlarged on click, as shown in the figure.

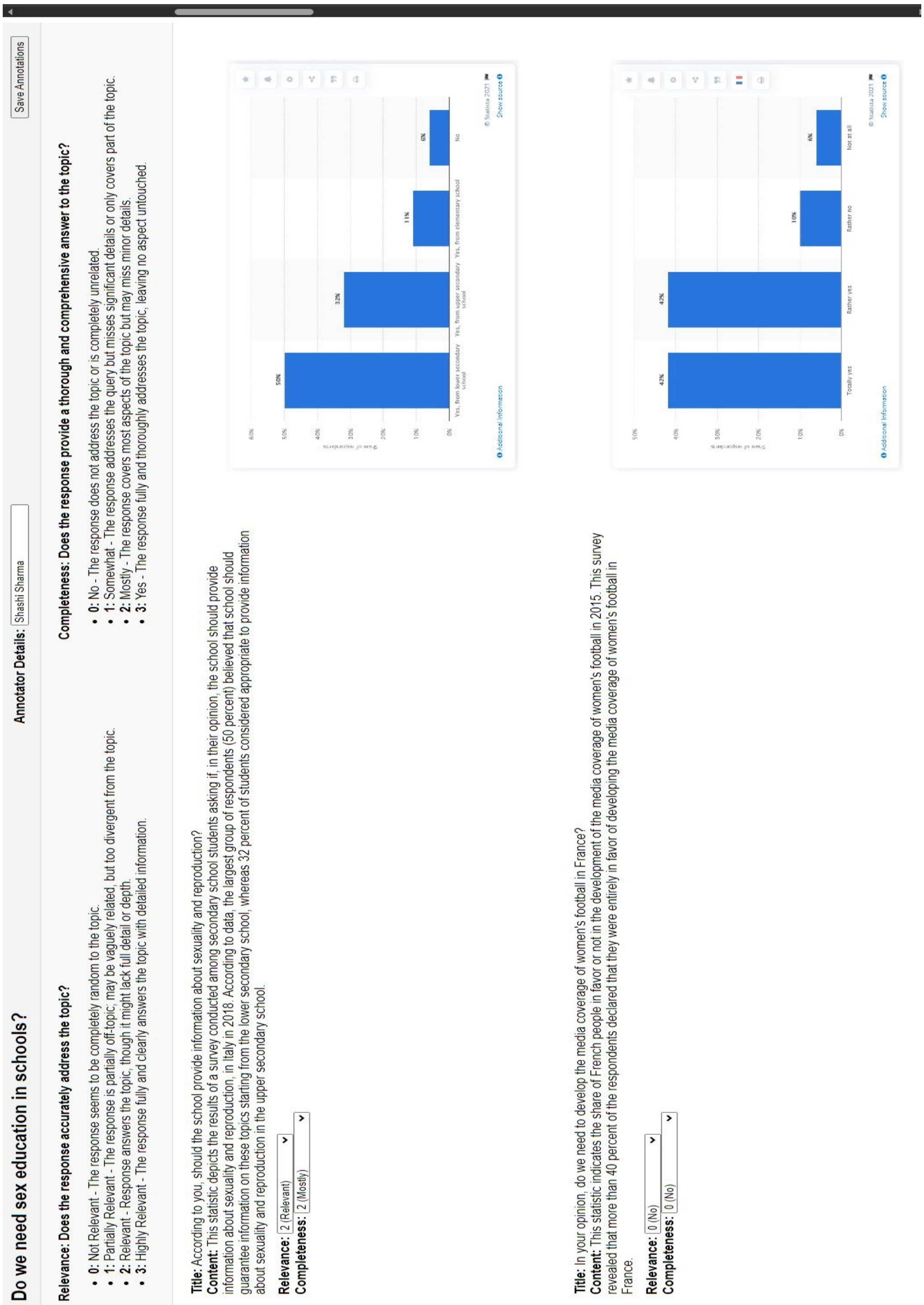


Figure A.1: An interface for displaying retrieval results and facilitating human evaluation.

