A Retrieval Based Integrated Fact Verification and Question Answering System

A Semantic Retrieval Framework for Reliable Web-Based Information Analysis

Arcot Naveen #1, Dr.D.William Albert #2, G Sreeramulu #3

#1 *M.Tech Student, Dept. of CSE, Bheema Institute of Technology and Science, Adoni, India*

#2 *Professor, Head of Dept. CSE, Bheema Institute of Technology and Science, Adoni, India*

#3 *Professor, Dept. of CSE, Bheema Institute of Technology and Science, Adoni, India*

[1]arcotnaveen234@gmail.com, [2] dr.albertdwgtl@gmail.com, [3]gudla.sri698@gmail.com

Abstract

The rapid growth of online information has increased the demand for intelligent systems capable of delivering accurate and trustworthy answers. Although Large Language Models (LLMs) generate fluent and contextually relevant responses, they are often prone to hallucination and factual inconsistencies. This creates a need for systems that not only retrieve and generate answers but also verify their correctness.

This project presents "A Retrieval-Based Integrated Fact Verification and Question Answering System", which combines Retrieval-Augmented Generation (RAG) with an automated fact verification mechanism.

Upon receiving a user query, the system retrieves the most relevant document segments and provides them as context to a generative language model for answer generation. The generated answer is accompanied by source references to ensure transparency. Additionally, an integrated fact verification module evaluates the factual claims in the response, assigns a truthfulness score, and provides a confidence level with justification.

The proposed system improves reliability by grounding responses in retrieved documents and validating them through automated evaluation. Implemented using Python, Streamlit, LangChain, FAISS, and Google Gemini APIs, the system demonstrates enhanced credibility compared to standalone generative models.

Keywords

Retrieval-Augmented Generation, Fact Verification, Question Answering, Large Language Models, FAISS, Semantic Retrieval, Natural Language Processing.

Introduction

In recent years, the advancement of Artificial Intelligence (AI) and Natural Language Processing (NLP) has led to the development of intelligent systems capable of understanding and generating human language. Large Language Models (LLMs) have significantly improved the way machines interact with users by providing conversational and context-aware responses. These systems are widely used in applications such as chatbots, virtual assistants, search engines, and educational tools.

Traditional search engines mainly rely on keyword-based retrieval methods, where users are provided with a list of documents related to their query. While this approach ensures access to relevant sources, it requires users to manually read and interpret the information. On the other hand, generative AI systems directly produce answers in natural language format, improving user experience. However, generative systems may sometimes produce incorrect or fabricated information, commonly referred to as hallucinations.

The increasing spread of misinformation and unreliable online content has highlighted the importance of trustworthy AI systems. Therefore, there is a need for systems that combine the strengths of retrieval-based methods with generative models while also verifying the factual accuracy of generated responses.

The internet contains a vast amount of information, but not all of it is reliable or factually correct. Webpages may contain outdated, misleading, or incorrect statements. While retrieval-based question answering systems allow users to extract information from specific sources, they do not assess whether the retrieved information itself is trustworthy.

Most existing question answering systems focus only on generating answers from the provided content. They assume that the source information is correct. However, in real-world scenarios, users often need to verify the authenticity of the information available on a webpage before relying on it.

Therefore, there is a need for an intelligent system that not only retrieves answers from a given URL but also evaluates the correctness of the information present in that source. This motivates the development of an integrated framework combining question answering with fact verification.

Problem Statement

Existing retrieval-based question answering systems extract and summarize informationfrom given documents without validating the factual accuracy of the content. If the webpage contains incorrect or misleading information, the system may reproduce it

without verification.
The primary problem addressed in this project is:
To design and implement a retrieval-based system that answers user queries from a given URL and evaluates the correctness of the information present in the source using an automated fact verification mechanism.
The system aims to enhance reliability by assessing the truthfulness of claims found within the webpage content and providing a truth score along with a confidence level.

Objectives

The primary objective of this project is to develop an integrated system that performs retrieval-based question answering and evaluates the correctness of information present in a given webpage.
The specific objectives are:
1. To design a system that extracts textual content from user-provided URLs.
2. To implement semantic embedding techniques for representing webpage content in vector form.
3. To develop a retrieval mechanism that identifies relevant information based on user queries.
4. To generate structured and context-aware answers from the retrieved content.
5. To design an automated fact verification module that evaluates the correctness of the information present in the URL.
6. To assign a truthfulness score and confidence level for the evaluated content.

