



AI Multimodal Document Processing System Overview

Ms. Khushboo Thakur¹, Ms. Vedashree Shembekar², Ms. Renuka Bhaskar³, Ms. Khushi Awghad⁴, Dr. Mrs. P. V. Dudhe⁵

^{1 2 3 4} Student, Information Technology Engineering, Prof. Ram Meghe Institute of Technology and Research, Badnera -Amravati., Maharashtra, India

⁵ Assistant Professor, Information Technology Engineering, Prof. Ram Meghe Institute of Technology and Research, Badnera -Amravati., Maharashtra, India

Student Email: kt409790@gmail.com

Abstract—

How to Cite this Article:

Thakur, K., Shembekar, V., Bhaskar, R. & Awghad, K. (2026). AI Multimodal Document Processing System Overview. International Journal of Creative and Open Research in Engineering and Management, <i>02</i>(9), 1-9.
<https://doi.org/10.55041/ijcope.v2i9.218>

License: This article is published under the terms of the Creative Commons Attribution 4.0 International License (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author(s) and the source are credited.

© The Author(s). Published by International Journal of Creative and Open Research in Engineering and Management.



<https://doi.org/10.55041/ijcope.v2i9.218>



With the increasing volume of digital information, people work with various document formats such as PDFs, scanned documents, text files, handwritten notes, images. Manually extracting useful information from these documents can be time-consuming and difficult. The *AI Multimodal Document Processing System* addresses this problem by providing an intelligent and automated approach to document processing and understanding.

It uses Optical Character Recognition (OCR), document parsing, and Artificial Intelligence to extract meaningful content from different document formats. Google Gemini AI is used to generate summaries, identify key entities such as names, organization, names, dates, and monetary values, and also create quizzes from the document content. It also provides a chat feature that allows users to ask questions and receive answers based on uploaded documents.

The system is developed using *Next.js*, *React*, *TypeScript*, *Node.js*, *Express.js*, *MongoDB Atlas*, *Tesseract.js*, and *Google Gemini AI*. *MongoDB* and *GridFS* are used to securely manage uploaded documents and processed information. Users can also export quizzes, chat conversations, and extracted information in various formats.

This particular system integrates document processing, summarization, information extraction, learning, and document-based interaction into a single platform, making it useful for students, professionals, researchers and other users handling large amounts of document-based information.

Keywords— Artificial Intelligence (AI), Document Processing, Optical Character Recognition (OCR), Natural Language Processing (NLP), Real-time Document Interaction



I. INTRODUCTION

Digital documents are now widely used in almost every field of work and study.

Students, teachers, researchers, businesses, and institutions deal with a wide range of files, including PDFs, scanned documents, images, reports, and text-based files. Manually finding and extracting the required information from such files can be time-consuming, especially when they contain a large amount of content.

The development of Artificial Intelligence (AI), Natural Language Processing (NLP), Optical Character Recognition (OCR), and Generative AI has introduced new methods to automate document analysis [1][3].

OCR is particularly useful for scanned files and images, as it converts the text inside them into a format that can be processed by software. AI-based models can then use this extracted data for further analysis and produce meaningful results. Although there are many file-processing tools available, most are designed for specific purposes.

A person might need one tool for text extraction, another for generating summaries, and a different software for asking questions about a document. Multiple tools in this way can make document-related tasks inconvenient and time-consuming.

The AI-Powered Document Processing System was developed to bring these features together in a single platform.

It allows users to upload PDFs, images, and text files and perform various operations on their content. These include generating summaries, identifying key entities, answering questions, creating quizzes, and interacting with documents through a chat-based interface [1][6].

This approach enables users to work with their documents without relying on multiple separate applications.

II. LITERATURE REVIEW

The improvement of AI-based file-processing programs has been supported with the aid of research in regions together with multimodal AI, data retrieval, natural language processing, OCR, and smart automation. These studies provide useful ideas for designing structures which can paintings with extraordinary form document content material.

Smith et al. Explored using multimodal records with Retrieval-Augmented Generation (RAG), showing how statistics apart from simple text may be integrated into retrieval-primarily based AI applications [1]. This technique is in particular beneficial when an AI system needs to reply questions primarily based on the content material of a specific record. Another place of associated studies is AI-based conversational systems. Studies on AI chatbots have tested how automated conversations can enhance interplay between users and software program structures [2]. This concept is applied within the proposed gadget thru the document-chat capability, which lets in customers to invite questions and get hold of responses based on the statistics to be had in their uploaded documents.

