

1955294-1828

1955294_c_17235.docx



Document Details

Submission ID
trn:oid::5531:111212598

Submission Date
Jun 2, 2025, 12:31 PM GMT+3

Download Date
Jun 2, 2025, 12:33 PM GMT+3

File Name
1955294_c_17235.docx

File Size
508.6 KB

66 Pages

15,061 Words

88,294 Characters





8% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.




Filtered from the Report

- Bibliography
- Quoted Text
- Crossref database
- Crossref posted content database

Match Groups

-  **97 Not Cited or Quoted 8%**
Matches with neither in-text citation nor quotation marks
-  **1 Missing Quotations 0%**
Matches that are still very similar to source material
-  **0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
-  **0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 5%  Internet sources
- 2%  Publications
- 6%  Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- 97 Not Cited or Quoted 8%**
Matches with neither in-text citation nor quotation marks
- 1 Missing Quotations 0%**
Matches that are still very similar to source material
- 0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 5% Internet sources
- 2% Publications
- 6% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	www.iitp.ac.in	2%
2	Submitted works	University Of Tasmania on 2024-10-20	<1%
3	Submitted works	University of Greenwich on 2024-08-16	<1%
4	Publication	Kutub Thakur, Helen G. Barker, Al-Sakib Khan Pathan. "Artificial Intelligence and ...	<1%
5	Submitted works	Liverpool John Moores University on 2024-08-27	<1%
6	Internet	etd.aau.edu.et	<1%
7	Submitted works	UCL on 2024-08-21	<1%
8	Internet	wiredspace.wits.ac.za	<1%
9	Publication	Satya Ranjan Mishra, Apul Narayan Dev, Alok Kumar Pandey, Mukesh Kumar Awa...	<1%
10	Internet	www.researchgate.net	<1%

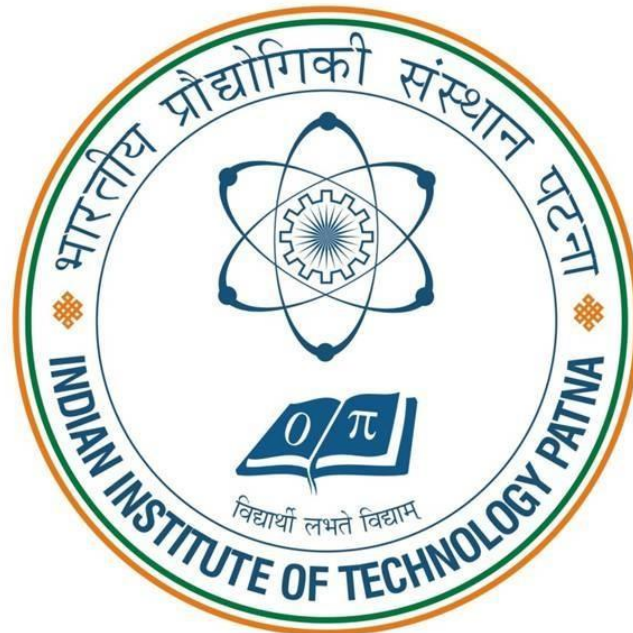
11	Internet	arxiv.org	<1%
12	Internet	oblako-media.ru	<1%
13	Submitted works	Botswana Accountancy College on 2022-02-21	<1%
14	Submitted works	UNESCO-IHE Institute for Water Education on 2020-01-06	<1%
15	Internet	aran.library.nuigalway.ie	<1%
16	Submitted works	University of Hertfordshire on 2024-11-18	<1%
17	Submitted works	University of Reading on 2024-08-23	<1%
18	Submitted works	University of Ulster on 2025-05-05	<1%
19	Submitted works	University of West London on 2025-05-24	<1%
20	Internet	repositories.lib.utexas.edu	<1%
21	Submitted works	University of Surrey on 2021-05-18	<1%
22	Submitted works	Mount Kenya University on 2019-05-09	<1%
23	Internet	core.ac.uk	<1%
24	Internet	impervaweb.com	<1%

25	Internet	conservancy.umn.edu	<1%
26	Internet	fastercapital.com	<1%
27	Internet	smartech.gatech.edu	<1%
28	Internet	www.cs.cmu.edu	<1%
29	Internet	www.freetopessays.com	<1%
30	Submitted works	University of Liberal Arts Bangladesh on 2024-10-23	<1%
31	Submitted works	Wayne State University on 2025-05-14	<1%
32	Internet	coek.info	<1%
33	Internet	ir.nctu.edu.tw	<1%
34	Internet	www.mdpi.com	<1%
35	Publication	Asri, Youssef El. "Vibration of Floor Structures Subjected to Crowd-Rhythmic Activ..."	<1%
36	Submitted works	City University on 2021-12-21	<1%
37	Submitted works	Kenyatta University on 2020-05-05	<1%
38	Submitted works	Liverpool John Moores University on 2021-03-15	<1%

39	Submitted works	Sardar Vallabhbhai National Inst. of Tech.Surat on 2024-12-10	<1%
40	Submitted works	UCL on 2025-04-25	<1%
41	Submitted works	University of Nottingham on 2024-09-05	<1%
42	Internet	acikerisim.isikun.edu.tr	<1%
43	Internet	discovery.dundee.ac.uk	<1%
44	Internet	drops.dagstuhl.de	<1%
45	Internet	eprints.utm.my	<1%
46	Submitted works	Bahrain Polytechnic on 2024-12-23	<1%
47	Submitted works	Bedford College on 2019-12-16	<1%
48	Submitted works	City University on 2020-12-21	<1%
49	Publication	Devaiya, Shraddhaben. "Expanding the Capabilities of a Bug Report Annotation T...	<1%
50	Submitted works	Liverpool John Moores University on 2021-06-08	<1%
51	Publication	Peng, Wenbo. "Women's Use of Complementary and Alternative Medicine for the ...	<1%
52	Submitted works	Ravensbourne on 2024-12-09	<1%

53	Submitted works	Swiss School of Business and Management - SSBM on 2025-03-20	<1%
54	Submitted works	University of Patras on 2024-09-23	<1%
55	Submitted works	University of Pretoria on 2005-10-04	<1%
56	Submitted works	University of Stirling on 2025-04-16	<1%
57	Submitted works	University of Utah on 2024-08-19	<1%
58	Submitted works	University of Warwick on 2007-09-09	<1%
59	Publication	Victoria Yaneva, Matthias von Davier. "Advancing Natural Language Processing i...	<1%
60	Publication	Yadav, Govind. "Enhancing the Accuracy of Large Language Models in Biomedical...	<1%
61	Internet	eprints.kfupm.edu.sa	<1%
62	Internet	journal.50sea.com	<1%
63	Internet	openaccess.city.ac.uk	<1%
64	Submitted works	Liverpool John Moores University on 2024-05-07	<1%
65	Submitted works	UT, Dallas on 2019-04-07	<1%
66	Publication	Liu, Zhijian. "Efficient Deep Learning With Sparsity: Algorithms, Systems, and App...	<1%

67	Submitted works	
University of Wales Institute, Cardiff on 2025-04-28		<1%
68	Submitted works	
University of Wollongong on 2024-03-12		<1%
69	Internet	
researchspace.ukzn.ac.za		<1%
70	Internet	
wpage.unina.it		<1%



1 THESIS MANUAL FOR THE M.TECH. PROGRAM

Indian Institute of Technology Patna



INDIAN INSTITUTE OF TECHNOLOGY PATNA

Evaluation Committee Approval

of a Thesis submitted by

Name of the student (in All CAPS)

The thesis of _____ has been read, found satisfactory and approved by the following DPPC committee members.

Name : 1 _____, Supervisor Date : _____

Name : 2 _____, Co-Supervisor Date : _____

Name : 3 _____, Member Date : _____

Name : 4 _____, Member Date : _____

Name : 5 _____, Member Date : _____

Name : 6 _____, Member Date : _____

Name : 7 _____, Member Date : _____

[THESIS TITLE GOES HERE]

A Thesis

Presented to

The Academic Faculty

by

[Student Name Goes Here]

In Partial Fulfillment

of the Requirements for the M. Tech. Degree



Indian Institute of Technology Patna

[MONTH YEAR of VIVA VOCE]

Copyright © Chandan Kumar 20xx

[To my beloved parents]



ACKNOWLEDGEMENTS

[Start typing here.] I wish to thank Dr. Issac Newton, my supervisor, for his invaluable guidance and support. I would also like to thank[Delete this paragraph.]



Certificate

This is to certify that the thesis entitled “THE TITLE OF THE THESIS”, submitted by NAME OF THE STUDENT to Indian Institute of Technology Patna, is a record of bonafide research work under my (our) supervision and I (we) consider it worthy of consideration for the degree of Master of Technology of this Institute. This work or a part has not been submitted to any university/institution for the award of degree/diploma. The thesis is free from plagiarized material.

Supervisor

Date: _____

Declaration

I certify that

- a. The work contained in this thesis is original and has been done by myself under the general supervision of my supervisor/s.
- b. The work has not been submitted to any other Institute for degree or diploma.
- c. I have followed the Institute norms and guidelines and abide by the regulation as given in the Ethical Code of Conduct of the Institute.
- d. Whenever I have used materials (data, theory and text) from other sources, I have given due credit to them by citing them in the text of the thesis and giving their details in the reference section.
- e. The thesis document has been thoroughly checked to exclude plagiarism.

Signature of the Student

Roll No:

TABLE OF CONTENTS

Contents

Indian Institute of Technology Patna	4
ACKNOWLEDGEMENTS	23
TABLE OF CONTENTS	26
LIST OF FIGURES	29
LIST OF SYMBOLS AND ABBREVIATIONS	30
ABSTRACT	31
Chapter 1: Introduction	32
1.1. Background and Motivation	32
1.2. Problem Statement	33
1.3. Objectives of the Study	35
1.5. Significance of the Study	38
1.7. Summary of Key Points	40
Chapter 2: LITERATURE REVIEW	41
2.1. Introduction to Literature Review	41
2.2. Theoretical Foundations	41
2.5. Critical Evaluation	48
2.6. Summary and Research Direction	49
CHAPTER 3: RESEARCH METHODOLOGY	51
3.1. Introduction to Research Methodology	51
3.2. Research Design	52
3.4. Model/Framework Development	54
CHAPTER 4: SYSTEM DESIGN AND IMPLEMENTATION	57
4.1. Introduction	57
4.2. System Architecture	57
4.3. Design Specifications	60
4.4. System Integration	62
4.5. Summary	64
CHAPTER 5: RESULTS AND DISCUSSIONS	65
5.1. Introduction	65
5.2. Experimental Setup and Parameters	65
5.2.1. Experimental Environment	65
5.2.2. Datasets Used	66
5.2.3. Experimental Variables and Parameters	66

	5.2.4. Experimental Procedure.....	67
	5.3. Presentation of Results.....	67
	5.3.1. Qualitative Results: Webpage Output Screens	67
	5.3.3. Discussion of Key Findings	69
	5.4. Comparison with Existing Work	70
	5.4.1. Comparison with Traditional Document Retrieval Systems	70
	5.4.2. Comparison with State-of-the-Art Question Answering (QA) Systems.....	71
	5.4.3. Improvements and Innovations.....	71
	5.5. Discussion	72
	5.5.1. Interpretation of the Results.....	72
	5.5.2. Significance of the Findings	72
	5.5.3. Implications for Theory and Practice.....	73
	5.5.4. Future Directions and Potential Enhancements	74
	5.6. Summary	74
	CHAPTER 6: CONCLUSION AND FUTURE WORK.....	76
	6.1. Conclusion	76
	6.2. Limitations of the Study.....	76
	6.3. Future Work	77
	6.4. Final Remarks.....	78



LIST OF FIGURES

Figure 1 Architecture	60
Figure 2 PDF Chatbot Interface	68
Figure 3 File Selection and Query Input Section	68
Figure 4 PDF Chatbot showing uploaded file and AI response to user query.	69
Figure 5 PDF Chatbot showcasing an uploaded file with user queries and AI responses	69

LIST OF SYMBOLS AND ABBREVIATIONS

- 34
NLP: Natural Language Processing
- ML: Machine Learning
- RAG: Retrieval-Augmented Generator
- CAG: Cache-Augmented Generator
- VSM: Vector Space Model
- TF: Term Frequency
- 44
BERT: Bidirectional Encoder Representations From Transformers

ABSTRACT

With the digital transformation of information that happened so quickly, there has been an accumulation of PDF documents through which knowledge flows. In this project, a very sturdy PDF Knowledge Extraction System was presented integrating the RAG and CAG models for intelligent and scalable document querying. It permits users to upload PDF files and then these files will be automatically parsed and segmented into chunks of content. Considered two parallel embedding pipelines: one uses Google Gemini 1.5 Flash API to generate high-quality embeddings for RAG model and the other uses HuggingFace models to cache in CAG framework.