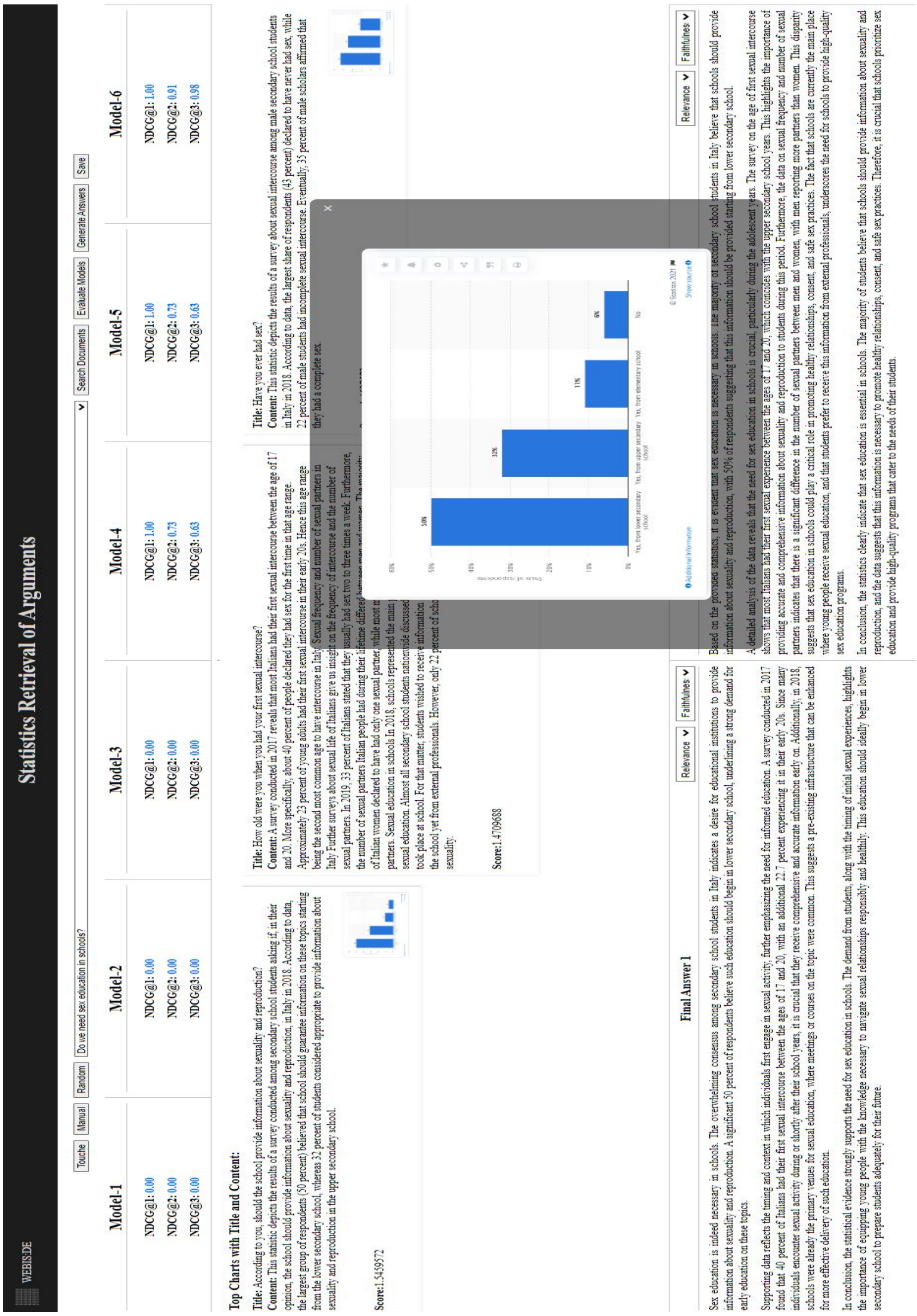


Figure A.2: An interface displaying top retrieved documents at NDCG@3 with LLM-generated answers, facilitating human evaluation.

Bibliography

Hazem Abdelazim, Mohamed Tharwat, and Ammar Mohamed. Semantic embeddings for arabic retrieval augmented generation (arag). *International Journal of Advanced Computer Science and Applications*, 14(11), 2023. doi: 10.14569/IJACSA.2023.01411135. URL http://dx.doi.org/10.14569/IJACSA.2023.01411135.

Mohammad Aliannejadi, Zahra Abbasiantaeb, Shubham Chatterjee, Jeffrey Dalton, and Leif Azzopardi. Trec ikat 2023: A test collection for evaluating conversational and interactive knowledge assistants. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2024, page 819–829. ACM, July 2024. doi: 10.1145/3626772.3657860. URL http://dx.doi.org/10.1145/3626772.3657860.