Multimodal Generative AI has additionally developed significantly in recent years. Google DeepMind delivered Gemini as a multimodal model capable of processing unique sorts of information, together with text and snap shots [3]. Google has further highlighted its capacity to aid responsibilities related to more than one styles of input [4]. Such abilities are relevant to record-processing packages due to the fact files can also incorporate both written information and visible content. Intelligent Document Processing (IDP) has



also been investigated as a manner of lowering guide effort in document-related sports. Ling et al. Provided technique that mixes Robotic Process Automation (RPA) with Machine Learning for file processing [5].

Their work demonstrates how automation may be used to deal with repetitive activities that might otherwise require significant guide attempt.

RAG was at the start proposed to improve question-answering structures with the aid of allowing a language version to retrieve relevant information before producing a response [6].

Along with file analysis, monitoring and evaluation are vital for AI-based totally programs. Phoenix gives tools for gazing and comparing AI structures, whilst Grafana can be used for tracking machine-related statistics [7][8].

In addition, CrewAI affords assist for growing AI-agent workflows, even as Qdrant and Chroma are associated with retrieval structures and vector-primarily based statistics garage [9][10][11].

OCR is some other important component whilst operating with scanned files and images. Research on PyTesseract has demonstrated how OCR may be used to apprehend and extract written content material from pictures [12]. Other studies have investigated deep-gaining knowledge of techniques for extracting facts from dependent files, consisting of formatted bills [13]. These approaches highlight the significance of choosing appropriate extraction strategies according to the sort of record being processed.

Research in Natural Language Processing has additionally contributed to the improvement of contemporary language-primarily based

programs. Devlin et al. Introduced BERT, a transformer-primarily based version designed to enhance the expertise of language and contextual relationships [14]. While the proposed system relies particularly on Generative AI for file analysis, work consisting of BERT represents an important development within the evolution of structures capable of expertise herbal language.

This technology displays the developing use of retrieval, agent-based totally processing, and observability in cutting-edge AI applications.

Overall, the reviewed research covers numerous individual areas required for wise document processing, which include OCR, multimodal analysis, retrieval, herbal language understanding, AI-primarily based interaction, storage, and machine monitoring.

III. RESEARCH GAP

Existing research and packages provide numerous useful solutions for running with virtual files however their capability is frequently divided throughout one-of-a-kind gear. For example, a person may additionally use one application to extract text, some other to summarize a file, and some other to interact with its contents through a chat interface. Moving between multiple applications can make the general report-processing workflow less handy.

The kind of record formats creates another realistic assignment. Some PDF documents include without delay selectable text and can consequently be processed without OCR. In comparison, scanned PDFs and picture-primarily based files do no longer provide textual content inside the same form and require additional recognition step before further analysis may be achieved. Research on smart document processing and OCR demonstrates the importance of coping with



those exceptional enter conditions appropriately [5][12][13].

A similarly requirement is to attach report processing with different beneficial features in preference to treating every operation separately. Users may additionally want to save their files securely, obtain AI-generated records, ask questions on the content, create quizzes, and keep the generated outcomes in usable codecs. Providing these functions via separate programs can make the workflow unnecessarily complex.

The AI-Powered Document Processing System is designed to deal with this gap by means of combining these sports within a single platform. The workflow starts with record add and content extraction, followed with the aid of AI-based totally processing and technology of useful consequences. By helping one-of-a-kind file types and presenting features together with summarization, entity extraction, question answering, record chat, quiz era, stable storage, and end result export, the device goals to offer an extra entire report-processing experience instead of focusing on most effective one assignment.

IV. METHODOLOGY

The proposed AI-Powered Document Processing System is designed as a series of processing degrees that rework an uploaded report into useful and comprehensible records. The workflow is designed to handle distinct record formats and follow the best processing approach at every stage.

The technique starts off evolved when the consumer uploads a PDF, photo, or textual content record. Before processing the document, the gadget performs the required validation after which shops the record securely. MongoDB Atlas is used for coping

with utility-related facts, while GridFS is used to deal with the uploaded files.

After garage, the machine determines how the file content material should be extracted. When a PDF already incorporates system-readable text, the content material may be acquired without delay. However, scanned PDFs and photograph-based totally files require Optical Character Recognition (OCR) to convert the seen text into virtual textual content that may be processed similarly [12][13].