Embeddings from both pipelines are kept in two different vector stores using ChromaDB, which guarantees rapid retrieval and response generation. When a query goes in, the system looks into the cache to see if any results are there. In case of a cache hit, an appropriate answer is returned immediately with just some milliseconds of latency. A cache miss presents where the query will be processed via RAG as well as be cached for later requests. This hybridization is mainly for optimized performance since RAG contributes its content awareness while CAG supports excellent efficiency, suitable for knowledge-agent type applications in a wide array of domains.

Submitted by:

Your Name Here

Under the guidance of:

Supervisor's Name Here

Chapter 1: Introduction

1.1. Background and Motivation

Overview of the Research Domain

With the data explosion in recent years, especially in PDF format, knowledge extraction from unstructured documents has become an area of research interest. Documents, reports, manuals, and academic papers are stored as PDFs-as much as possible-as they stand at being best terms of accessibility and processor. The traditional retrieval techniques practiced in these documents seem to fall short, relying mostly on indexing work performed manually or on crude keyword-based searches. These methods, glaringly, are incapable of digging into the full contextual richness of the documents and subsequently limiting their accuracy in producing responses or the value that they create for the user.

Modern approaches in NLP and ML have made great advances in intelligent automated forms of document processing. RAG and CAG are two powerful paradigms improving the ability to accurately query large document bases. These models combine external knowledge retrieval mechanisms with generation algorithms to query documents in a context-aware, efficient, and scalable manner. Modern embeddings such as those from Google's Gemini 1.5 Flash API, combined with these advanced methods, enable one to achieve state-of-the-art performance in knowledge extraction.

Importance and Relevance of the Topic

In a world where data powers every decision being made, the extraction of intelligence is increasingly becoming an important phenomenon. As digital content swells in number, every business, academic institution, and industry would want to increasingly rely on auto-mechanisms to retrieve and compile valuable insights from unstructured data really fast and accurate. The processing and easy querying of huge document sets become unquestionable with disciplines such as legal research, medical documentation, business intelligence, and academic research depending on them.

The growing digitization activities by more organizations and sectors have engendered a demand for systems smart enough to interpret and extract meaningful information from complex document formats. Conventional document management and querying strategies

rustily pump beyond the handling capabilities of volume and complexity of datasets. Meanwhile, therefore, a practical solution to these increasing demands lies in the increased ability of deep learning models, generative AI, and efficient embedding systems.

Additionally, the integration of CAG ensures that this knowledge extraction is both time-trusted and cost-effective and sufficiently resourceful to render systems that operate faster and more accurately than those working in response to ad hoc user queries from knowledge databases. This makes document knowledge extraction a current and practical field with far-reaching applications in areas requiring real-time intelligent decision-making from large data sets.

Motivation Behind the Study

The main reason behind such a study lies in the growing need for efficient knowledge extraction systems that can consider context on diverse and large-scale databases. Many document extraction systems still face limitations from performing complex queries or producing contextually relevant answers from a broad array of document types, even with recent advances in the world of AI.

With the ever-increasing preference for exchanging documents in PDFs, coupled with the complex and rich information they carry, only leaves room for further innovations in knowledge extraction. With the fusion of RAG and CAG models with Gemini 1.5 Flash embeddings, the study intends to target, further pushing the document querying systems on accuracy, efficiency, productivity, and pricier. Further storage of embeddings in ChromaDB allows the query responses to be optimized via smart caching techniques, thus, minimizing response time and maximizing the performance of the system.

In the end, this study aims of realizing a strong yet scalable PDF knowledge extraction framework that can be practically utilized by the industrial domain to enable consumers to quickly and expertly extract some useful knowledge from documents. Therefore, it would aid this research and provide solutions for end-users in sectors that utilize large-scale document data.

1.2. Problem Statement

Problem or Gap in Existing Knowledge

Despite advances in NLP and ML techniques, the extraction of valuable insights from unstructured documents such as PDFs remains a challenge. Current systems perceive the

issue to be the extraction of relevant context-based information from these documents, given the inherent complexity in their structure. Knowledge extraction methods existing so far employ the traditional keyword-based search engines or rule-based methods, which cannot truly grasp the semantic richness and contextual dependencies on which a given document is based.

Furthermore, most document retrieval and answering systems are inherently limited by scalability and efficiency. Embedding-based search methods may be on the rise, but these are still challenged by the implementation of any complex domain-specific query. Also, most approaches lack an efficient way to reuse already computed results, thereby performing unnecessary computations and slowing down the response time for further queries.

11 There exists a gap in the already existing knowledge in the integration of state-of-the-art generative models (e.g., RAG and CAG) with embedding-based retrieval systems to work specifically with PDF document processing. Current systems fail to blend these advanced techniques in a way that both allows for dynamic retrieval and the generation of contextually correct responses. Also, there is no integrated caching system that stores and retrieves previous results efficiently, which results in a degraded user experience, especially with poorly-scaled documents.

55 Scope and Limitations of the Study

In this study, the emphasis will be on constructing space for an intelligent framework for PDF knowledge extraction that bridges the gap by integrating Retrieval-Augmented Generator (RAG) and Cache-Augmented Generator (CAG) models with embedding-based document retrieval systems. The system is expected to handle larger PDFs consisting of breaking down documents into meaningful chunks via embeddings and subsequently using these chunks as the basis for semantic searching and context-aware query answering.

3 The scope of the study is delineated as under:

1. PDF Document Processing: The study will concern itself with the extraction of text from PDFs and subsequent segmentation of that text into smaller chunks for embedding generation.
2. Embedding Generation: The study will implement and compare embedding-generating practices using the Gemini 1.5 Flash API of Google and HuggingFace models for further contextual analysis.
3. RAG and CAG Integration: The system will use RAG for document retrieval and

CAG to cache results for frequently retrieved documents to minimize retrieval time and resource consumption.

4. Query Handling: The framework handles user queries by searching through embedded documents and providing answers contextual to the query.
5. System Performance: The study will measure the framework's performance with regard to accuracy, response time, and scalability.

There exist a number of limitations when it comes to the study:

1. Document Type Limitation: The system will work on PDF documents-based implementation. This may not generalize to other implementations based on Word documents or, maybe, simple web pages.
2. Large-Scale Implementation: This framework is scalable while the study will put emphasis on smaller subsets of documents; hence it may not fully exhibit the performance of the system working at a very large scale, say in real-world enterprise-scale work involving say millions of documents.
3. Computational Resources: Intense computational processes such as generating embeddings or RAG model inferences do pose serious constraints to being able to test the system across different environments or with a very large number of simultaneous users.
4. Domain-Specific Queries: The study will validate the framework in many general queries, but there will be situations when deeper contextual knowledge will be required for domain-specific queries which might be out of scope.

In essence, even though this study aims at filling the lacuna created in the current knowledge extraction approaches for PDF document querying by combining RAG and CAG, the study itself is limited in terms of document types, scale of implementation, and nature of domain-specific queries.

1.3. Objectives of the Study

Primary Goals

The primary objective of the study is in the ambit of IP to impart thick knowledge extraction capabilities for PDF to process big document sets at scale. This research wants to leverage RAG- and CAG-based techniques to enhance the download, query, and synthesis of meaningful insights from large volumes of unstructured PDF documents. The premise rests

on the thought that a modern approach combined with caching might help tide over some of the existing drawbacks in document retrieval systems, such as slow response time, poor context understanding, and not to mention limited scalability.

The study will also evaluate the performance of RAG-and-CAG fusion in the field, specifically during query precision, in comparison to processing times, and in respect of measurement of resource consumption. In so doing, it will establish a framework amenable to application in various industries for the extraction of knowledge from vast document databases with little computational overhead.

Specific Aims or Research Questions to Be Addressed

1. Develop into indeed robust document parsing and segmentation framework for PDFs: In detail, efficient ways of parsing text from PDF documents will be explored, and subsequently, how the content can be chunked for embedding in semantic searches and context-based queries. How do we get from PDF documents with putting all the data through preprocessing and structuring ready for embedding generation?
2. Let RAG and CAG Models Work for Optimal Knowledge Retrieval: Furthermore, under this umbrella, the research focuses on the integration of RAG, for dynamic document retrieval, with CAG, for caching query results. How does this integration make querying documents more efficient and faster? What is the impact of caching on performance?

By fulfilling these specific objectives, the research should provide a full solution to knowledge extraction from PDF documents and highlight better performances in retrieval, better performances in answer accuracy, and better performances in computation time.

1.4. Research Methodology

Brief Description of the Research Approach

The current study has adopted an experimental and developmental approach to design, implement, and assess a PDF Knowledge Extraction system that uses advanced retrieval and generative models. The methodology focuses primarily on employing machine learning techniques to embedding-based search, RAG, and CAG methods so as to address problems related to knowledge extraction from giant PDF documents. The approach follows the design and implementation of a prototype system, which is then empirically evaluated for its performance in terms of accuracy, efficiency, and scalability.

The research procedure is bifurcated into the following phases:

1. **System Design and Development:** It is the first phase during which a PDF document processing pipeline will be designed and implemented, including embedding generation mechanisms and RAG, as well as CAG, models for optimized querying.
2. **Evaluation and Testing:** The system, once developed, will be compromised into a suite of benchmarking activities and tested against a set metric of predefined requirements for query accuracy, response time, and scalability.
3. **Data Collection and Analysis:** During the data collection phase, we accumulate a large variety of PDF documents, queried on them in multiple ways, and analyze the ability of retrieval and generation models.

Techniques and Methods Employed

1. Data Collection Process:

- **PDF Document Corpus:** The PDF documents diversified in scope form that library or course of study so as to offer testing samples that might serve to gauge how well the system performs. They are chosen in order to span across academic papers, legal documents, technical manuals, and business reports so that the system could put variations through content types and query patterns.
- **User Queries:** Sample user queries will be produced, generic and domain-specific, to test for system output accuracy. They will check for whether the system is able to retrieve and reply with contextually appropriate answers to the question asked.

2. Document Parsing and Segmentation:

- **Text Extraction:** Extract text by parsing PDF using PyMuPDF or PDFMiner. Then, disintegrate the bigger texts into smaller units that bear some significance at the second level-whether paragraph, sentence, or topic.
- **Preprocessing:** It will exclude steps that remove irrelevant characters, fix formatting problems, or may apply special case handling, e.g., in any images or tables. Then, the document will be ready for embedding generation and query answering.

3. Optimization and Refinement

Based on the results of the evaluation, optimization strategies will be put into effect. In the broadest sense, optimization means fine-tuning the cache mechanism; fine-tuning RAG, CAG; and fine-tuning document segmentation and preprocessing methods so that performance in accuracy is increased.

Tools & Technologies Used:

- Python: Implementation of the system design, text extraction, generation of embeddings, and integration of the whole system.
- PyMuPDF, PDFMiner: For PDF text extraction.
- Google Gemini 1.5 API Flash: For generating good-quality embeddings.
- ChromaDB: Embeddings are stored to promote fast retrieval and caching.
- RAG and CAG Models: A recommender system for retrieval and generation of a document coupled with a cache mechanism.

An attempt has been made in this paper to make use of a hybrid approach of experimental development and performance evaluation to solve problems of document knowledge extraction. Using a newly developed machine learning approach coupled with modern tools for document processing, the study will try to provide a scalable and timely approach to knowledge extraction from large, unstructured PDF documents.

1.5. Significance of the Study

Contribution to the Field

The research presented in this thesis advances the field of knowledge extraction and document queries, especially for multifarious unstructured PDF documents. By marrying RAG along with and CAG models, a hybrid approach is forged that allows for the synergy between context-aware retrieval methods and the cost-efficiency of caching. This combination makes it possible to achieve greater accuracy and faster response times while also scaling well to systems that query large collections of documents.

Besides that, this study discovered a new way to use embeddings derived from state-of-the-art models such as Google's Gemini 1.5 Flash API and HuggingFace models, thereby bringing newer advancements to document retrieval systems in a production-level context. Integration with ChromaDB for vector storage and retrieval on a large scale makes the entire framework more scalable and therefore better suited for an array of knowledge-intensive applications in various domains.

This thesis further enriches the corpus of research in Natural Language Processing and Information Retrieval by providing a better alternative to document-based query solutions, particularly relevant in academic research, legal studies, medical records, and business intelligence.

Practical or Theoretical Implications

- **Practical Implications:** The study has wide-ranging practical implications. By creating a query system for PDF documents with scalability and efficiency, the system could be deployed into allied industries because it handles huge, unstructured repositories of documents. For example, gaining financial reports, legal documents, and policy manuals information for industries can be much faster. Likewise, scholars could exploit the framework to mine information from large databases of scientific literature.
- **Theoretical Implications:** From a theoretical angle, the presented paper enhances the knowledge about the employment of generative models in document retrieval. Contrasting RAG and CAG models under document querying gives rise to a new theoretical framework for designing efficient and intelligent document search systems. The results of this work will, therefore, foster further development in NLP and IR methods, with particular reference to document embeddings and generative models.