Alexander Bondarenko, Lukas Gienapp, Maik Fröbe, Meriem Beloucif, Yamen Ajjour, Alexander Panchenko, Chris Biemann, Benno Stein, Henning Wachsmuth, Martin Potthast, and Matthias Hagen. Overview of Touché 2021: Argument Retrieval. In K. Selçuk Candan, Bogdan Ionescu, Lorraine Goeuriot, Henning Müller, Alexis Joly, Maria Maistro, Florina Piroi, Guglielmo Faggioli, and Nicola Ferro, editors, *Experimental IR Meets Multilinguality, Multimodality, and Interaction. 12th International Conference of the CLEF Association (CLEF 2021)*, volume 12880 of *Lecture Notes in Computer Science*, pages 450–467, Berlin Heidelberg New York, September 2021. Springer. doi: 10.1007/978-3-030-85251-1_28.

Patrice Béchard and Orlando Marquez Ayala. Reducing hallucination in structured outputs via retrieval-augmented generation, 2024. URL https://arxiv.org/abs/2404.08189.

Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. Benchmarking large language models in retrieval-augmented generation, 2023. URL https://arxiv.org/abs/2309.01431.

Sukmin Cho, Soyeong Jeong, Jeongyeon Seo, Taeho Hwang, and Jong C. Park. Typos that broke the rag's back: Genetic attack on rag pipeline by simulating documents in the wild via low-level perturbations. *ArXiv*, abs/2404.13948, 2024. URL `https://api.semanticscholar.org/CorpusID:269293062`.

Wenjing Dai, Meng Wang, Zhibin Niu, and Jiawan Zhang. Chart decoder: Generating textual and numeric information from chart images automatically. *Journal of Visual Languages and Computing*, 48:101–109, 2018. ISSN 1045-926X. doi: https://doi.org/10.1016/j.jvlc.2018.08.005. URL `https://www.sciencedirect.com/science/article/pii/S1045926X18301162`.

Yihao Ding, Kaixuan Ren, Jiabin Huang, Siwen Luo, and Soyeon Caren Han. Pdf-mvqa: A dataset for multimodal information retrieval in pdf-based visual question answering, 2024. URL `https://arxiv.org/abs/2404.12720`.

Abhimanyu Dubey et al. The llama 3 herd of models, 2024. URL `https://arxiv.org/abs/2407.21783`.

Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. RAGAS: Automated Evaluation of Retrieval Augmented Generation. *arXiv e-prints*, art. arXiv:2309.15217, September 2023. doi: 10.48550/arXiv.2309.15217.

Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. Clip2video: Mastering video-text retrieval via image clip, 2021. URL `https://arxiv.org/abs/2106.11097`.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024. URL `https://arxiv.org/abs/2312.10997`.

Lukas Gienapp, Harrisen Scells, Niklas Deckers, Janek Bevendorff, Shuai Wang, Johannes Kiesel, Shahbaz Syed, Maik Fröbe, Guido Zuccon, Benno Stein, Matthias Hagen, and Martin Potthast. Evaluating generative ad hoc information retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, volume 3 of *SIGIR 2024*, page 1916–1929. ACM, July 2024. doi: 10.1145/3626772.3657849. URL `http://dx.doi.org/10.1145/3626772.3657849`.

Kailash A. Hambarde and Hugo Proença. Information retrieval: Recent advances and beyond. *IEEE Access*, 11:76581–76604, 2023. doi: 10.1109/ACCESS.2023.3295776.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V. Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering, 2024. URL `https://arxiv.org/abs/2402.07630`.

Jonathan Herzig, Thomas Müller, Syrine Krichene, and Julian Martin Eisenschlos. Open domain question answering over tables via dense retrieval, 2021. URL `https://arxiv.org/abs/2103.12011`.

Siqing Huo, Negar Arabzadeh, and Charles Clarke. Retrieving supporting evidence for generative question answering. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, SIGIR-AP '23. ACM, November 2023. doi: 10.1145/3624918.3625336. URL `http://dx.doi.org/10.1145/3624918.3625336`.

Wei Ji, Yinwei Wei, Zhedong Zheng, Hao Fei, and Tat-seng Chua. Deep multimodal learning for information retrieval. In *Proceedings of the 31st ACM International Conference on Multimedia*, MM '23, page 9739–9741, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 9798400701085. doi: 10.1145/3581783.3610949. URL `https://doi.org/10.1145/3581783.3610949`.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023. URL `https://arxiv.org/abs/2310.06825`.