Scope

The scope of this project includes:
1. Accepting a webpage URL as input from the user.
2. Extracting textual information from the provided webpage.
3. Splitting and preprocessing the content for efficient semantic representation.
4. Generating vector embeddings and storing them in a FAISS vector database.
5. Retrieving relevant content based on user queries.
6. Generating answers grounded in the webpage content.
7. Evaluating the correctness of the retrieved information using an automated fact verification process.
8. Providing a truth score, confidence level, and explanatory justification.

The system is limited to textual data obtained from web pages and does not currently support multimedia inputs such as audio or video. The verification process relies on retrieved content and the language model's reasoning capabilities.

Research

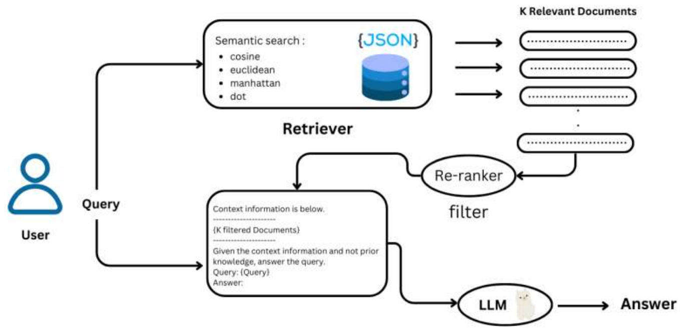
The key contributions of this project are:
• Development of an integrated framework combining Retrieval-Augmented Generation with automated fact verification.
• Implementation of a truth scoring mechanism to assess answer reliability.
• Integration of semantic similarity search using vector embeddings.
• Enhancement of transparency through source citation and confidence evaluation.

Proposed System

The proposed system is designed to integrate retrieval-based question answering with automated verification of information present in a user-provided webpage. The system accepts a URL as input, extracts textual content, retrieves relevant information based on user queries, and evaluates the correctness of the information using an automated fact verification mechanism.
Unlike traditional question answering systems that only generate responses, the proposed framework emphasizes assessing the reliability of the webpage content. The integration of semantic retrieval, generative modeling, and verification enhances transparency and trustworthiness.

System Architecture

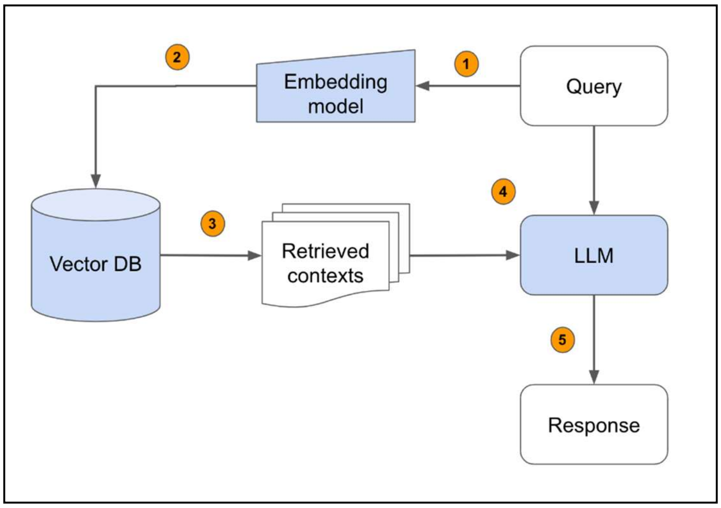
The system architecture consists of multiple interconnected modules that work sequentially.



Architectural Flow

1. User provides a webpage URL and a query.
2. Webpage content is extracted.
3. Extracted text is preprocessed and split into chunks.
4. Chunks are converted into vector embeddings.
5. Embeddings are stored in FAISS vector database.
6. Relevant chunks are retrieved based on query similarity.
7. Language model generates answer using retrieved context.
8. Fact verification module evaluates correctness of the information.
9. System outputs answer, sources, truth score, and confidence level.

System Flow:



Module Description

User Interface Module

The system uses Streamlit to provide an interactive interface. The user inputs:

- Webpage URL
- Question related to the webpage

The interface displays:

- Generated answer
- Source references
- Truth score
- Confidence level

This module ensures user-friendly interaction and real-time processing.