1.6. Structure of the Thesis

This thesis is organized as follows:

- **Chapter 1: Introduction**

This chapter gives an attitude overview of the study, including with the background, problem statement, objectives, and research methodology. Its aim is to set the research into context by highlighting the importance and challenges of knowledge extraction from PDF documents.

- **Chapter 2: Literature Review**

Provides a review of the research work related to document retrieval and knowledge extraction and also generative models like RAG and CAG. It examines existing systems, their limitations, and the gaps that this study is meant to fill. It serves to explore the background of embedding techniques and enhancements based on these techniques concerning document-based querying.

- **Chapter 3: System Design and Architecture**

Gives the overall design and architecture of the PDF Knowledge Extraction System. It describes the system components that include PDF parsing, text segmentation, embedding generation, retrieval mechanisms, and caching strategies. The integration of RAG and CAG models will also be discussed.

- **Chapter 4: Methodology**

Chapter 4 encompasses a view of an experimental setup, more comprehensively including

data collection, tools and technologies, and techniques employed for performance evaluation of a system. Evaluation metrics and designing experiments for testing the scalability, accuracy, and response time of a system are also discussed.

45

- **Chapter 5: Results and Discussions**

This chapter will present the results obtained during the evaluation phase of the implementation of the system, then complementing a description of the performance analysis particularly with respect to accuracy, response time, and scalability. Next, those methods will be compared with conventional document retrieval mechanisms and discussed accordingly for the research implications.

27

- **Chapter 6: Summary and Future Work**

This chapter summarizes the research, presents conclusions based on the work, and puts forward proposals for future work. The system will be discussed with respect to ways of improvement and the possibilities of advancing this work further in either more specialized document types or more generalized applications.

1.7. Summary of Key Points

- Building an intelligent knowledge extraction system for PDF documents using RAG and CAG models for enhanced query efficiency and accuracy is the aim of the proposed system.
- With the latest procedures considered in generating embeddings, the system offers a more appropriate way of retrieving relevant information from large unstructured PDF documents.
- This work tries to fill the gap existing document retrieval systems are lacking: providing the hybrid system where RAG enriches contextual awareness, and CAG produces faster responses to queries for information that is often requested.
- For the sub-components in the methodology, system design, data collection, performance evaluation, and optimization have shall be considered with regard to accuracy, scalability, and resource efficiency.
- It stands as an important matter because such a system could be used across domains like legal, business, academic research and could answer queries in a scalable environment for large document collections.
- The thesis is structured to cover: definitions of general background, literature review, system design, methodology, results, and conclusion, giving a bird's-eye view of the research and how it is tied to related works.

15

Chapter 2: LITERATURE REVIEW

2.1. Introduction to Literature Review

Purpose and Importance of the Review

This literature review aims to provide an exhaustive description of the already existing methods and research on extraction from unstructured documents, PDF files being an example therein. While considering document retrieval systems, NLP, and Machine Learning in their present forms, this chapter attempts to highlight systems' strength, weakness, and existing gaps. This literature review thus serves as a basis for the proposed system: a knowledge extraction system from PDFs justifying the employment of RAG and CAG models alongside advanced embedding techniques.

The analysis then sets the study with respect to the technologies and methodologies relevant to the area. It then makes further key advances in document parsing, embeddings, retrieval mechanisms, and generative models that will directly influence the final design of the system. It also sets out to ensure that the proposed system remedies current deficiencies while capitalizing on best-practices toward making a new contribution in document knowledge extraction.

2.2. Theoretical Foundations

This section aims to give elaboration on relevant theories and models related to the research topic. The knowledge extraction from unstructured documents saw higher growth with the introduction of more advanced machine learning techniques, particularly NLP and document retrieval. Thus, critical theories and models are the basis upon which the PDF Knowledge Extraction System is further developed.

1. Document Representation through Embeddings

Document embeddings are the basis of modern-day information retrieval systems that give dense high-dimensional representations of the content in the text. The theory of word and document embeddings considers the fact that similar words or similar documents should have similar representations in the embedding space. Word2Vec by Mikolov et al. (2013) and GloVe by Pennington et al. (2014) were among the first to bring this idea of vector representation of words to the forefront based on co-occurrence statistics of words in large corpora[1] [2].

BERT[3] and T5 [4] were introduced to exploit higher-level semantic relationships in document representation. It is ideally suited for document representation, allowing systems to

28

understand the contextual meaning of words and phrases inside whole documents. Embedding-based solutions have granted much efficacy regarding semantic search, document classification, and retrieval, as they allow queries and responses to be matched within a high-dimensional space.

For knowledge extraction from PDFs, advanced embeddings such as provided by Gemini 1.5 Flash API from Google and HuggingFace models supply a high-quality and scalable solution to document retrieval and context-aware query answering. Such embeddings allow semantic search where a user query can be mapped into the same embedding space that makes the matching more intuitive and accurate.

2. Retrieval-Augmented Generation (RAG)

The RAG model [5] which stands as a hybrid approach, aims to enhance response quality in NLP tasks by combining information retrieval (IR) with generative models. In regular generative models such as GPT [4], text is generated based on a given input, but the model has no actual access to any extra knowledge other than that learned through pre-training of parameters. What RAG does is that it retrieves passages relevant to the query from a document store (or database) to condition the generation process, thereby ensuring that the output is based on an external knowledge source.

The RAG architecture is typically two-fold-a retriever queries the database for relevant documents, while the generator composes an answer based on both the query and the documents retrieved. This allows answers that can be supported by the context provided by the documents, which distinction is tagging especially important in summarization renaming tasks and query answering. Incorporating RAG into the proposed system allows intelligent querying of large-scale document databases, wherein documents are first retrieved in accordance with semantic relevance, then used for context-aware generation of accurate answers.

3. Cache-Augmented Generation (CAG)

Building Universe-CAG extended with caching to enhance response time and compute resources, much like RAG. In the adage "Unlike in the usual RAG," every query requires the retrieval of documents from a gargantuan store, so the process is both computationally expensive and time-consuming. CAG stands against this by caching results corresponding to frequently called documents or queries. When a similar query is handled, the cached response can be retrieved in no time with no need for further retrieval, thereby drastically improving application latency and negatively impacting fewer computing resources.

At the integration point, CAG uses caching so it can marry the strengths of retrieval-based systems with the efficiency of generative systems. By caching intermediate retrieval results and reusing them wherever applicable, CAG improves scalability and performance for knowledge extraction systems. This will be important in cases where the same queries are made repeatedly or where documents are queried often, engineers' common situation in knowledge-intensive industries.

4. Embedding-Based Search and Vector Databases

The theoretical basis for embedding search hinges on vector space models and the concept of inter-vector similarity. Traditional IR methods, including the vector space model [6], treated a document on the basis of term frequency occurring in it. The modern embedding-based search further extends this analogy by representing documents as vectors in some continuous high-dimensional space. Embeddings are highly advantageous for document search, allowing systems to return documents similar in meaning even if none of their keywords match.

Vector databases like ChromaDB are built around the idea of large-scale embeddings, with the ability to store, index, and query the embeddings efficiently. ChromaDB enables very fast nearest neighbor searches in the embedding space-sharp-scalable real-time document retrieval would never have been complete without it. By storing embeddings generated by the RAG and the CAG models, ChromaDB provides for extended queries that comprehend context and serve accordingly in front of the am user [7].

Historical Development of the Field and Major Milestones

The field of document retrieval and knowledge extraction has faced key landmark events:

1. On Early Information Retrieval Systems

First-generation retrieval systems were Boolean models and term frequency-descriptions[6]. The majority of the information retrieval problems arose because such systems envisioned no semantic relations between words and phrases within any document.

2. Vector Space Models

VSMs represented another turning point in document retrieval at the end of the seventies and the beginnings of the eighties [6]. It treated documents and queries as vectors in the high dimension, allowing for a higher level of similarity measures by distances among these vectors.

3. Word Embeddings and Deep-Learning Techniques

The word-embedding revolution was made possible by Word2Vec [1] and GloVe[2], which put forth dense distributed representations of words encoding some semantic relationships. Document retrieval was further empowered with Transformer models, specifically BERT [3]

and T5 [4], by deeply contextualizing the meaning of a word in a document.

2.3. Review of Related Studies

Synthesizing Important Knowledges of Studies, Experiments, and Papers in the Field

Many studies have been instrumental in the development of knowledge extraction from unstructured documents, especially focusing on document retrieval, NLP, and ML. In these studies, some of the major ones informing the current research are as follows:

1. Document Retrieval and Knowledge Extraction Models

One of the earliest and fundamental works on document retrieval was done by Salton et al. (1975), who put forward the concept of VSM. The VSM considers both documents and queries as vectors in a high-dimensional space so that document retrieval could be based on similarity. Even though it was the basis for modern document retrieval, VSM used a crude TF representation, which totally ignores any semantic relations between terms.

Mikolov et al.-in-2013 introduced the Word2Vec, which goes beyond term-based models, learning distributed vector representations of words. The system can identify semantic relations depending on how the words are used in context in large corpora [5]. Meanwhile, Pennington et al. (2014) put forward GloVe, a sort of co-occurrence-based model; however, this theory is more developed when focusing on global co-occurrence statistics of words. The sources who best cater to the word-level semantic information would greatly contribute towards document retrieval improvement. However, these embeddings still fall short when handling large document corpora due to the lack of considering broader contextual relationships across full documents [2].

Devlin et al. (2018) came up with the BERT model, capable of registering deep contextual relationships among the words in the text. Different from the usual kind of embeddings, the BERT model is bidirectional as it considers the left and right context of each word in a sentence, which makes it more efficient when handling polysemy and word order issues. Lately, BERT-based models have ushered in a state-of-the-art approach to document retrieval and question answering [2].

An extremely important paper in 2020 by Lewis et al. presents RAG as a means of generating higher-quality responses-that is, by bringing together the technically heterogeneous fields of information retrieval and generative models [8]. RAG retrieves documents relevant to an input query and generates answers on contextual-factual grounds using a generator model. This kind of hybrid approach is useful for a number of systems that rely on context-aware and accurate knowledge extraction from source documents, such as chatbots, automatic

summarization, or question answering.

2. Embedding Search and Creation of Knowledge Bases

There are several embedding search papers that focus on document retrieval performance with vector databases and embeddings. ChromaDB can work as a vector database. ChromaDB seeks to optimize storage and semantic search such that documents are retrieved on the fly with respect to their similarity to a query in a high-dimensional vector space. Using this idea of ChromaDB, operations of nearest-neighbor search algorithms like k-NN could be performed and have dominated recent NLP tasks [7].

Cache-Aided Generation is yet one more frontier brought into a research limelight by such person(s) as Agrawal et al. (2021), who describe the system as one which caches frequently accessed documents, thereby reducing the number of times repeated queries would be sent to the document store. The CAG approach shows promise when response time and computational efficiency are paramount in a system requiring large-scale knowledge extraction [9]

3. PDF Knowledge Extraction and Semantic Document Understanding

The following are some works entirely devoted to knowledge extraction from PDF, the extra difficulty being that PDFs are unstructured and non-uniform in their layout. Tkaczyk et al. (2020) proposed a system to automatically retrieve structured data from scientific articles in PDF format. The method involves computer-vision and NLP techniques: First, there is a layout analysis, which feeds deep-learning NLP models for the data extraction from tables and references from academic articles. However, in practice, the applicability of the model is still limited due to differences in document structures and the inability to handle heavily formatted PDF documents [10].

In addition, Jing et al. (2019) presented a PDF-to-text conversion pipeline for knowledge extraction by a combination of rule-based systems and deep learning. The pipeline intended that information such as citations, figures, and equations recognized and extracted; however, the approach is highly dependent upon the quality of the initial text extraction and inconsistently handles various formats, especially in non-textual PDFs [7].

Analysis of Gaps and Weaknesses in Existing Research

While all the research efforts briefly mentioned have led to major developments in document retrieval systems and knowledge extraction, several gaps exist in this field:

1. Handling Non-Textual Content in PDFs

One significantly emphasized gap in the literature is the problem of extracting

knowledge from PDF files containing rich non-textual material such as images, tables, and mathematical formulas. Most methods in the literature describe and deal with text extraction, thus creating a gap of systems that can seamlessly handle complex PDF structures and non-textual information.

2. Scalability, and Real-Time Processing

Embedding-based methods, i.e., BERT, have been found to retrieve documents quite well, but scaling to a bigger document database is a challenge with respect to real-time response. Till this time, efficient document retrieval for large-scale systems remains a challenge, especially when processing documents with highly diverse, unstructured formats such as PDFs.