Daekyoung Jung, Wonjae Kim, Hyunjoo Song, Jeong-in Hwang, Bongshin Lee, Bohyoung Kim, and Jinwook Seo. Chartsense: Interactive data extraction from chart images. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, page 6706–6717, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450346559. doi: 10.1145/3025453.3025957. URL `https://doi.org/10.1145/3025453.3025957`.

Shankar Kantharaj, Rixie Tiffany Ko Leong, Xiang Lin, Ahmed Masry, Megh Thakkar, Enamul Hoque, and Shafiq Joty. Chart-to-text: A large-scale benchmark for chart summarization, 2022. URL `https://arxiv.org/abs/2203.06486`.

Sujit Khanna and Shishir Subedi. Tabular embedding model (tem): Finetuning embedding models for tabular rag applications, 2024. URL `https://arxiv.org/abs/2405.01585`.

Jay Lal, Aditya Mitkari, Mahesh Bhosale, and David Doermann. Lineformer: Rethinking line chart data extraction as instance segmentation, 2023. URL `https://arxiv.org/abs/2305.01837`.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning, 2023. URL `https://arxiv.org/abs/2308.03281`.

Yen-Ting Lin and Yun-Nung Chen. Llm-eval: Unified multi-dimensional automatic evaluation for open-domain conversations with large language models, 2023. URL `https://arxiv.org/abs/2305.13711`.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023. URL `https://arxiv.org/abs/2304.08485`.

Junyu Luo, Zekun Li, Jinpeng Wang, and Chin-Yew Lin. Chartocr: Data extraction from charts images via a deep hybrid framework. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1916–1924, 2021. doi: 10.1109/WACV48630.2021.00196.

Ananth Majumdar. Elasticsearch as a nosql database for fast distributed financial service applications. *Journal of Artificial Intelligence & Cloud Computing*, 2022. URL `https://api.semanticscholar.org/CorpusID:272314913`.

Yu. A. Malkov and D. A. Yashunin. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs, 2018. URL `https://arxiv.org/abs/1603.09320`.

Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. Mteb: Massive text embedding benchmark. *arXiv preprint arXiv:2210.07316*, 2022. doi: 10.48550/ARXIV.2210.07316. URL `https://arxiv.org/abs/2210.07316`.

Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models, 2024. URL `https://arxiv.org/abs/2307.06435`.

Thong Nguyen, Mariya Hendriksen, Andrew Yates, and Maarten de Rijke. Multimodal learned sparse retrieval with probabilistic expansion control, 2024. URL https://arxiv.org/abs/2402.17535.

Chunhe Ni, Jiang Wu, Hongbo Wang, Wenran Lu, and Chenwei Zhang. Enhancing cloud-based large language model processing with elasticsearch and transformer models, 2024. URL https://arxiv.org/abs/2403.00807.

Averi Nowak, Francesco Piccinno, and Yasemin Altun. Multimodal chart retrieval: A comparison of text, table and image based approaches. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5488–5505, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.307. URL https://aclanthology.org/2024.naacl-long.307.

Boci Peng, Yun Zhu, Yongchao Liu, Xiaohe Bo, Haizhou Shi, Chuntao Hong, Yan Zhang, and Siliang Tang. Graph retrieval-augmented generation: A survey, 2024. URL https://arxiv.org/abs/2408.08921.

David Pride, Matteo Cancellieri, and Petr Knoth. Core-gpt: Combining open access research and large language models for credible, trustworthy question answering, 2023. URL https://arxiv.org/abs/2307.04683.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

D. Rafailidis, S. Manolopoulou, and P. Daras. A unified framework for multimodal retrieval. *Pattern Recognition*, 46(12):3358–3370, 2013. ISSN 0031-3203. doi: https://doi.org/10.1016/j.patcog.2013.05.023. URL https://www.sciencedirect.com/science/article/pii/S0031320313002471.

Mohaimenul Raiaan, Md. Saddam Hossain, Kaniz Fatema, Nur Fahad, Sadman Sakib, Most. Marufatul Jannat Mim, Jubaer Ahmad, Mohammed Eunus Ali, and Sami Azam. A review on large language models: Architectures, applications, taxonomies, open issues and challenges. *IEEE Access*, PP:1–1, 01 2024. doi: 10.1109/ACCESS.2024.3365742.

Jon Saad-Falcon, O. Khattab, Christopher Potts, and Matei Zaharia. Ares: An automated evaluation framework for retrieval-augmented generation systems. *ArXiv*, abs/2311.09476, 2023. URL `https://api.semanticscholar.org/CorpusID:265221210`.

