VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

(An Autonomous Institute Affiliated to University of Mumbai
Department of Computer Engineering)

Department of Computer Engineering

Project Report on

AutoSlides: Generative Presentation

Submitted in partial fulfillment of the requirements of Third Year (Semester–VI), Bachelor of Engineering Degree in Computer Engineering at the University of Mumbai Academic Year 2024-25

By

Harsh Ahuja (D12A/2)
Dhruv Chatrani (D12A/13)
Joel Dias (D12A/19)
Harsh Saindane (D12A/53)

Project Mentor

Mrs. Manisha Mathur

University of Mumbai

(AY 2024-25)

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

(An Autonomous Institute Affiliated to University of Mumbai
Department of Computer Engineering)

Department of Computer Engineering

VESIT
V.E.S.
Since 1962

CERTIFICATE

This is to certify that ____ of Third Year Computer Engineering studying under the University of Mumbai has satisfactorily presented the project on "---" as a part of the coursework of Mini Project 2B for Semester-VI under the guidance of ——------------------ -----. in the year 2024-25.

Date

Internal Examiner External Examiner

Project Mentor
Mrs. Manisha Mathur

Head of the Department
Dr. Mrs. Nupur Giri

Principal
Dr. J. M. Nair

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea / data / fact / source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

Harsh Ahuja (2)

(Signature)

Dhruv Chatrani (13)

(Signature)

Joel Dias (19)

(Signature)

Harsh Saindane (53)

Date:

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

It gives us immense pleasure to express our deep and sincere gratitude to Assistant Professor **Mrs. Manisha Mathur** for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We are deeply indebted to Head of the Computer Department **Dr.(Mrs.) Nupur Giri** and our Principal **Dr. (Mrs.) J.M. Nair ,** for giving us this valuable opportunity to do this project.

We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support and selfless help throughout the project work. It is a great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement several times.

Computer Engineering Department

COURSE OUTCOMES FOR T.E MINI PROJECT 2B

Learners will be to:-

CO No.	COURSE OUTCOME
CO1	Identify problems based on societal /research needs.
CO2	Apply Knowledge and skill to solve societal problems in a group.
CO3	Develop interpersonal skills to work as a member of a group or leader.
CO4	Draw the proper inferences from available results through theoretical/ experimental/simulations.
CO5	Analyze the impact of solutions in societal and environmental context for sustainable development.
CO6	Use standard norms of engineering practices
CO7	Excel in written and oral communication.
CO8	Demonstrate capabilities of self-learning in a group, which leads to lifelong learning.
CO9	Demonstrate project management principles during project work.

ABSTRACT

The AutoSlides project addresses a prevalent challenge in professional and academic environments: the time-intensive process of creating effective presentations from comprehensive documents. This report introduces AutoSlides, an innovative platform designed to streamline and enhance the presentation creation process by leveraging advanced technologies such as Large Language Models (LLMs) and sophisticated data formatting techniques.

AutoSlides aims to tackle common pain points faced by professionals across various sectors who struggle with distilling key information from lengthy documents into concise, visually impactful slides. The platform's primary objectives include improving efficiency, enhancing information retention, and facilitating clearer communication through optimized visual representations of complex data. This report outlines the project's motivation, stemming from extensive research on user needs and identified gaps in current presentation software. It provides a detailed exploration of AutoSlides' technical architecture, elucidating how LLMs are employed to comprehend and summarize document content.

The report concludes by outlining future development plans for AutoSlides. These include integration with popular presentation software, development of industry-specific customization features, and exploration of multilingual support to expand the platform's global applicability.

By offering a comprehensive solution to streamline presentation creation, AutoSlides positions itself as an innovative tool in the realm of information visualization and communication. It has the potential to transform how knowledge is shared and presented across diverse professional environments, making complex information more accessible and engaging for audiences.

Index

CHAPTER 1 : INTRODUCTION

1.1 Introduction

In professional and academic settings, creating effective presentations from detailed reports or documents is often a time-consuming and effort-intensive task. With the increasing complexity of data and the demand for efficient communication, individuals and organizations face challenges in condensing large volumes of content into concise, visually engaging slides.

To address this challenge, *AutoSlides* has been developed as a smart, AI-powered platform that leverages Large Language Models (LLMs) and advanced data formatting techniques to automate the generation of presentation slides from comprehensive documents. The goal is to streamline the process of transforming raw text into structured, visually coherent slide decks while preserving key information and enhancing clarity.

AutoSlides extracts core insights, organizes content, and formats it into slide-ready templates, making it ideal for students, educators, researchers, and professionals. It reduces manual effort, ensures consistency, and significantly cuts down the time needed to prepare impactful presentations. Furthermore, the platform's integration with existing document and presentation formats adds to its usability across various domains.

1.2 Motivation

The need for AutoSlides originates from common pain points faced by users in academia and industry. Creating presentations from reports often involves reading, summarizing, designing slides, and iterating through several revisions—activities that demand both time and design skills.

Surveys and user feedback indicate that professionals frequently find it difficult to balance design aesthetics with content accuracy, often resulting in either information overload or oversimplification. Additionally, traditional presentation tools offer little to no automation, leaving users to handle everything manually.

AutoSlides was conceptualized to bridge this gap—by enabling users to generate high-quality slides from raw content using natural language understanding and generation capabilities of LLMs. The aim is to simplify content distillation, improve user productivity, and support better communication of complex ideas.