Data Acquisition Module

The system extracts textual data from the provided URL using a web content loader. The loader fetches and processes webpage text for further analysis.
This module forms the data input layer of the system.

Text Preprocessing And Chunking

Extracted text is divided into smaller chunks using a recursive character-based splitting method.
Parameters used:

- Chunk size: 1000 characters
- Overlap: 100 characters

Chunking ensures:

- Efficient embedding generation
- Better semantic retrieval
- Reduced token overload

Embedding Generation Module

Each text chunk is converted into a high-dimensional vector representation using a generative embedding model.
Let:

- **D={d1,d2,...,dn}**be document chunks
- **E(di)**be embedding of chunk **di**

Each chunk is mapped to a vector in semantic space.

Vector Storage Using Faiss

The embeddings are stored in a FAISS vector database. FAISS enables efficient nearest-neighbor search for high-dimensional vectors.
When a user submits a query qqq, it is converted into embedding E(q)E(q)E(q). The system retrieves the most similar document chunks based on cosine similarity.
Cosine Similarity is computed as:

$$\textbf{Similarity(q,di)}=\frac{E(q)\cdot E(di)}{\|E(q)\|\|E(di)\|}$$

Higher similarity indicates higher semantic relevance.

Retrieval-Based Question Answering Module

The retrieved chunks are passed to the language model along with the user query. The model generates a response grounded in the retrieved content.
This ensures that the answer is based only on the provided webpage rather than general model knowledge.

Integrated Fact Verification Module

The fact verification module evaluates the correctness of the information present in the retrieved content related to the query.
The verification process:

1. Identify factual claims in the generated answer.
2. Compare claims with retrieved webpage content.
3. Assign:
   - Truth Score (0–100)
   - Confidence Level (Low/Medium/High)
   - Explanation

This module improves reliability by assessing whether the webpage information supports the claims made in the answer.

Mathematical Model

Let:

- Q = User query
- D = Set of document chunks
- E(x) = Embedding function
- S(Q,Di) = Similarity score

Steps:

1. Convert all document chunks into embeddings.
2. Convert query into embedding.
3. Compute similarity scores using cosine similarity.
4. Retrieve top-k relevant chunks.
5. Generate answer using retrieved context.
6. Evaluate truth score using verification prompt mechanism.

ALGORITHMS

ALGORITHM 1: RETRIEVAL-BASED QUESTION ANSWERING

1. Input URL and user query
2. Extract webpage content
3. Split text into chunks
4. Generate embeddings for chunks
5. Store embeddings in FAISS
6. Convert query into embedding
7. Retrieve top-k similar chunks
8. Generate answer using language model

Algorithm 2: Fact Verification Process

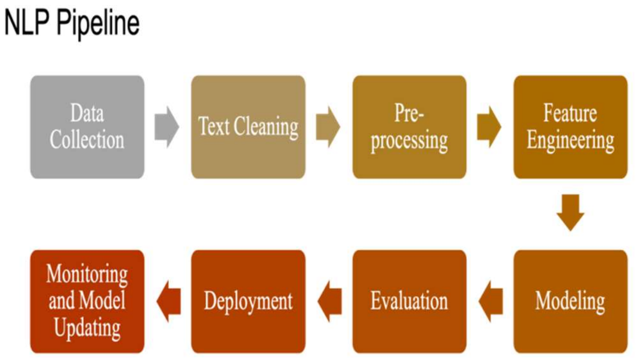
1. Take generated answer
2. Identify key factual statements
3. Compare statements with retrieved evidence
4. Evaluate consistency
5. Assign truth score
6. Provide confidence level and explanation

Advantages Of The Proposed System

1. Provides structured answers from specific webpages
2. Reduces blind reliance on generative AI
3. Evaluates correctness of webpage information
4. Enhances transparency with source attribution
5. Provides quantitative truth assessment

System Design

The system design defines the structural and functional organization of the proposed Retrieval-Based Integrated Fact Verification and Question Answering System. The design follows a modular approach where each component is responsible for a specific task in the processing pipeline. This modular architecture ensures flexibility, scalability, and ease of maintenance.
The system is designed to process a user-provided webpage URL, extract relevant information, generate answers to user queries, and evaluate the correctness of the information present in the webpage.