Tobias Schimanski, Jingwei Ni, Mathias Kraus, Elliott Ash, and Markus Leippold. Towards faithful and robust llm specialists for evidence-based question-answering, 2024. URL `https://arxiv.org/abs/2402.08277`.

Sakib Shahriar, Brady Lund, Nishith Reddy Mannuru, Muhammad Arbab Arshad, Kadhim Hayawi, Ravi Varma Kumar Bevara, Aashrith Mannuru, and Laiba Batool. Putting gpt-4o to the sword: A comprehensive evaluation of language, vision, speech, and multimodal proficiency, 2024. URL `https://arxiv.org/abs/2407.09519`.

Mohamad-Hani Temsah, Amr Jamal, Khalid Alhasan, Fadi Aljamaan, Ibraheem Altamimi, Khalid Malki, Abdulrahman Temsah, Robin Ohannessian, and Ayman Al-Eyadhy. Transforming virtual healthcare: The potentials of chatgpt-4omni in telemedicine. *Cureus*, 16:e61377, May 2024. doi: 10.7759/cureus.61377.

Nandan Thakur, Luiz Bonifacio, Xinyu Crystina Zhang, Odunayo Ogundepo, Ehsan Kamalloo, David Alfonso-Hermelo, Xiaoguang Li, Qun Liu, Boxing Chen, Mehdi Rezagholizadeh, and Jimmy J. Lin. Nomiracl: Knowing when you don't know for robust multilingual retrieval-augmented generation. *ArXiv*, abs/2312.11361, 2023. URL `https://api.semanticscholar.org/CorpusID:266359301`.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. URL `https://arxiv.org/abs/1706.03762`.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*, 2022.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*, 2023.

Yining Wang, Liwei Wang, Yuanzhi Li, Di He, Tie-Yan Liu, and Wei Chen. A theoretical analysis of ndcg type ranking measures, 2013. URL `https://arxiv.org/abs/1304.6480`.

Siwei Wu, Yizhi Li, Kang Zhu, Ge Zhang, Yiming Liang, Kaijing Ma, Chenghao Xiao, Haoran Zhang, Bohao Yang, Wenhu Chen, Wenhao Huang, Noura Al Moubayed, Jie Fu, and Chenghua Lin. Scimmir: Benchmarking scientific multi-modal information retrieval, 2024. URL https://arxiv.org/abs/2401.13478.

Hao Yu, Aoran Gan, Kai Zhang, Shiwei Tong, Qi Liu, and Zhaofeng Liu. Evaluation of retrieval-augmented generation: A survey, 2024. URL https://arxiv.org/abs/2405.07437.

Duzhen Zhang, Yahan Yu, Jiahua Dong, Chenxing Li, Dan Su, Chenhui Chu, and Dong Yu. Mm-llms: Recent advances in multimodal large language models, 2024. URL https://arxiv.org/abs/2401.13601.

H. Zhang and M. O. Shafiq. Survey of transformers and towards ensemble learning using transformers for natural language processing. *Journal of Big Data*, 11(1):25, 2024. doi: 10.1186/s40537-023-00842-0. URL https://doi.org/10.1186/s40537-023-00842-0.

Ruochen Zhao, Hailin Chen, Weishi Wang, Fangkai Jiao, Xuan Long Do, Chengwei Qin, Bosheng Ding, Xiaobao Guo, Minzhi Li, Xingxuan Li, and Shafiq Joty. Retrieving multimodal information for augmented generation: A survey, 2023. URL https://arxiv.org/abs/2303.10868.

Xinxin Zheng, Feihu Che, Jinyang Wu, Shuai Zhang, Shuai Nie, Kang Liu, and Jianhua Tao. Ks-llm: Knowledge selection of large language models with evidence document for question answering, 2024. URL https://arxiv.org/abs/2404.15660.

Lei Zhu, Tianshi Wang, Fengling Li, Jingjing Li, Zheng Zhang, and Heng Tao Shen. Cross-modal retrieval: A systematic review of methods and future directions. *ArXiv*, abs/2308.14263, 2023. URL https://api.semanticscholar.org/CorpusID:261243752.

Tianyu Zhu, Myong Chol Jung, and Jesse Clark. Generalized Contrastive Learning for Multi-Modal Retrieval and Ranking. *arXiv e-prints*, art. arXiv:2404.08535, April 2024. doi: 10.48550/arXiv.2404.08535.