The extracted content is then organized for evaluation. At this stage, the text can be cleaned and divided into smaller quantities when the record is big. Breaking a prolonged file into workable chunks facilitates the AI version paintings with the content greater successfully. For query-answering duties, Retrieval-Augmented Generation (RAG) may be used to discover relevant quantities of the file earlier than producing a reaction [1][6].

Once the content has been prepared, it is passed to the AI processing layer. Gemini is used to investigate the report and support features along with summarization, entity extraction, and query answering [3][4]. The processed records can then be used for other functions supplied via the machine.

The platform additionally allows users to generate quizzes from their uploaded files. Through the report-chat interface, users can ask questions about the content material and have interaction with the information in a conversational manner. This makes the system more useful than virtually extracting and displaying the textual content.



Finally, the processed facts offered to the person thru the software. The generated results can also be stored or exported in formats inclusive of PDF, JSON, CSV, and TXT, permitting customers to apply the information outside the platform when required.

Extraction/OCR → Preprocessing → AI Analysis → Chat/Quiz → Results → Export

V. SYSTEM ARCHITECTURE

The AI-Powered Document Processing System is organized the use of a purchaser-server architecture. Different components of the application, inclusive of the frontend, backend, database, report garage, OCR engine, and AI offerings, work together to system the files submitted by the user.

The most important additives of the device are described beneath.

5.1 Frontend

The consumer-facing a part of the software is constructed with Next.js, React, and TypeScript. It affords the displays thru which users can check in, log in, add documents, view their files, and get admission to the generated consequences. Features along with summaries, entity extraction, quizzes, and report-primarily based chat are also supplied via the frontend.

Communication among the frontend and server is treated via REST APIs the usage of Axios. Zustand is used for handling software state, such as facts associated with authentication and the currently logged-in user.

5.2 Backend

Node.js and Express.js shape the backend layer of the system. This layer handles maximum of the processing and coordinates communication among the user interface,

garage systems, OCR services, and AI offerings.

The backend manages sports which includes user authentication, record uploads, document processing, OCR, AI requests, quiz advent, record chat, database operations, and end result exports. Separate REST API endpoints are maintained for regions which includes authentication, documents, quizzes, chat, and exports.

5.3 Database and File Storage

MongoDB Atlas is used to save the utility's established records. This consists of user information, report metadata, extracted content material, summaries, recognized entities, generated quizzes, and chat messages.

The real uploaded files are maintained the use of MongoDB

GridFS. This allows the software to keep documents within the same database surroundings used for different application facts. GridFS handles huge documents by means of dividing them into smaller components, that can then be retrieved or streamed whilst required.

5.4 OCR and Document Extraction

The system makes use of unique extraction techniques depending at the type of file obtained. For PDFs that already contain device-readable textual content, the gadget uses pdf-parse to acquire the to be had content. When the input is an image or a scanned document, Tesseract.js performs Optical Character Recognition (OCR) to pick out the textual content.

This distinction is crucial because textual content cannot always be at once decided on from scanned pages or photographs. OCR makes it viable to transform such content into digital text that may be processed via the final additives of the device. Previous studies on text reputation and extraction also



demonstrates the usefulness of OCR for working with photo-based and structured documents [12][13].

5.5 AI Processing

The AI processing layer uses Google Gemini to work with the content material extracted from files. It supports several capabilities supplied through the application, together with precis technology, entity extraction, quiz advent, and query answering primarily based on record facts [3][4].

The use of Generative AI also lets in the machine to provide information past primary textual content extraction. By combining record content with multimodal AI competencies, the device can assist extra meaningful evaluation of different kinds of records [1][3].

For file-associated questions, the system follows the principle of using data from the uploaded file as the basis for generating responses. This is carefully related to Retrieval-Augmented Generation (RAG), in which relevant facts retrieved and then supplied to the language version before generating an answer [1][6]. This method helps the document-chat function stay connected to the available record content material.

5.6 Email and Authentication Services

Nodemailer, together with Gmail SMTP, is used to ship One-Time Password (OTP) verification emails. OTP-based verification is used for the duration of person registration and is also to be had as part of the password-reset technique.

Several safety mechanisms are protected in the software. Passwords and OTP information are included using bcrypt, while JSON Web Tokens (JWT) are used to manage authenticated access. The backend also applies request validation, price restricting, and secure

HTTP headers to provide additional safety for the APIs and person information.