Design Overview
The overall system is divided into multiple layers, each performing a specific function. The flow of data begins from user input and progresses through processing, retrieval, generation, and verification stages before producing the final output.

The system is organized into the following layers:
1. Input Layer
2. Data Processing Layer
3. Storage Layer
4. Retrieval Layer
5. Application Layer
6. Output Layer

Input Layer
The input layer acts as the interface between the user and the system. It is implemented using Streamlit, which provides a simple and interactive web-based interface.
The user provides:
- A webpage URL
- A query related to the webpage

The input layer validates the inputs and triggers the execution of the system pipeline. It ensures that the system receives properly formatted data for further processing.

Data Processing Layer
This layer is responsible for extracting and preparing data from the given webpage.
Functions:
- Web content extraction
- Text cleaning and preprocessing
- Text chunking

The webpage content is extracted using a URL loader. The extracted text is then divided into smaller chunks using a recursive character splitting method. This step is essential to handle large documents and maintain contextual continuity.

Storage Layer
The storage layer is responsible for maintaining vector representations of the processed text.
Key Component:
- FAISS Vector Database

Each text chunk is converted into a vector embedding and stored in the vector database. This allows efficient similarity-based retrieval.
The storage layer ensures:
- Fast access to embeddings
- Efficient similarity computation
- Scalability for large datasets

Retrieval Layer
The retrieval layer performs semantic search based on the user query.
Functions:
- Convert query into embedding
- Compute similarity with stored vectors
- Retrieve top relevant document chunks

Cosine similarity is used to measure the relevance between the query and document embeddings. The most relevant chunks are selected and passed to the next stage.

Application Layer
This layer consists of two major components:
1. Answer Generation Module
- Uses a generative language model
- Produces context-aware answers
- Uses retrieved chunks as input

2. Fact Verification Module
- Evaluates correctness of information
- Compares answer with retrieved content
- Assigns:
  - Truth Score
  - Confidence Level
  - Explanation

This layer is the core of the system where both question answering and verification are performed.

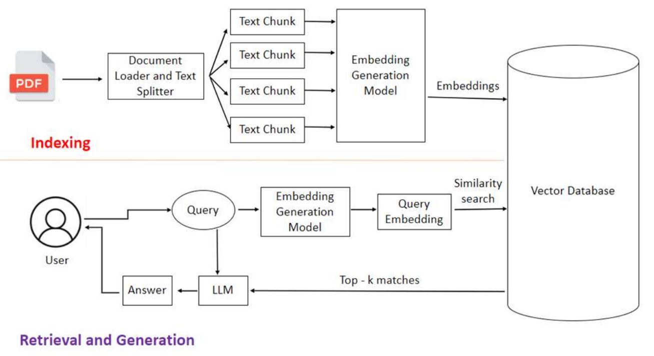
Output Layer
The output layer presents the final results to the user.
Outputs include:
- Generated answer
- Source references
- Truth score

- Confidence level
- Explanation

The results are displayed through the Streamlit interface in a structured and readable format.

Implementation

Architecture



Observations
From experimental evaluation, the following observations were made:

- Semantic retrieval improves relevance compared to keyword search.
- Chunking enhances contextual continuity.
- Truth scoring provides additional reliability assessment.
- The system effectively evaluates information present in the webpage.

However, verification depends on model reasoning and retrieved evidence quality.

Limitations Observed During Implementation

- Verification is limited to retrieved webpage content.
- Does not cross-check with multiple external trusted sources.
- Performance depends on quality of embedding model.

API-based model introduces response latency

Testing

Objectives
The main objectives of testing are:

- To verify correct extraction of webpage content
- To ensure proper functioning of embedding and retrieval process
- To validate answer generation based on retrieved data
- To evaluate correctness of truth score and confidence level
- To analyze system behavior under different input conditions

Test Case Design
The system was tested using structured test cases as shown below:

TEST CASE ID	URL TYPE	QUERY TYPE	EXPECTED OUTPUT
TC1	Wikipedia	Informational	Accurate answer + High truth score
TC2	Blog	Conceptual	Moderate score
TC3	Article	Definition	Relevant answer + Medium/High score
TC4	Mixed content	Analytical	Varying truth score

Summary of Testing
From testing and validation, it is observed that:

- The system performs reliable retrieval using embeddings
- Answer generation is context-aware
- Fact verification provides meaningful evaluation

System behaves consistently across different inputs

Results and Discussion

This presents the results obtained from implementing the proposed Retrieval-Based Integrated Fact Verification and Question Answering System. The system was tested using multiple publicly available webpages across different domains. The objective of the evaluation was to analyze retrieval relevance, answer generation quality, and the effectiveness of the fact verification mechanism.
The performance of the system is discussed in terms of reliability, transparency, and practical applicability.