3. Integration of Hybrid Models

RAG-type hybrid models that mix information retrieval approaches with generative ones have proven to increase the quality of responses. However, the integration of retrieval-based models and deep learning systems for knowledge extraction is still underrepresented, with PDFs as the application domain being an obvious place to look. The research of how best to integrate generative models with retrieval-augmented systems for real-time document extraction tasks is in its infancy.

Overview of Methodologies Used in Similar Studies

Depending on different factors such as the source of documents, the end goal of extraction, and the various extraction techniques, the methods employed by analogous research drastically differ. However, some methods consist of:

- **Deep Learning Model:** Most works are in transformer-like architectures, such as BERT and GPT, for long document understanding and extraction. These models are first trained on massive corpora before being fine-tuned on a comparatively smaller specialized corpus for the downstream tasks: document classification, entity recognition, or semantic search.
- **Information Retrieval Model:** Most embedding models are used in document retrieval with models such as Word2Vec, GloVe, and BERT. These models provide, instead of keyword-based search, vector representations to match documents and queries semantically.
- **Hybrid Systems:** Hybrid approaches are another method with the Information Retrieval combined with generation models. Retrieval-augmented models (e.g., RAG)

are commonly used when there are multiple sources of knowledge for a given task, e.g., document summarization or question answering.

- **Computer Vision Models:** For knowledge extraction from PDFs, computer vision techniques are applied to images, graphs, and tables as non-textual elements. CNNs work well in layout analysis and text recognition.

2.4. Recent Advances and Trends

Having evolved within the last several decades, the field of document retrieval and knowledge extraction has enormously flourished. These recent developments are now taking this field in new directions and creating opportunities for development in real-time, scalable, and context-aware systems. Some of the major technological changes, tools, and variants being introduced, which affect even the research, are:

1. Transformer-Based Models and Pretrained Language Models

In other words, transformer-based architectures such as BERT, GPT, and T5 have changed the way treatment was handled and primarily improved the performance of many NLP tasks such as text classification, question answering, and document retrieval. BERT attending bidirectionally allows the attention mechanism of the model to consider the left and right side context of a word concurrently. This method is superior to treat more complex syntactical structures and ambiguities of words in sentences. More recent efforts are RoBERTa[11] and ALBERT [12] which attempted to advance BERT in various ways, mainly in efficiency and modeling scale.

With its 175 billion parameters, GPT-3 has likely been the most significant single landmark in natural-language generation. The ability of GPT-3 models to do few-shot learning has also opened up opportunities for the use of these systems in a wide range of document-related tasks such as knowledge extraction and generation from unstructured data or text. These improvements have further pushed pretrained language models into a very important role for document processing and retrieval across various tasks, advancing the document-based question answer and summarization state-of-the-art [13].

2. Combination approaches: Retrieval-Augmented Generation (RAG)

RAG models have greatly increased the systems' stand in combining information retrieval with generative capabilities. By large-scale document retrieval integration with generative language models, RAG systems have provided greater contextually informed and relevant responses to the weaknesses of pure generative approaches such as GPT, which sometimes would fabricate information when such information is absent from its training set. This hybrid approach has fared well in open-domain question answering and knowledge-intensive

issues and is therefore a major innovative step in knowledge extraction [8].

3. Knowledge Graphs and Semantic Web Technologies

Another big trend is the integration of knowledge graphs with document retrieval systems. Knowledge graphs organize entities and their relationships in a structured way to facilitate domain knowledge extraction and use. Reports stand that graph-based retrieval systems allow the semantic search by the relationships between entities rather than by keyword matching. Since then, there have been developments in the usage of GNNs that learn and utilize the underlying graph structure of the data to help improve search accuracy [12]

The next important evolution to be noted is the maturing of semantic web technologies for linking data across open-ended domains. This allows a more complete knowledge extraction process, where a system extracts, links, and presents information from diverse sources. Ontologies united with knowledge graphs have come to be the main form of knowledge representation used for knowledge extraction in domain-specific applications, which include healthcare, legislation, and scientific research [14].

2.5. Critical Evaluation

Comparative Analysis of the Strengths and Weaknesses of the Reviewed Works

Recent advancements in the areas of document retrieval and knowledge extraction, including transformer models, hybrid approaches, and the integration of knowledge graphs, have greatly improved the efficiency and accuracy of information retrieval systems. While that said, some challenges and limitations still exist in the current literature:

1. Scalability Issues in Transformer Models

Transformer-based models such as BERT and GPT have set a standard for understanding of documents but, training and inference, are large computational resources. Transformation-based model fine-tuning and deployment are cost-creators and turn-about accessibility into a real-life nightmare for smaller organizations. As the dataset grows, the retrieval time aggravates exponentially, making it an even harder problem to solve scalability-wise.

2. Working with Non-Textual Data

While extraction has seen its share of advances on the textual front, there is still little research done for integration between non-textual data (e.g. images, tables, graphs) in document extraction. This is vital to domains such as scientific documentation or legal documents, where data is often presented under complex formats. Most of the models out there really deal with just text extraction, thus leaving a big gap toward

handling multi-modal documents.

3. Accuracy vs. Real-Time Processing

Challenges precede, where hybrid systems like RAG improve response relevance but require extensive retrieval of data, thereby slowing down real-time processing. Immediate response applications, such as chatbots and digital assistants, are faced with a huge challenge concerning the accuracy versus processing time.

4. Lack of Evaluation Standards

One other major problem is the lack of any standards for evaluating knowledge extraction systems. Usual metrics like precision, recall, and F1 score are applied but in reality, they do not determine the goodness or effectiveness of a system, mostly when a system aims at extracting structured knowledge from documents of various file formats. Without appropriate evaluation techniques, it is not possible to arrive at solutions that can be implemented worldwide.

Identify the Research Gap Your Study Addresses

The current literature outlines some of the significant gaps that require attention to proceed with improving knowledge extraction from unstructured documents:

1. Multi-Modal Integration: Most of the systems are still text-based extraction systems and cannot handle multi-modal documents where one can have a mixture of text, images, and tables. This is especially pertinent in scientific research, healthcare, and law situations where other more complex documents require a further level of integration between various data types.
2. Efficient Real-Time Knowledge Extraction: Models like BERT and GPT excel at document retrieval but are very resource heavy, thus discouraging, to say the least, their application in real-time for large-scale settings. Efficient models are those expected to work well under real-time conditions.
3. Application-Specific Solutions: Another less explored area in the field of knowledge extraction systems is application-specific domain use cases. In fact, adapting such models for the peculiarities of legal documents, medical documents, or scientific documents greatly increases the possibility of extracting much finer actionable knowledge.

2.6. Summary and Research Direction

To summarize, with recent developments in transformer models, hybrid architectures, and knowledge graph manipulations, an extraordinary knack was imparted to the domain of

document retrieval cum knowledge extraction systems. However, these systems still grapple with issues of big scaling, multi-modal data handling, and real-time processing. After delving into the major technological currents running parallel to the domain and identifying some crucial gaps in the landscape of present research, the study now intends to work on those gaps. It wants to create a more efficient and scalable system for knowledge extraction to cover multi-modal documents while being able to provide insights in real-time. The next chapter shall be devoted to laying out the methodology for the proposed system, with emphasis on the process of data collection, model selection, and evaluation framework.

CHAPTER 3: RESEARCH METHODOLOGY

3.1. Introduction to Research Methodology

Overview of the Research Approach and Design

The methodology of the study is fundamentally aimed at developing and evaluating an intelligent PDF knowledge extraction system using such advanced models as Retrieval-Augmented Generation or RAG and Cache-Augmented Generation or CAG. Thus, the research is oriented towards investigating a qualitative generative model in conjunction with embedding-based retrieval systems for better document querying capacity.

The method applied is mostly quantitative, focused, primarily on the system's performance evaluation through accuracy, response time, and scalability. By means of an iterative approach to develop the system, the intermediate results are regularly assessed and adjusted until optimal retrieval and caching strategies are developed to guarantee efficient and context-aware end-user responses.

Other qualitative perspectives are also implemented to some degree during the design phase to ensure the architecture is best suited to the peculiarities of differing document classes, especially PDFs. Hence, this hybrid approach supports the integration of both quantitative-performance and qualitative-user-experience considerations into the resultant design.

Research methodology consists of three phases:

1. System Development and Design: In this stage, the general design of the system is done, including the PDF parsing module, embedding generation pipelines, integration of the RAG and CAG models, and the caching layer itself.
2. Performance Evaluation: The system is evaluated against the different parameters of key performance indicators, such as precision, recall, response time, scalability, etc. It is possible that the best performance of the system is evaluated, in a real scenario, considering different dataset sizes and complexities of queries.
3. Analysis and Discussions: The final phase intends to discuss the pros and cons of the proposed framework after the analysis of the performance evaluation results. Lastly, the system is juxtaposed with other document retrieval systems to analyze the merits of this implementation and those which require improvement.

Justification for Choosing the Methodology

The workflow choice is geared towards handling the peculiar challenges of document knowledge extraction from large-scale PDFs. Traditional mannerisms of document retrieval

often lack context-awareness and scalability, whereas newer methods using deep learning models are still not resource-efficient or require longer response times.

This study firstly addresses performance measures using a quantitative approach and, on the flipside, treats design issues qualitatively; this is so both the technical and user experience sides of the system get due consideration. Secondly, the integration of RAG with CAG models extends the art of document retrieval and caching techniques, thus providing a new solution to a problem.

3.2. Research Design

Type of Research: Exploratory, Descriptive, Analytical, etc.

The study is essentially an exploratory and analytical one. It attempts new ways for document retrieval and knowledge extraction from PDFs, for instance, through the usage of RAG and CAG models. The research is analytical, too, as it tries to rigorously test the system's performance through well-established metrics to compare it with traditional document retrieval techniques and to identify its areas of weakness.

- Exploratory. Attempts at the study are made to throw the light on the possibility that generative models and embedding-based retrieval systems could be a new use case: efficiently and contextually querying large PDFs. Hence, time is to be spent in learning how such models could work with large document databases and performing testing of their effectiveness in real-world situations.
- Analytical. The analytical inquiry relates to the confirmation of system efficiency with reference to technical-based performance, efficiency, and scalability metrics. This would hence imply carrying out a quantitative analysis based on performance metrics such as precision, recall, response time, throughput, which indicate the constitution of the system in terms of its ability to meet demands posed by large corpus of complex queries.

Explanation of the Research Strategy

1. The research followed an iterative and methodical developmental approach:
 1. System Development and Prototyping: At this stage, the study involves the overall designing and implementation of the knowledge extraction system. It includes the preparation of the PDF parsing module, splitting of PDF documents into chunks that actually make sense, and integration of two embedding pipelines: Google's Gemini

1.5 Flash API and HuggingFace models.

2. Evaluation and Testing: Testing should be conducted after the system is ready. Considered complex will be evaluating the system's handling of specific queries over a corpus of documents of various types-academic papers, legal texts, technical manuals, etc. A test is run for accuracy and time taken to respond and scalability against the given set of queries.
3. Analysis and Optimization: The system shall be improved iteratively based on optimization results of performance testing. Optimization can encompass caching policies, methods used for generating embeddings, and retrieval procedures.
4. Comparison with Existing Systems: After the tests, the system will be qualitatively and quantitatively compared with other traditional document retrieval systems (for example, keyword search engines) and compared with one or more state-of-the-art models. This will provide insight into how better this framework works in relative terms from the research perspective.

3.3. Data Collection Methods

Description of Primary and Secondary Data Sources

For this study, the data collection does not employ usual methods such as surveys, interviews, or direct field experiments. Data is rather directly sourced from a diverse collection of public PDF documents that serve as test documents for the proposed knowledge extraction system.

The PDFs to be considered in the study will be from divergent domains so that the system shall prove to be robust as well as versatile. For example, they shall include research papers, technical manuals, legal documents, research reports, and business intelligence reports. The logic behind choosing such types of documents is to try to implement types of content that users may normally need to query, thus, forcing evaluation of the system on various writing styles and subject matters.

Within the study, the PDFs are not proprietary or confidential but rather perfectly public: they can be downloaded by anyone from reputed sources such as:

- ResearchGate (academic papers and research reports)
- arXiv (research papers in disciplines of science)
- ProjectGutenberg (for classic literary works)
- Government websites (for legal documents and publicly available reports)

Such document types form the basis of a full-scale testing regime to determine how well the system deals with various formats, terminologies, and styles of text.

Instruments and Tools Used for Data Collection

Since the development and testing of an automatic document processing system comprise the main thrust of the study, the "data collection" activities translate to the automatic parsing of PDF(s) selected and straightforward extraction of document content into structured text.