5.7 Overall Request Flow

The interplay between the main additives can be represented as follows:

User → Frontend → Backend API → File Storage → Text Extraction/OCR → Gemini AI → MongoDB → Frontend

The workflow begins whilst a person selects and uploads a record thru the frontend. The backend receives the uploaded file the usage of Multer and shops it in GridFS. When document processing is asked, the backend retrieves the saved report and determines the best extraction technique.

For a textual content-based totally PDF, pdf-parse is used to gain the file textual content. If the report contains photo-based or scanned content material, Tesseract.js is used rather to carry out OCR. Once the text has been received, it is passed to the Gemini AI layer for tasks such as summarization and entity extraction.

The generated statistics is then stored in MongoDB in conjunction with the applicable report facts. Finally, the frontend retrieves the processed outcomes and gives them to the person via the appropriate sections of the software. This go with the flow connects document add, extraction, AI processing, storage, and consumer interplay into one continuous system.

VI. IMPLEMENTATION

The AI-Powered Document Processing System is constructed via connecting numerous technologies that deal with different parts of the software. The implementation includes the person interface, server-side processing, database and document management, OCR, Generative AI, authentication, and result exporting. Together, these additives support the entire workflow



from importing a report to generating and supplying beneficial information.

6.1 Frontend Implementation

The frontend is evolved the use of Next.js sixteen, React 19, TypeScript, and Tailwind CSS. It consists of the primary monitors required to use the utility, together with the landing page, login and registration pages, forgot-password page, dashboard, and character report workspace. The dashboard acts as the primary place for importing and dealing with files. Once processing is finished, the file workspace presentations the generated records in separate sections. Users can view the extracted text, summary, recognized entities, quizzes, and record-primarily based chat from the identical workspace.

The foremost frontend components encompass:

- Upload Modal – permits customers to select files or upload them via drag-and-drop.
- Chat Section – offers a communication interface for asking questions about a selected document.
- Quiz Section – shows the generated quiz and calculates the consumer's rating.
- General Chat Widget – presents get right of entry to a standard-purpose AI assistant.
- Markdown Render – presents AI-generated responses with suitable formatting.

Zustand is used to preserve worldwide utility information, especially authentication-related kingdom. Axios is answerable for sending requests between the frontend and backend REST APIs.

6.2 Backend Implementation

The server side of the software is evolved using Node.js and Express.js. It provides the APIs required for authentication, file processing, quiz operations, report chat, and exporting generated information.

During file add, Multer handles the incoming multipart shape information. The uploaded file is then placed in MongoDB GridFS, even as details related to the report are maintained within the Document series.

The backend is prepared into separate services for obligations consisting of Gemini verbal exchange, OCR processing, record handling, and e mail shipping. Dividing those duties into person services maintains the backend structured and makes future maintenance less complicated.

6. Three Document Upload and Processing

Document processing is completed through 3 most important levels.

1. Upload Phase

The user selects a PDF, photograph, or textual content record

from the dashboard. Multer gets the uploaded record and transfers it to GridFS for garage. At the equal time, data such as the filename, file type, file owner, and modern-day processing reputation is recorded in MongoDB.

2. Text Extraction Phase

When the user selects the Process alternative, the backend retrieves the saved report from GridFS and chooses an extraction technique consistent with its format:

- PDF → textual content is obtained the use of pdf-parse
- Image → text is diagnosed the use of Tesseract.js
- Text file → the present content is transformed into readable textual content.

After extraction, the ensuing text is stored with the corresponding record information. Using distinctive extraction techniques allows the application to handle both ordinary digital documents and files where the textual content exists best as part of picture. OCR is in



particular essential for scanned and photograph-based totally documents [12][13].

3. Processing Completion

Once the desired content has been extracted and saved, it turns into available for the following AI-based totally processing capabilities of the application.

6.Four AI-Based Processing

After the extraction level, the file content material is supplied to Google Gemini for evaluation. The AI layer helps the subsequent features:

- Document summarization
- Entity extraction
- Quiz technology
- Document-based totally question answering

When a file carries a big quantity of textual content, the content is separated into smaller chunks earlier than being furnished to the AI model. The outputs acquired from those quantities are then combined to supply the specified result. This technique permits the application to work with prolonged documents greater efficiently.