Evaluation Criteria
Since the system focuses on both question answering and verification of webpage information, the evaluation is based on the following criteria:

1. Retrieval Relevance – Whether the system retrieves semantically relevant document chunks from the provided URL.
2. Answer Grounding – Whether the generated answer is based on retrieved evidence.
3. Truth Score Consistency – Whether the assigned truth score aligns with the supporting content.
4. Confidence Level Appropriateness – Whether the confidence level reflects evidence strength.
5. Response Clarity – Whether the generated answer is structured and understandable.

These criteria were analyzed qualitatively through multiple test cases.

Retrieval Performance Analysis
The use of semantic embeddings and FAISS vector search significantly improved retrieval quality compared to traditional keyword-based methods.
Observations:

- Queries with synonymous phrasing were successfully matched with relevant chunks.

- Chunk overlap helped maintain contextual continuity.
- Irrelevant sections of the webpage were filtered effectively during similarity search.

The cosine similarity-based retrieval ensured that only contextually relevant content was used for answer generation.

Answer Generation Analysis

The generated answers were:

- Context-aware
- Structured
- Grounded in retrieved content

Since the model was provided with retrieved chunks, it generated responses that reflected the webpage content rather than relying solely on pre-trained knowledge.

However, minor summarization variations were observed depending on how the language model interpreted the retrieved content.

Fact Verification Performance

The fact verification module evaluated whether the generated answer was supported by the retrieved webpage content.

Key observations:

- When the webpage clearly supported the claim, the system assigned high truth scores (above 85) with high confidence.
- When the retrieved content was partially relevant, the system assigned moderate scores (50–80) with medium confidence.
- If the retrieved content did not strongly support the claim, lower truth scores were assigned.

The explanation provided by the system helped interpret how the truth score was determined.

This mechanism improved transparency by:

- Quantifying reliability
- Providing justification
- Avoiding blind acceptance of webpage content

Comparison With Traditional Systems

Traditional Keyword-Based QA Systems:

- Rely on exact word matching.
- Do not evaluate correctness.
- Require manual interpretation of results.

Standalone Generative Models:

- Generate fluent answers.
- May hallucinate information.
- Do not verify webpage authenticity.

Proposed System:

- Uses semantic similarity for retrieval.
- Generates grounded responses.
- Evaluates correctness of webpage information.
- Provides truth score and confidence level.

Thus, the proposed system enhances reliability compared to conventional approaches.

Limitations Identified

Although the system demonstrates improved reliability, the following limitations were observed:

1. Verification depends on the quality of retrieved evidence.
2. The system does not cross-verify information with multiple independent sources.
3. Truth scoring relies on language model reasoning.
4. Performance depends on API response latency.
5. Only textual webpage content is supported.

These limitations provide scope for further improvement.

Summary Of Results

From experimental evaluation, it can be concluded that:

- Semantic retrieval improves contextual relevance.
- Answer grounding reduces blind hallucination.
- Fact verification enhances transparency.
- Truth scoring provides measurable reliability assessment.

The proposed system successfully integrates retrieval-based question answering with webpage-level verification, addressing the identified research gap.

Conclusion

This project presented a **Retrieval-Based Integrated Fact Verification and Question Answering System** designed to enhance the reliability of information extracted from user-provided webpages. The system integrates semantic retrieval techniques, generative language modeling, and automated verification mechanisms into a unified framework.

Unlike traditional question answering systems that focus only on retrieving or generating responses, the proposed system emphasizes evaluating the correctness of information present in the given URL. By incorporating vector embeddings and similarity-based retrieval using FAISS, the system ensures that answers are grounded in webpage content. The integration of a fact verification module further enhances transparency by assigning a truth score, confidence level, and explanatory justification.