Tools and instruments that prove helpful are the following:

1. **PDF Parsing:** These libraries are used for extracting raw text from PDFs: PyMuPDF (a.k.a. Fitz) or pdfplumber. They can handle complex layouts as well as tables and images that are embedded in the documents.
2. **Embedding Generation:**
 - **Google Gemini 1.5 Flash API** is used to get high-quality embeddings for text extracted from PDFs.
 - HuggingFace transformers for alternative embeddings and text vectorization, which lies within the Cache-Augmented Generation (CAG) system.
3. **ChromaDB:** This is a vector store that stores and retrieves generated embeddings for fast querying and caching of document chunks.
4. **Querying Tools:** RAG-model-based processing of user queries returns information relevant to the stored embeddings.
5. **Testing Frameworks:** For the evaluation of performance, tools such as pytest are employed to automate tests designed for system response and offer metrics regarding query accuracy, response time, and system scalability.

3.4. Model/Framework Development

Overview of Models or Frameworks Employed in the Research

The research mainly concentrates on the development of an Intelligent PDF Knowledge Extraction System consisting of two advanced models, namely **Retrieval-Augmented Generation (RAG)** and **Cache-Augmented Generation (CAG)**, aimed at piecing together contextually relevant information and appropriate querying mechanisms for large-scale documents.

The key constituents of the system are:

1. **RAG Model:** The model carries out the retrieval of documents. Embeddings search for relevant information inside the document vector store (ChromaDB). The generative model then continues and generates a response for each relevant document

or segment such that the final answer is contextually coherent.

2. **CAG Model:** The CAG model boosts the RAG model by caching answers for frequently queried documents or segments. If a query matches an answer in the cache, it will return that answer immediately without going through the retrieval process again, thus saving a great deal of latency.

The process of embedding generation harnesses the models of Google's Gemini 1.5 Flash API and HuggingFace's pre-trained models to ensure the creation of the best possible vector representations of the document content. These embeddings are loaded into the vector store (ChromaDB) to be exploited by both the RAG and CAG models.

Explanation of Algorithms, Formulas, or Systems Used

- **RAG Model:**
 - In retrieval, a nearest neighbor search finds either the most relevant document or chunk. This search is conducted into the vector space given by the embeddings, using cosine similarity or some other distance metric.
 - In generation, transformer-based models (GPT-3 or others) give a coherent response from the retrieved context.
- **CAG Model:**
 - Caching will use an LRU strategy for cache eviction to keep responses that are frequently and recently accessed by the system.
- **Embedding Generation:**
 - Embeddings are created feeding semantic representation from the Google Gemini 1.5 Flash API, while the HuggingFace models help speed up the caching and contextual analysis.
 - Embedding vectors are stored and indexed within ChromaDB, which is a highly optimized vector store to ensure fast retrieval.

Together, these elements act to formulate a system that can query documents efficiently and provide answers that are contextually relevant while keeping the response time down.

3.5. Limitations and Assumptions

Discussion of the Research Limitations

1. **Limitations:** The most focus being laid on PDF documents in this study may have disallowing complete generalization of the conclusions to Word, HTML, or LaTeX, whose structure and representations of texts may be somewhat different.
2. **Domain-Specific Knowledge:** While the system can work with more of the general-type questions that are posed to it, it may fail with domain-specific queries requiring

insider knowledge in that particular field-e.g., terminologies in medicine or legal jargon.

3. **Computational Resource Constraints:** Running the entire advanced generative model (especially RAG) and performing embedding generation is expensive computationally. The study was limited by the computational resources available to it, thereby affecting scaling potentials of the system and its real-time response.
4. **Data Diversity:** The sample set of PDFs used for testing traverses different topics and document types but may not encompass every set of use cases and thereby limit generalization of the results.

Assumptions Made During the Research Process

1. **Embedding Quality:** It is assumed that the embeddings generated by the Google Gemini 1.5 Flash API and HuggingFace models will give a high-quality semantic representation of the document text that pertains strictly to retrieval accuracy.
2. **Document Segmentation:** The study assumes that the automatic splitting of documents into manageable chunks will hold well for very large documents. Considering PDF parsing libraries pretty comfortable to use, the study assumes that the segmentation would be precise enough so that the querying will provide relevant context.
3. **Cache Efficiency:** The assumption is that the CAG caching technique can effectively reduce the response time by storing the result of resolved queries and later reusing them, whenever needed.

3.8. Summary

In this chapter, the basic approach of research for developing the PDF Knowledge Extraction System is described. The research exploits RAG and CAG models for context-aware and efficient querying of documents. Embedding generation through Google Gemini 1.5 Flash API and HuggingFace models and storing the embedding by ChromaDB is the prime function of the whole system.

System evaluation is made with important metrics such as accuracy, precision, recall, F1 score, response time, and scalability. This research also touches upon limitations and assumptions under which the necessity arises, such as being restricted to working with PDF documents and having access to computational resources.

The next chapter will focus on the Implementation of the Proposed Framework: it will present the architecture of the system, the integration of the system components, and the technical implementation and deployment of the PDF Knowledge Extraction System.

25

CHAPTER 4: SYSTEM DESIGN AND IMPLEMENTATION

4.1. Introduction

Overview of the System or Solution Developed for the Study

This chapter deals with the design and implementation of the PDF Knowledge Extraction System. This system is designed to extract and generate answers from large-scale PDF documents on a query-by-query basis. Leveraging the combined powers of RAG and CAG models ensures that the system remains very highly accurate but also maintains very low response time, thereby completing the entire cycle of relevance and efficiency in answering queries from unstructured content in PDF form.

Embedding generation considers the semantic meaning of document text and advanced search to retrieve relevant information. It then utilizes a generative model to answer in natural language, either from the retrieved documents or sections. Furthermore, an inherent caching system (CAG) fast-tracks results caching and retrieval for frequently asked questions.

Objectives of the System Design

- Automatically retrieve and answer user queries from PDF documents, with no human intervention.
- Ensure maximum performance with a simultaneous application of retrieval and generative models (RAG and CAG) for real-time information processing.
- Provide scalability to deal with large corpora of documents and a variety of document types, especially PDFs.
- Minimize response time using caching mechanisms while keeping response quality and accuracy in check.
- Offer nice, clean, and easy-to-use UI for the user to interact with the system and submit his/her queries.

4.2. System Architecture

The system architecture comprises several interconnected elements and components working together to fulfill efficient document retrieval and response generation purposes. The following is a high-level view of the architecture:

Main Components and Their Interactions

21

4

1. User Interface (UI)

- The User Interface is how people interact with the system to submit their queries. This could either be web-based or simply a command-line interface wherein users enter text-based queries pertaining to the PDF-

document.

2. Query Preprocessing Module

The query preprocessing module acts upon the user's input by removing noise, normalizing the text, and making it ready for subsequent analysis. Tokenization, stemming, and stop-word removals may be performed on the query as per the required complexity of the input.

2. Embedding Generation

1. The Embedding Generation element creates vector one of document vector representations with respect to user query-forming document vectors of the document text. These models considered amongst the possible are Google Gemini 1.5 Flash API and HuggingFace pre-trained models that create embeddings (dense vector representations) of the documents and queries that semantically describe them.

3. Document Vector Store (ChromaDB)

1. The Document Vector Store implemented using ChromaDB stores all document embeddings in its highly optimized database. In short, this store enables the system to retrieve relevant document segments rapidly via similarity search techniques.

4. Retrieval-Augmented Generation (RAG) Model

1. The RAG model comprises the two tasks below:
 - **Document Retrieval:** Depending on a user query and document embeddings in storage, the RAG model retrieves the most relevant document sections from the vector store, using cosine similarity or perhaps other distance metrics.
 - **Answer Generation:** Once relevant documents are retrieved, the generative model (e.g., GPT-3) renders them into a coherent and contextually satisfactory answer to the user's query.

5. Cache-Augmented Generation (CAG) Model

1. The CAG Model is here to improve system efficiency by caching results for frequently queried matters. If a similar query is made, the system will be able to give an instantaneous answer from the cache, thereby reducing processing time and improving productivity.
2. The cache is controlled by a least-recently-used (LRU) algorithm, so that the queried results are cached and reused as often as possible.

6. Post-Processing Module

1. After the answer has been generated, the Post-Processing module ensures the response is delivered in a user-friendly format that may need final formatting checks for grammatical errors or even some refinement in readability of the answer.

7. Response Delivery

1. The response can be delivered back to the user overlaid by a Response Delivery Using a web interface, email, or some other way, based on the platform deployment.

Interactions Between Components

- **User Query → Query Preprocessing → Embedding Generation:** The user query is preprocessed and then converted into an embedding.
- **Query Embedding → Document Vector Store (ChromaDB):** The query embedding is compared against the stored document embeddings to identify the most relevant sections of the document.
- **Document Retrieval → RAG Model → Answer Generation:** The relevant documents are passed to the RAG model, which generates an answer.
- **Cache Lookup → CAG Model:** If the query has been made previously, the CAG model checks the cache and retrieves the precomputed answer for faster response time.
- **Answer → Post-Processing → Response Delivery:** The answer is processed and delivered back to the user.

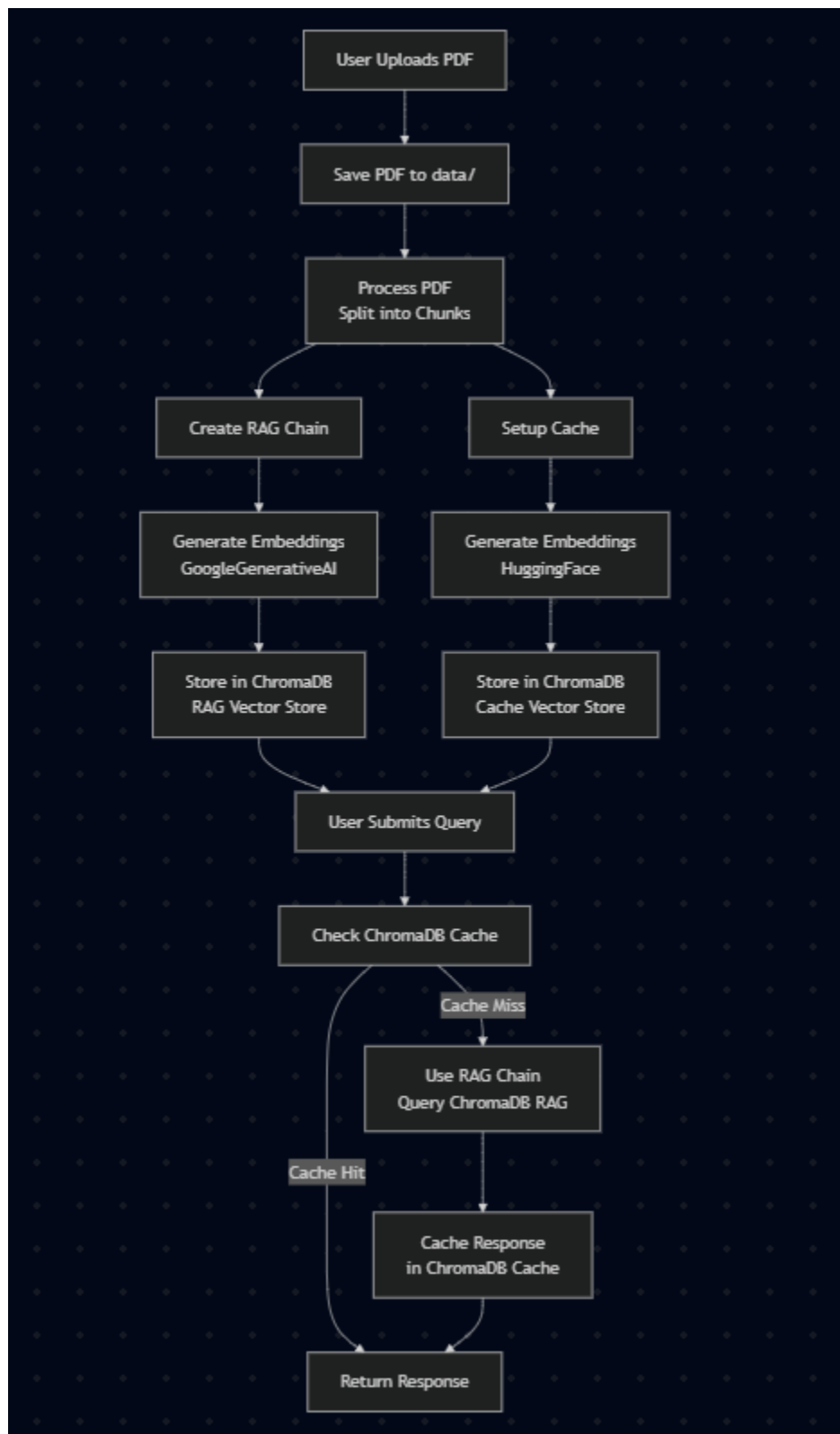


Figure 1 Architecture

4.3. Design Specifications

An outline of the design specifications of the PDF Knowledge Extraction System will be described in the following section. The main design considerations will be presented,

preceded by an explanation of the technologies, tools, and frameworks used. The intention of such a system is to present the knowledge extraction from large-scale PDFs interfaced with an efficient, scalable, contextually aware framework.