In short, the machine produces a shorter version of the document that highlights its major facts and vital factors. Entity extraction is used to discover applicable details along with names, corporations, dates, financial values, technical concepts, and different extensive records discovered inside the content.

Gemini changed into decided on due to its Generative AI and multimodal abilities [3][4]. The file query-answering technique is likewise linked with concepts mentioned in multimodal RAG and knowledge-intensive question answering research [1][6].

6.5 Quiz Generation

The utility can create quizzes automatically from the content material of an uploaded document. The user can specify the variety of questions required, and then the applicable file content is sent to Gemini for question technology.

The generated quiz can incorporate each a couple of-desire and short-answer questions. Along with every query, the device can provide the expected answer and an explanation. The generated facts are proven and stored in MongoDB earlier than being proven to the user.

Users can try the quiz via the frontend and receive their score after finishing it. This feature may be beneficial for students who want to revise material contained in notes, textbooks, or other study files.

6.6 Document Chat

The document-chat function presents a conversational way of working with an uploaded report. Instead of manually searching through the complete record, customers can ask questions about its contents via the chat interface.

When a question is submitted, the backend obtains the applicable file textual content together with latest messages from the communication. Gemini then generates a reaction the usage of the available file records. The consumer's question and the generated response are saved inside the Chat Message series, allowing the conversation to be persevered while required. This implementation follows the idea of grounded query answering, wherein the response is attached to applicable source facts. It is likewise related to the Retrieval-Augmented Generation technique discussed in preceding research [1][6].

6.7 Authentication and Security



Security measures are integrated into special elements of the software. User passwords are not saved directly; as an alternative, bcrypt is used to create password hashes. JWT tokens are used to authenticate requests to protected APIs.

During registration and password healing, customers can affirm their identification through email-based totally OTPs. The OTP values also are protected the use of bcrypt and are configured to expire mechanically after a described duration thru MongoDB's TTL functionality. Additional protection is supplied on the API level. Helmet is used for stable HTTP headers, Express Rate Limit enables control excessive requests, and Express Validator is used to validate incoming records. Access to documents, quizzes, and chat information is limited so that customers can paintings most effective with their own facts.

6.8 Export System

The machine affords alternatives for saving processed information in several normally used formats. The supported exports consist of:

- Quiz: PDF and JSON
- Chat Transcript: PDF and TXT
- Extracted Entities: JSON and CSV

PDF files are generated the usage of PDFKit, at the same time as JSON and CSV files are made out of the corresponding structured facts. Export APIs are protected by way of authentication so that generated facts can best be accessed with the aid of authorized users.

Overall, the implementation brings together report extraction, OCR, Generative AI, database and document storage, authentication, file-based conversations, quiz generation, and end result exporting inside a single software. This combination permits the device to automate numerous document-related sports at the same time as preserving

the whole workflow prepared for the person [1][3][5][6].

VII. RESULTS AND DISCUSSION

The AI-Powered Document Processing System became tested as a complete workflow, beginning from record add and finishing with AI-generated outcomes and export. The machine combines PDF parsing, OCR, Gemini AI, file storage, quiz technology, and file-based chat in one platform. This approach is constant with the use of multimodal processing and RAG-based document interplay mentioned in [1] and [6].

A. Document Processing Results

The machine successfully helps PDF, image, and text inputs. Digital PDFs are processed the use of PDF parsing, whilst scanned and picture-based totally documents are processed the use of Tesseract.Js OCR. This lets in the machine to address both device-readable and image-based files, much like the report textual content-popularity strategies mentioned in [12] and [13]. Extracted textual content is then saved for in addition processing.

B. AI Analysis Results

Google Gemini AI is used to generate summaries, essential factors, and based entities from the extracted content. For big documents, the text is split into smaller chunks before AI processing.

The use of generative AI for expertise and reworking record content material is supported via the abilities of multimodal models described in [3] and [4].

C. Quiz and Document Chat

The device can generate quizzes from uploaded documents, supporting customers revise and test their understanding. It additionally offers report-primarily based chat, where users can ask questions the usage of the uploaded record as context. This sort of



context-based question answering follows the fundamental idea of Retrieval-Augmented Generation discussed in [6].

D. Storage and Security

MongoDB Atlas stores person statistics, report metadata, entities, quizzes, and chat messages, whilst GridFS stores uploaded documents. The device additionally makes use of OTP verification, JWT authentication, password hashing, and protected routes to control get right of entry to user information.