Through experimental evaluation, it was observed that:

- Semantic retrieval improves contextual relevance.
- Answer generation becomes more structured and grounded.
- Fact verification increases transparency and user trust.
- Truth scoring provides measurable reliability assessment.

The system successfully addresses the identified research gap by combining retrieval-based question answering with webpage-level verification. This integrated approach contributes to improving reliability in AI-driven information systems.

Key Achievements

The major achievements of this project include:

1. Development of a real-time retrieval-based question answering system using semantic embeddings.
2. Implementation of efficient vector storage and similarity search using FAISS.
3. Integration of generative AI for context-aware answer generation.
4. Design of an automated fact verification mechanism.
5. Implementation of truth scoring and confidence evaluation.
6. Development of an interactive Streamlit-based user

interface.

The system demonstrates a practical application of AI techniques in enhancing trustworthiness of web-based information.

Future Work

Although the proposed system achieves its intended objectives, several improvements can be considered for future enhancement:

1. **Multi-Source Cross Verification** The system can be extended to verify information across multiple independent webpages rather than relying on a single URL.
2. **Integration with Structured Knowledge Bases** Incorporating external trusted databases such as encyclopedic knowledge bases can improve verification reliability.
3. **Improved Claim Extraction Mechanism** A dedicated claim detection model can be implemented instead of prompt-based identification.
4. **Support for Multimedia Content** Future versions may include verification of audio, video, or PDF documents.
5. **Performance Optimization** Caching mechanisms and optimized indexing can reduce API latency and improve response time.
6. **Quantitative Evaluation Metrics** Incorporating formal evaluation metrics such as precision, recall, and F1-score for verification accuracy would strengthen research validation.

These improvements can further enhance scalability, robustness, and real-world applicability of the system.

Final Remarks

The increasing presence of misinformation on the internet highlights the need for intelligent systems capable of evaluating content reliability. The proposed Retrieval-Based Integrated Fact Verification and Question Answering System demonstrates how retrieval techniques and AI-based verification can be combined to enhance transparency and trust.

This work contributes toward building more reliable AI systems and lays the foundation for further research in automated fact verification and trustworthy information retrieval.

References

[1] Patrick Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.
[2] Tom Brown et al., "Language Models are Few-Shot Learners," *Advances in Neural Information Processing Systems*, vol. 33, pp. 1877–1901, 2020.
[3] Jacob Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," *Proceedings of NAACL-HLT*, 2019.
[4] Mike Lewis et al., "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation," *ACL*, 2020.
[5] Douwe Kiela et al., "FAISS: A Library for Efficient Similarity Search and Clustering of Dense Vectors," *Facebook AI Research*, 2017.
[6] Facebook AI Research, "FAISS Documentation," 2023.
[7] Google Research, "Generative AI and Large Language Models," 2023.
[8] Chin-Yew Lin, "ROUGE: A Package for Automatic Evaluation of Summaries," *ACL Workshop*, 2004.
[9] William Yang Wang, "Liar, Liar Pants on Fire: A New Benchmark Dataset for Fake News Detection," *ACL*, 2017.
[10] James Thorne et al., "FEVER: A Large-scale Dataset for Fact Extraction and Verification," *NAACL-HLT*, 2018.
[11] Danqi Chen et al., "Reading Wikipedia to Answer Open-Domain Questions," *ACL*, 2017.
[12] Hamed Zamani et al., "Neural Information Retrieval: A Survey," *Information Retrieval Journal*, 2020.
[13] Stephen Robertson and H. Zaragoza, "The Probabilistic Relevance Framework: BM25 and Beyond," *Foundations and Trends in IR*, 2009.
[14] Yinhan Liu et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," *arXiv preprint arXiv:1907.11692*, 2019.
[15] Colin Raffel et al., "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *JMLR*, 2020.
[16] Iz Beltagy et al., "Longformer: The Long-Document Transformer," *arXiv preprint*, 2020.
[17] Sean Welleck et al., "Neural Text Generation with Unlikelihood Training," *ICLR*, 2020.
[18] Kenton Lee et al., "Latent Retrieval for Weakly Supervised Open Domain Question Answering," *ACL*, 2019.
[19] LangChain, "LangChain Documentation," 2023.
[20] Streamlit, "Streamlit Documentation," 2023.
[21] Google, "Gemini API Documentation," 2024.
[22] Natural Language Processing Survey Papers, Various Authors, 2018–2024.