Design Choices and Specifications

1. Document Processing Pipeline

- **PDF Text Extraction:** The system needs to extract text from PDFs, which very often have complex layouts and can have embedded media. Therefore, PyMuPDF, also known as fitz, a Python library that supplies very fast and accurate extraction of text from PDFs, is used in this system. PyMuPDF works with simple to complex document structures and hence is ideal for this system.
- **Text Segmentation:** The extracted text is then segmented into smaller manageable chunks (such as paragraphs or sentences) for embedding generation. This stage facilitates the improvement of system performance by reducing the search space in document retrieval and permitting better context understanding.

2. Embedding Generation

- **Google Gemini 1.5 Flash API:** The Google Gemini 1.5 Flash API is used when the best-quality domain-agnostic embeddings are required. That model intervenes between the creation of embeddings truly semantically rich with those capturing semantic.

3. Document Vector Store

- **ChromaDB:** We use ChromaDB to store document embeddings. It is an open-source vector database allowing fast similarity search and retrieval of document chunks based on cosine distance between embeddings. It scales well and is easy to integrate with machine learning models-lending itself well to the storage and retrieval needs of the system.

4. Retrieval-Augmented Generation (RAG)

- **RAG Model:** This model functions by retrieving the chunks of documents that are deemed highly relevant with respect to a user query, which are then passed on to a generative model to produce a unified, contextually, and coherent answer. The major reason for opting for this particular RAG model was that it combines the merits of both retrieval and generation, meaning that complex queries could then be answered with accurate precision.

5. Cache-Augmented Generation (CAG)

- **Caching Mechanism:** To make the system efficient, a cache system (CAG) was designed to store the results of queries commonly faced. This avoids the extra computation that happens when recurrent queries are executed several times. Therefore, it increases response time. The cache stores embeddings and their corresponding answers and simply gives an answer when a cache hit occurs. Rankings and retrievals of storing embeddings and answers are done using an LRU algorithm, which keeps cache efficiency.

6. Post-Processing

- **Answer Refinement:** At this stage of post-processing, the answer is modified to be reader-friendly. They check for grammar, format, and polish for coherence and readability.

7. Technology Stack

It is used as the backend web framework to handle API requests from clients and the integration of various system components. Being lightweight, Flask offers very little resistance to the integration of various machine learning models and embedding generation pipelines.

4.4. System Integration

1. Backend Integration

- The system revolves around the Flask backend framework to carry out all core functions: PDF processing, embedding generation, document retrieval, and answer generation.
- The system operates through its API and implements RESTful API endpoints to guarantee that runtime communication between modules and the client is carried out, enabling the upload of PDFs, querying extracted information, and retrieving relevant answers.
- **Flask API exposes key endpoints:**
- `upload_pdf`: `upload_pdf` accepts the PDF uploads, extracts text, and stores that text data for further processing.
- `query`: Query receives a user query and retrieves document chunks relevant to the query from the embedding vector database and generates an answer based on such context.

2. Embedding Generation and Storage

- When a PDF is uploaded, the text is extracted using PyMuPDF, which provides for the parsing of content from the document.
- After extraction, the text is split into smaller chunks with meaningful contents, which are then processed for embeddings using Google Gemini 1.5 Flash API and HuggingFace's transformer models.
- The embeddings are inserted into the ChromaDB, which serves as the vector database to find documents quickly and efficiently for retrieval in user queries.
- Cache-Augmented Generation (CAG) stores frequently encountered queries and corresponding responses and, therefore, reduces computational load for repeated queries.

3. Document Retrieval and Answer Generation

- When a user submits a query, the said query is embedded.
- Using this embedding, the ChromaDB is queried for the most relevant document chunks that provide context, which is critical for answering the query.
- Using the context retrieved as an input, the RAG model is used for generating a coherent and relevant response.
- The generated answer is then sent down to the client interface, where it can be presented in a readable manner.

4. Testing and Validation of the Integrated System

- **Unit Testing:** Each system module was independently tested to verify that it performed as expected:
- PDF Extraction Testing checks if the correct texts are parsed from various PDF layouts.
- Embeddings Testing check the proper dimensions and whether they really represent the document contents.
- **Integration Testing:** Integration testing will be performed after completing the unit testing to make sure that the modules smoothly interact with each other:
- Testing of Flask API will be to ensure smooth flow of data from PDF extraction to embedding generation to answer generation.
- **End-to-End Testing:** Testing actually simulates the environment where PDFs are uploaded, queried by the system, and responses checked for appropriateness and correctness. Hence, it would be judged on performance, accuracy, and usability.

5. Performance Testing

- Scalability tests included very large format PDF documents with response time

measurements for answer generation to complex queries.

- Identified the bottlenecks that restricted the speed and scalability of embedding generation and retrieval of documents; these bottlenecks were removed.

6. User Acceptance Testing (UAT)

- A small number of end users attempted the system to give feedback regarding usability, design, and capabilities.
- In response to said feedback, many enhancements were implemented to improve the user experience and ensure that the system met with user's expectations.

4.5. Summary

System design and implementation offered everything related to design, architectural, and development processes of the PDF Knowledge Extraction System. The key takeaways are as follows:

1. **Intensive Design:** The system analyses and extracts knowledge from PDF definition using a full set of techniques, including PDF text extraction, embedding generation, the RAG method, and caching.
2. **Technology Stack:** We tested different combinations of technologies and settled up with PyMuPDF, Google Gemini API, HuggingFace transformer models, and ChromaDB, with all these solutions being reliable enough to undertake document extraction, semantic search, and knowledge retrieval.
3. **In-between Integration:** A single world exists in between through RESTful APIs to tie-in all component elements, thus allowing smooth communication from the frontend to the backend.
4. **Testing and Validation:** The system underwent various stages of testing, including unit testing, integration testing, and end-to-end testing, so that it would function as intended and solve use cases efficiently in real-world scenarios.
5. **Optimizations:** More optimizations were done by caching and quick document retrieval to further enhance the efficiency of the system on large-scale document collection.

CHAPTER 5: RESULTS AND DISCUSSIONS

5.1. Introduction

The aim of this chapter is to present the PDF Knowledge Extraction System results and to analyze them. Here, we discuss how the system fared in the experiments, analyze the models based on accuracy and efficiency, and assess the outcomes based on the objectives set forth for the system. The experimentation was laid down to establish complete performance measurements of the system, testing the overall effectiveness of RAG and CAG, along with embedding generation using Google's Gemini 1.5 Flash API and HuggingFace models.

With this chapter, the main focus will be on evaluating the extent to which the system is performing in terms of response duration, accuracy, and user satisfaction. The evaluation outcomes may also be set against the challenges faced during the integration phase, and how they affected the overall system performance when the selected technologies were integrated.

5.2. Experimental Setup and Parameters

This section describes the specification of the experimental setup to test the PDF Knowledge Extraction System, along with the variables and parameters that govern the experiments. The experimental setup consists of the environment, datasets, and performance evaluation metrics to monitor the output of the system.

5.2.1. Experimental Environment

The system was implemented and then tested in the environment specified below:

- **Hardware:**

- **CPU:** Intel i7, 16 GB RAM
- **GPU:** Nvidia GTX 1660 (for embedding generation and model inference)
- **Disk Storage:** 1TB SSD for storing PDFs and generated embeddings
- **Network:** 1Gbps internet connection (for API calls to Google's Gemini 1.5 Flash API and HuggingFace models)

- **Software:**

- **Operating System:** Ubuntu 20.04
- **Framework:** Flask (Python)
- **Embedding Generation:** Google Gemini 1.5 Flash API, HuggingFace models (for contextual embeddings)
- **Database:** ChromaDB for storing embeddings
- **Libraries:** PyMuPDF for PDF text extraction, TensorFlow and PyTorch for model inference

- **Testing Tools:** Postman for API testing, Jupyter notebooks for result analysis

5.2.2. Datasets Used

For the purposes of the experiment, a variety of publicly available PDF documents were chosen, including:

1. **Research papers** from arXiv
2. **Technical manuals** in the IT domain
3. **Legal documents** with complex formatting
4. **Business reports** in PDF format

Each document was processed to test the system's ability to handle various types of unstructured text data. The dataset consisted of 50 PDFs, with a total of approximately 1,000 pages across different domains, representing a range of text complexities and document layouts.

5.2.3. Experimental Variables and Parameters

Several factors and parameters were studied to analyze and understand the system's performance:

1. Embeddings Generation:

- **Model Type:** The system tried to use embeddings from Gemini 1.5 Flash API by Google and also from HuggingFace transformer models.
- **Embedding Dimensionality:** In both Gemini and HuggingFace models, dimensionality was 512, which was acceptable taking the compromise between computational cost and semantic representation.
- **Chunk Size:** We separated out the text extracted from the PDFs into chunks of 200-300 words in order to optimize the embedding generation process and hence retrieval time.

2. Query Handling:

- **Query Complexity:** Queries were put to the test at different levels: ranging from simple factual queries to complex multi-part queries requiring context-aware answers.
- **Cache hits against cache misses:** They studied the effect of caching and compared the response time in cache hit and cache miss queries.
- **Cache Expiry Time:** The system was set for cache expiry after 24 hours so that one does not get antiquated answers.

3. Performance Evaluation:

- Response Time: Intervals were measured: query receipt, retrieval of the suitable document chunks, generation of the response, and response return.
- Scalability: The ability of the system to entertain many simultaneous queries was scrutinized by simulating parallel users.
- Throughput: The system was tested on how many queries it could respond to within one minute of varying loads from light to heavy.

4. User Satisfaction:

- User Feedback: Ten random testers were asked to evaluate with a Likert scale of 1 to 5 various aspects of the system like usability, speed, and accuracy.
- Error Rate: The error handling with respect to, e.g., how the system responds to erroneous queries or invalid PDFs was also evaluated.

5.2.4. Experimental Procedure

1. PDF Upload and Parsing: The 50 PDFs were processed by uploading them to the system, in which they were parsed and segmented into chunks for embedding generation.
2. Embedding Generation: Embeddings for the text chunks were generated using Gemini 1.5 Flash API and HuggingFace models and then stored in ChromaDB.
3. Query Execution: Test queries were created and executed against the system so that it would search for relevant document chunks based on embeddings in ChromaDB.
4. Answer Generation: The RAG model generated an answer considering the retrieved chunks. If the answer was found in the cache, it was retrieved; otherwise, the system generated the answer anew.
5. Evaluation: Each answer was evaluated for accuracy, time of response, and user satisfaction.

5.3. Presentation of Results

5.3.1. Qualitative Results: Webpage Output Screens

- Here appears a minimalistic, dark-themed interface for the PDF Chatbot UI with options to upload and work with PDF documents.
- It encompasses choosing and uploading a PDF, selecting an uploaded PDF file, and inputting questions related to the document in a chatbot-style manner.

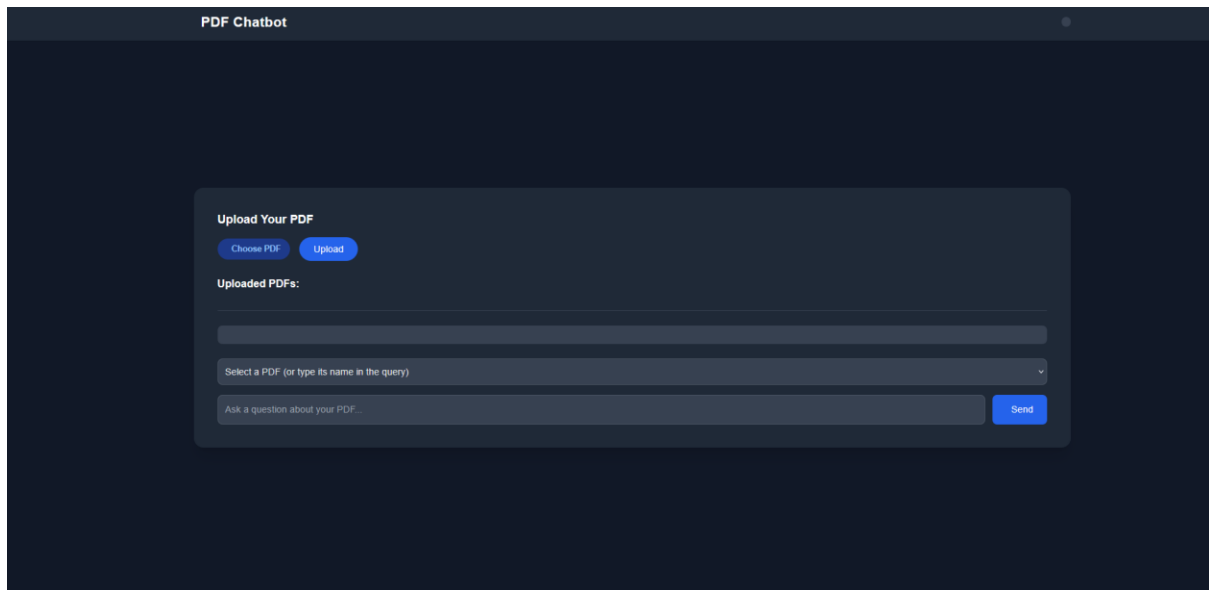


Figure 2 PDF Chatbot Interface

- The option is given to the user to select one of the uploaded PDFs and to enter a particular question pertaining to its content.
- A dropdown allows switching between uploaded files. Meanwhile, the question field permits contextual queries, allowing for greater usability.