E. Export and Discussion

The machine presents export alternatives for quizzes, chat conversations, and extracted entities in codecs inclusive of PDF, JSON, CSV, and TXT. Overall, the results display that OCR, file processing, generative AI, conversational interplay, and assessment era may be blended into one sensible platform. However, the contemporary challenge does not include a benchmark dataset or measured accuracy and reaction-time effects, so quantitative performance claims cannot be made yet.

Table I: Example Table Title

VIII. CONCLUSION AND FUTURE SCOPE

A. Conclusion

The AI-Powered Document Processing System presents a unified platform for processing and understanding unique

sorts of documents. It combines PDF extraction, OCR, and generative AI to transform files into beneficial records. OCR helps deal with scanned and photograph-primarily based files, as explored in in advance OCR studies [12], [13], whilst sensible record-processing methods assist automated facts extraction [5].

The device makes use of Google Gemini to generate summaries, extract entities, create

quizzes, and solution questions from uploaded files. Its document-primarily based question-answering method is related to Retrieval-Augmented Generation (RAG) [6], while its capacity to work with special forms of information follows the developing use of multimodal AI [1], [3]. Transformer-based totally language know-how, together with BERT [14], has additionally contributed to the improvement of contemporary AI-primarily based text-processing structures.

Overall, the undertaking combines file extraction, AI analysis, conversational interplay, garage, and export functions in one application. It affords a sensible foundation for packages in education, studies, and other regions in which customers need to understand large amounts of document content material efficaciously.

B. Future Scope

The system may be progressed with the aid of supporting greater report formats which include Word documents, spreadsheets, and presentations. Multilingual OCR also can be introduced to address documents written in extraordinary languages more efficaciously [12], [13].

Future variations can offer better expertise of tables, charts, pix, and complicated document layouts. More superior multimodal RAG techniques could improve document-based question answering [1], [6].

The device can also be evaluated the usage of popular datasets and metrics including OCR accuracy, answer relevance, summarization quality, and processing time. Monitoring equipment inclusive of Phoenix and Grafana could help analyze AI responses and gadget performance [7], [8].

Finally, cell aid, voice interaction, stronger privacy capabilities, domain-precise AI fashions, and advanced retrieval using



technologies together with Qdrant or Chroma [10], [11] could in addition increase the usefulness and scalability of the gadget.

REFERENCES

- [1] Smith et al., “Beyond Text: Optimizing RAG with Multimodal Inputs for Industrial Applications,” arXiv preprint arXiv:2410.21943, Oct. 2024.
- [2] M. Lee et al., “AI-Powered Chatbots: Enhancing Customer Engagement,” SciSpace, 2024.
- [3] Google DeepMind, “Gemini: A Family of Highly Capable Multimodal Models,” arXiv preprint arXiv:2312.11805, Dec. 2023. Available: <https://arxiv.org/abs/2312.11805>
- [4] Google, “Introducing Gemini: our largest and most capable AI model.”
- [5] Ling, X., Gao, M., Wang, D., “Intelligent document processing based on RPA and machine learning,” in 2020 Chinese Automation Congress (CAC), 2020, pp. 1349–1353. doi: 10.1109/CAC51589.2020.9326579.
- [6] P. Lewis et al., “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” in Proceedings of NeurIPS, 2020.
- [7] Arize AI, “Phoenix: AI Observability & Evaluation,” GitHub Repository. Available: <https://github.com/Arize-ai/phoenix>
- [8] Grafana Labs, “The Grafana Stack - Open-Source Observability Explained,” YouTube.
- [9] CrewAI, “Introduction to CrewAI.” Available: <https://docs.crewai.com/introduction>
- [10] Qdrant, “Agentic RAG With CrewAI & Qdrant Vector Database.”
- [11] Chroma, “LangChain Integration with Chroma.” Available: <https://python.langchain.com/docs/integrations/vectorstores/chroma/>
- [12] Saoji, S., Singh, R., Eqbal, A., Vidyapeeth, B., “TEXT RECOGNITION AND DETECTION FROM IMAGES USING PYTESSERACT,” 2021.
- [13] Geetha, M., Pooja, R., Swetha, J., Nivedha, N., Daniya, T., “Implementation of Text Recognition and Text Extraction on Formatted Bills using Deep Learning,” International Journal of Control, Automation and Systems, vol. 13, pp. 646–651, 2020.
- [14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 41