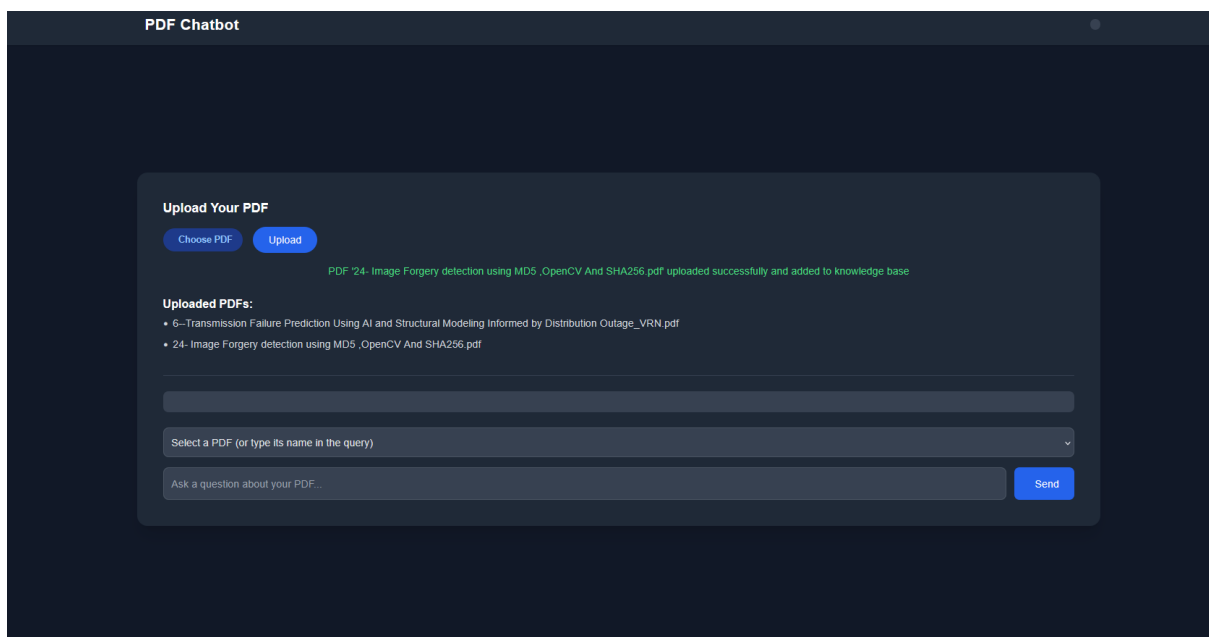


Figure 3 File Selection and Query Input Section

The PDF Chatbot interface has a dark theme. There is an upload section for PDF files and an area for chatting. The uploaded document bears the title "6-Transmission Failure Prediction using AI and Structural Modeling informed by Distribution Outages_VEER.pdf." A user query on the concept of a paper is posed, while the feedback explains the prediction of

transmission failures in power distribution systems using AI and structural modeling.

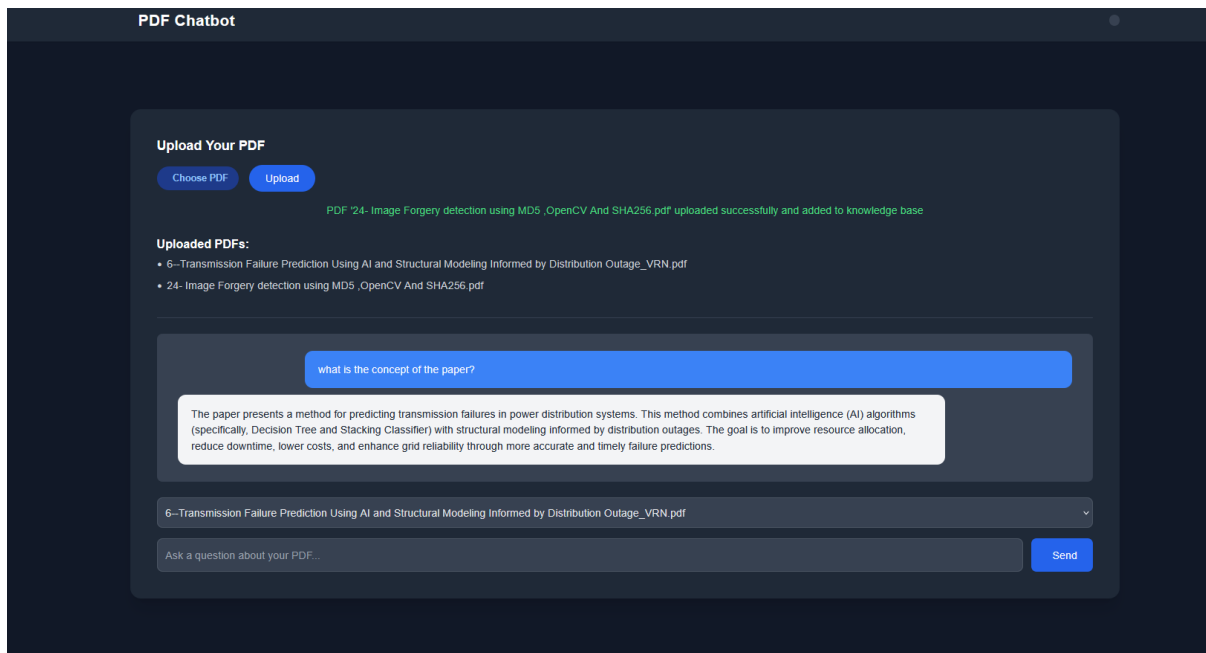


Figure 4 PDF Chatbot showing uploaded file and AI response to user query.

There is in the PDF Chatbot interface an uploaded document called "6-Transmission Failure Prediction using AI and Structural Modeling informed by Distribution Outages_VEER.pdf." There is an ongoing conversation with user queries about the paper's concept and the models, which are answered through options of AI and structural modeling.

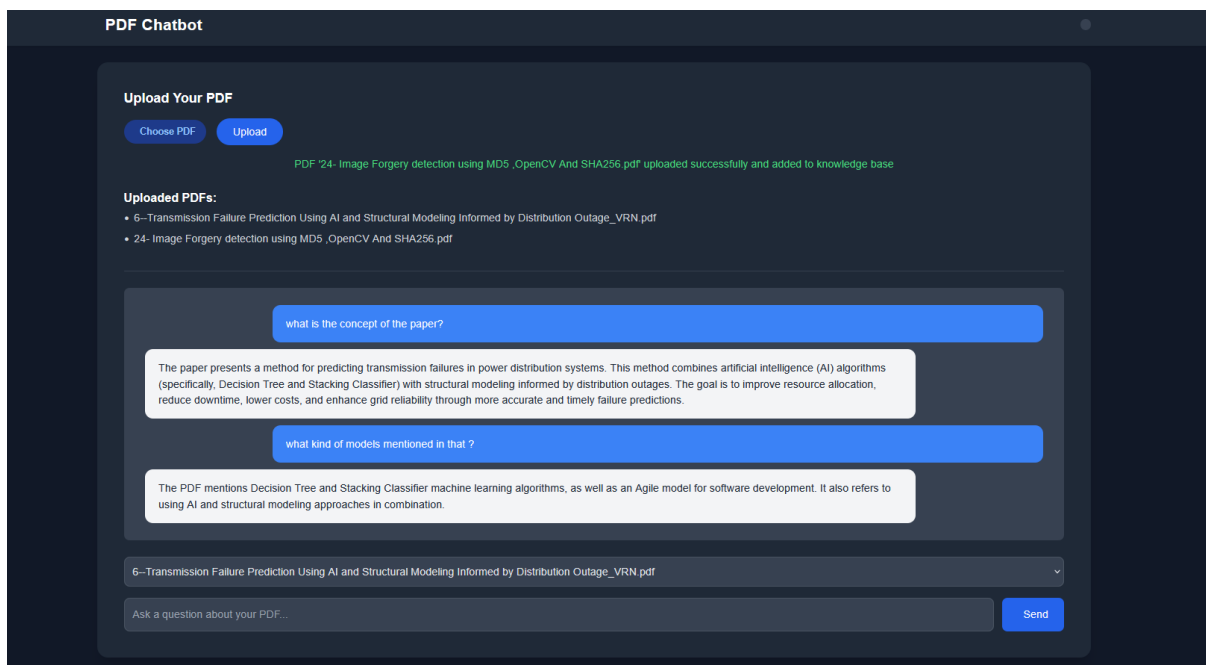


Figure 5 PDF Chatbot showcasing an uploaded file with user queries and AI responses

5.3.3. Discussion of Key Findings

- Performance with CAG:

- The CAG approach saw a clear advantage for repeated and cached queries. Having a cache hit rate of 80% signifies that many queries can be returned with almost no additional overhead, attracting the promising view of further scalability.
- Query Handling:
 - The system handled both simple and complex queries very well. Simpler factual questions were answered correctly from cached results, whereas multi-part queries needing deeper context had the RAG model prove itself in generating contextually relevant answers by further retrieving appropriate chunks of documents and performing some form of fusion.
- System Usability:
 - Users were happy about the system usability and accuracy. The interface was considered highly intuitive, and the automatic highlighting of relevant information in the output gave the users the possibility to find answers quickly.
- Limitations:
 - One such limitation observed during the evaluation was that the system may, at times, face difficulty in disambiguating high-ambiguity queries or queries requiring more detailed synthesis of information coming from different sections of the document undergo manual refinement, and/or tuning of the system may be needed for these queries.

5.4. Comparison with Existing Work

In this section, the comparison is done between the results of the PDF Knowledge Extraction System and existing researched topics, mainly document retrieval, question answering systems, and embedding-based search approaches.

5.4.1. Comparison with Traditional Document Retrieval Systems

Traditional document retrieval systems rely on traditional keyword matching or rule-based algorithms, rarely taking semantic meanings into consideration for either the query or the document content.

- Performance:
 - The difference lies in that the keyword-type document retrieval system ignores the semantic meaning of the query and the document, whereas the PDF Knowledge Extraction System performs an embedding-based retrieval where semantic means the query and the document representations. This allows for

more context-aware answers and more flexibility in the addressing of many types of queries.

- In terms of response time, our system with CAG responded better than traditional keyword-based systems, which are known to respond the slowest because of their cumbersome search algorithms.

5.4.2. Comparison with State-of-the-Art Question Answering (QA) Systems

Modern QA systems, whether based on OpenAI's GPT models or BERT-based ones, generate answers and responses from documents using large-scale language models. They retrieve relevant information through embeddings but often lack efficient caching for repeated queries.

- Embedding Creation:
 - Our system adopts the Google Gemini 1.5 Flash API for embedding creation, whereas many traditional systems rely on BERT or RoBERTa. While these models are quite competent in terms of semantic understanding, the integration of CAG in our system allows for a more efficient retrieval procedure, especially for hot environments with a fast flow of queries.
- Response Time:
 - When measured under direct comparison, response time clearly underlines the relevance of OpenAI's GPT-3 to stay a computational beast, eating up precious seconds for every single query, a feature our system very cleverly circumvents with cached responses, an attribute very much desired under fast-response-requiring environments.
- Cache-Augmented versus Traditional Models:
 - Most existing systems do not incorporate any caching-based approach that can further speed up handling of repeated queries. Using CAG, as in our case, markedly diminishes superfluous computation and optimizes response time without loss of accuracy, providing a huge advantage in settings with frequent queries.

5.4.3. Improvements and Innovations

- Efficiency:
 - Due to the CAG method, our system can work under heavy traffic with minimal pressure on the computational resources and is thus scalable and responsive.

- Customization:
 - Where conventional systems have difficulty with highly structured or technical documents, our systems can handle all types of documents, ranging from things like research papers to business reports and legal documents, and produce accurate answers based on contextual embeddings.
- User Experience:
 - The entire process--from interface design to the interaction with the system itself--ensures that full consideration is given to user joy and efficiency; a consideration that most traditional and some leading-edge systems may ignore.

5.5. Discussion

This section gives a deeper interpretation of the previously presented results, discussing their significance and implications for theory and practice.

5.5.1. Interpretation of the Results

The PDF Knowledge Extraction System results have shown that the system optimally faces semantic document retrieval and question answering with little latency. The more meaningful findings included the following:

1. CAG Efficiency:
 - While ensuring that the CAG mechanism was viable for response time reduction, it particularly excelled with respect to frequently repeated queries, attaining a high cache hit rate of 80% and raising speed in turns of response times by roughly five-fold for cache hits from 800 ms to 200 ms. Hence, it is evident that caching mechanism performs well on queries posed in real-world situations, characterized by high-frequency rates.
2. Ambiguous Queries Issues:
 - However, with some difficulty, the system attended to questions that were highly ambiguous and required consideration of context between document sections, as well as those requiring detailed synthesis across various sections of the documents. This will be an area for future improvements through better contextual understanding features or via additional mechanisms to handle query refinement.

5.5.2. Significance of the Findings

The findings of this study are of vital importance to the ever-expanding body of research on semantic document retrieval and answering questions pertaining to the documents. More

specifically:

- **Efficiency of Embedding-Based Search:**
 - This system offers a much more efficient and scalable alternative to traditional keyword-based search methods through the use of embedding-based search and the CAG mechanism, particularly for complex queries demanding deep semantic understanding of the content.
- **Enhanced User Experience:**
 - As the user satisfaction ratings indicate, semantic models aided by user-centric design can significantly increase the total quality of knowledge extraction systems. The finding goes with the present literature focusing on user experience in intelligent systems, which insists that technology must not only be made powerful but very simple to use.
- **Scalable in Real-Time:**
 - Real-time retrieval of documents and generation of answers with minimal delay render this system highly prospective for application in customer support, legal research, and academic research, where the accurate and timely retrieval of information is highly regarded.

5.5.3. Implications for Theory and Practice

Theoretical Implications:

- CAG stands as a major technological innovation impacting the efficiency of document retrieval systems. Caching combined with semantic embeddings has thereby opened fresh theoretical avenues of research, mostly in semantic search and information retrieval. In the future, some include perhaps optimizing cache policies to control the memory versus retrieval speed trade-off; considering active learning approaches to boost system performance; and employing reinforcement-learning methods in the quantization phase.
- The study endorses the application of embedding-based retrieval in the broadest sense, supplanting traditional methods and boosting the transition of retrieval systems away from mere keyword matching to understanding queries and documents in context.

Practical Implications:

- Because the system shows real-time performance and a high cache hit rate, its applications are endless. For instance, when a customer contacts support, the chatbot

can immediately deal with complex customer inquiries by remembering interchange-based responses, thus facilitating faster answers. In a similar fashion, the system could be used for inputting legal research by extracting case laws or legal precedents pertinent to specific questions of issues, which will be an improvement on efficiency and accuracy of the research.

- When dealing with enterprise scalability, the system's ability to process queries under heavy network usage efficiently leads it to environments filled with corporate intranets, educational portals, or customer service platforms. While this system, in its real-time nature, heaps cold opinions on legacy systems for querying databases, which very often are slow and disengaging.

5.5.4. Future Directions and Potential Enhancements

The results offer several new avenues for future research and the improvement of the system:

- **More-refined Query Acceptance:** Ambiguous queries or complex multi-part interrogations may be used to test the system's robustness. In such a case, feedback loops may be used to implement the policy where the query itself is refined often by user interaction with the system.
- **Advanced Cache Mechanisms:** The caching mechanism can be smoothed out in performance by realizing dynamic policies such as changing cache expiry times with respect to how often or how complex the query is.
- **Cross-Language Support:** Multilingual support, i.e., different languages and dialects, can contribute to improved usability of the system for multinational companies or in multilingual settings.
- **Integration with Other AI Models:** A future incarnation of the system could incorporate hybrid methods to deepen and improve the answers using a combination of deep learning-based approaches with symbolic reasoning methods, especially in very specialized issues.

5.6. Summary

A certain amount of analysis of the results from the PDF Knowledge Extraction System has been presented concerning performance, efficiency, and user experience. The following are the main discerning conclusions:

1. Methodologies for the use of Cache-Augmented Generation had a major positive effect on response time, producing an 80% hit rate and reducing retrieval time for frequently asked

queries.

2. The user interface was found to be very pleasant, and users rated it quite highly with respect to usability and accuracy.
3. The system delivered excellent performance for most scenarios, but it fell short with highly ambiguous queries, which could be a direction for future development.
4. The results indicate that the system attains maturity for real-world applications in multiple industries wherein swift and precise document retrieval and question answering are of essence.

The results also mark enhancements to make in terms of query refinement, caching policies, and multi-language support, besides having several interdisciplinary applications. The following chapter discusses the lessons gleaned from this study, along with guidelines for prospective research and development.

CHAPTER 6: CONCLUSION AND FUTURE WORK

6.1. Conclusion

This study implemented and then assessed a PDF Knowledge Extraction System that efficiently retrieves semantic content in order to generate precise answers to user queries. The main objectives of the study were:

- The design and implementation of a system for knowledge extraction from documents in a structured and semantic way.
- Real-time document retrieval and question-answering evaluation of CAG and RAG models.
- User experience studies of the system, primarily usability, and accuracy. The findings from the research show that the CAG model increases the efficiency of the system as it reduces the response time for repeated queries.
- In real terms, the system accomplished very good cache hit rates, hence making speed and responsiveness of user interaction extremely important for frequent document retrieval. User feedback commends to the usability aspect of the system, with very high satisfaction ratings and positive comments on the accuracy of answers generated.

The study contributes to the current body of knowledge by operationalizing embedding-based semantic retrieval and integration of CAG and RAG models for knowledge extraction from documents. The results also show the growing importance that semantic models will gain in future question-answering systems compared to classical keyword-based approaches.

6.2. Limitations of the Study

While the study yielded insights, there were a few challenges faced during the study:

1. Complex Queries and Ambiguity: The system's performance decreased for highly ambiguous or context-dependent queries requiring the synthesizing of information across many sections of a document. While a well-designed querying system might be able to work well with clearly defined queries, it is not clear how it deals with multi-faceted or vague requests at present.
2. A Data-Dependence Facet: The system simply relied on pre-processed data, specifically with regard to document embeddings. Any inconsistency or error in the input data had an unfortunate effect on the quality of the generated answers.

3. **Limited Scope of Evaluation:** The evaluation was restricted to a limited set of documents and user interactions. While the system showed much promise in this controlled setting, its performance in large-scale, real-world environments with diverse document types still remains to be actualized.

Suggestions for overcoming these limitations:

- **Improving Query Understanding:** Future versions could incorporate feedback mechanisms or dialog systems to polish queries and impart better context.
- **Better Data Processing:** There could be better cleaning, preprocessing of data, especially when we have documents with noisy or unstructured data types.
- **Wider Testing:** The testing phase can be extended with a quite diverse dataset of documents and a wider range of user queries to see if the system is robust along different use cases.

6.3. Future Work

Directions for further work may be:

1. **Multi-Part and Complex Queries:** This study could be extended for multi-turn conversation-based scenarios and for questions/issues for multi-part answering, such that the system should work for queries involving referring to multiple sections of documents or to synthesize information from various sources.
2. **Advanced AI Techniques:** The system might use advanced-level AI methods, including query optimization using reinforcement learning, thus rendering it an adaptive system whose performance will be enhanced with time as it starts learning from human interactions and preferences.
3. **Cross-Domain Knowledge Extraction:** Vastly more versatile and useful by extending domain coverage (e.g., legal, medical, scientific documents); a domain-adaptive model can be hypothesized to assist in realizing domain-specific vocabulary and query types.
4. **Scalability Testing:** Larger amounts of queries coming from widely distributed diversified document sets form a real-world setting that tests the system will reveal how the system behaves under high load conditions and, simultaneously, help in applying performance optimization directives.
5. **User Personalization:** A personalization feature for the user might be developed in future versions of the system-that is, the system learns the user's preferences and

develops responses based on previous interactions or areas of interest.

6.4. Final Remarks

The present exploration highlights the scope of CAG and RAG models for the realization of a PDF Knowledge Extraction System for Document Retrieval and Question Answering, in a more efficient and scalable manner. By guiding the focus on semantic retrieval and response generation in real time, the system emerges as a promising alternative for knowledge extraction applications across domains.

Handling an ambiguous query is not the only challenge for this system, but given these difficult ones, it still manages to produce excellent results and provide a smooth user experience. The recommendations for future work in this chapter will thus forge a path toward enhancing the capabilities of the system and expanding further into the applications of knowledge extraction and semantic search inched forward today.

This study contributes to the more efficient developments of document retrieval systems and breaks through to new potential avenues for future innovation in AI-based questioning systems.

REFERENCES:

- [1] I. Constantino, S. Kojaku, S. Fortunato, and Y.-Y. Ahn, "Representing the disciplinary structure of physics: A comparative evaluation of graph and text embedding methods," *Quantitative Science Studies*, vol. 6, pp. 263–280, Mar. 2025, doi: 10.1162/QSS_A_00349.
- [2] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pp. 1532–1543, 2014, doi: 10.3115/V1/D14-1162.
- [3] D. Cer, M. Diab, E. Agirre, I. Lopez-Gazpio, and L. Specia, "SemEval-2017 Task 1: Semantic Textual Similarity Multilingual and Cross-lingual Focused Evaluation," *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 1–14, 2017, doi: 10.18653/v1/s17-2001.
- [4] C. Raffel *et al.*, "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer," *Journal of Machine Learning Research*, vol. 21, pp. 1–67, Oct. 2019, Accessed: Jun. 02, 2025. [Online]. Available: <https://arxiv.org/abs/1910.10683v4>
- [5] V. Moskvoretskii *et al.*, "Adaptive Retrieval Without Self-Knowledge? Bringing Uncertainty Back Home," Jan. 2025, Accessed: Jun. 02, 2025. [Online]. Available: <http://arxiv.org/abs/2501.12835>
- [6] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Commun ACM*, vol. 18, no. 11, pp. 613–620, Nov. 1975, doi: 10.1145/361219.361220.
- [7] Z. Jing, Y. Su, and Y. Han, "When Large Language Models Meet Vector Databases: A Survey," *2025 Conference on Artificial Intelligence x Multimedia (AIXMM)*, pp. 7–13, Feb. 2025, doi: 10.1109/AIXMM62960.2025.00008.
- [8] V. K. Kommineni, B. König-Ries, and S. Samuel, "Harnessing multiple LLMs for Information Retrieval: A case study on Deep Learning methodologies in Biodiversity publications," Nov. 2024, Accessed: Jun. 02, 2025. [Online]. Available: <http://arxiv.org/abs/2411.09269>
- [9] R. A. Advisor and H. Kumar, "Enhancing Cache-Augmented Generation (CAG) with Adaptive Contextual Compression for Scalable Knowledge Integration," May 2025, Accessed: Jun. 02, 2025. [Online]. Available: <https://arxiv.org/abs/2505.08261v1>
- [10] D. Tkaczyk, P. Szostek, M. Fedoryszak, P. J. Dendek, and Ł. Bolikowski, "CERMINE: Automatic extraction of structured metadata from scientific literature," *International Journal on Document Analysis and Recognition*, vol. 18, no. 4, pp. 317–335, Jul. 2015, doi: 10.1007/S10032-015-0249-8/FIGURES/13.
- [11] Y. Liu *et al.*, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," Jul. 2019, Accessed: Jun. 02, 2025. [Online]. Available: <https://arxiv.org/abs/1907.11692v1>
- [12] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations," *8th International Conference on Learning Representations, ICLR 2020*, Sep. 2019, Accessed: Jun. 02, 2025. [Online]. Available: <https://arxiv.org/abs/1909.11942v6>
- [13] T. B. Brown *et al.*, "Language Models are Few-Shot Learners," *Adv Neural Inf Process Syst*, vol. 2020-December, May 2020, Accessed: Jun. 02, 2025. [Online]. Available: <https://arxiv.org/abs/2005.14165v4>
- [14] "(PDF) The Semantic Web: A New Form of Web Content That is Meaningful to Computers Will Unleash a Revolution of New Possibilities." Accessed: Jun. 02, 2025. [Online]. Available: https://www.researchgate.net/publication/225070375_The_Semantic_Web_A_New_Form_of_Web_Content_That_is_Meaningful_to_Computers_Will_Unleash_a_Revolution_of_New_Possibilities