1.3 Problem Definition

Despite the availability of modern presentation tools, most of them lack intelligent automation for content summarization and slide creation. Users must manually extract information, craft summaries, select visual layouts, and design slides—an inefficient and error-prone process.

Key challenges include:

- Manual summarization of long documents.
- Lack of content-design integration.
- No support for automated visual formatting.
- Limited scalability across different content types and domains.

AutoSlides addresses these issues by providing a centralized platform capable of understanding document semantics, extracting relevant information, and formatting it into professional-looking slides automatically.

1.4 Existing Systems

Current presentation tools such as Microsoft PowerPoint and Google Slides offer rich design capabilities but rely heavily on manual input. While some plugins and AI-assisted design tools exist, they mainly focus on design aesthetics and offer limited content intelligence.

Some LLM-based summarizers can process documents to generate summaries, but they do not provide end-to-end solutions for creating visual presentations. These tools often require significant post-processing, including slide structuring and formatting, which reintroduces manual workload.

AutoSlides stands apart by combining content understanding with visual formatting in a unified, automated workflow. It bridges the gap between summarization tools and slide design software to provide a holistic solution.

1.5 Lacuna of the Existing Systems

The shortcomings in current solutions include:

- No end-to-end automation from document to presentation.
- Dependence on user design skills and manual effort.
- Poor support for customizing outputs to specific use cases.
- Limited integration with existing document or learning management systems.

These gaps hinder productivity, especially for users with limited time or design expertise. AutoSlides overcomes these limitations through AI-driven content structuring, adaptive formatting, and support for customization, making it more inclusive and scalable.

1.6 Relevance of the Project

AutoSlides is highly relevant in today's fast-paced academic and professional environments, where time efficiency and effective communication are critical. By automating slide creation, it enhances the ability of users to focus on content quality and narrative flow rather than formatting.

Its integration of LLMs enables contextual understanding, ensuring that slides retain the core intent of the source document. Moreover, with support for future features like multilingual output, customization per domain, and export to popular platforms, AutoSlides is poised to become a valuable tool across education, business, and research sectors.

This project has the potential to transform how presentations are created and consumed, enabling knowledge sharing that is faster, clearer, and more impactful.

CHAPTER 2 : LITERATURE SURVEY

2.1 Survey of Existing Systems

The domain of automated presentation generation has witnessed notable advancements in recent years. This chapter reviews key research contributions that form the foundation for systems like AutoSlides. Each study addresses the challenge of converting textual documents into effective and visually coherent presentation slides using different methodologies.

2.1.1 Title of Paper: *Presentations by the Humans and For the Humans: Harnessing LLMs for Generating Persona-Aware Slides from Documents*
Published In: 2024
Abstract: This study presents an innovative framework leveraging Large Language Models (LLMs) to generate presentation slides tailored to the audience's expertise and time constraints. The system is trained on a large dataset of scientific papers and their corresponding slides, using supervised learning and human-in-the-loop feedback mechanisms.
Inference Drawn: This research makes a significant leap by integrating audience-awareness into slide generation. It highlights the effectiveness of LLMs in adapting content contextually. However, the system's reliance on LLMs and lack of active evaluation mechanisms indicate areas for improvement.

2.1.2 Title of Paper: *PPSGen: Learning to Generate Presentation Slides for Academic Papers*
Published In: 2023
Abstract: PPSGen is a system that automates slide generation using sentence scoring models and Integer Linear Programming (ILP) for content extraction and organization. It achieves superior performance over traditional systems.
Inference Drawn: The paper shows high potential for academic use, offering efficiency and customization. The use of ILP improves content relevance, but the approach may be limited by dependency on specific data formats and computational resources.

2.1.3 Title of Paper: *Presentations Are Not Always Linear! GNN Meets LLM for Document-to-Presentation Transformation with Attribution*
Published In: 2024
Abstract: This paper introduces GDP, a system combining document graph structures with Graph Neural Networks (GNNs) and LLMs for non-linear slide generation. It also explores GPT-based techniques and attribution mechanisms.
Inference Drawn: The integration of GNNs adds an innovative dimension to content structuring. However, the system's reliance on classifiers and limited multimodal processing remain as key constraints.

2.1.4 Title of Paper: *BIRAG: Basic Introduction to Retrieval-Augmented Generation*
Published In: 2024
Abstract: This work explores Retrieval-Augmented Generation (RAG) to enhance LLM output quality through external knowledge retrieval, improving accuracy and contextual relevance in generated content.
Inference Drawn: RAG proves beneficial in improving contextual understanding during slide generation. Nonetheless, increased model complexity and data preparation overhead may impact real-world deployment.

2.1.5 Title of Paper: *Learning-Based Slide Generator*
Published In: 2020
Abstract: The system utilizes BERT-based extraction for identifying key content from academic papers and organizing it into slide format.
Inference Drawn: This work lays a foundation for machine learning-based slide automation. While effective in content relevance, it offers limited customization and has significant technical dependencies.

2.2 Patent Search

An extensive review of patents was undertaken to understand existing innovations in AI-assisted presentation generation, document summarization, visual slide design, and retrieval-augmented content creation. The primary focus was on systems leveraging large language models, document layout analysis, and AI-based summarization techniques.

Particular attention was given to patents involving:

- Automated content extraction from academic and technical documents.
- LLM-based text summarization for slide-level granularity.
- Integration of user preferences (e.g., slide count, tone, target audience) into content generation.
- Visual design optimization tools that generate contextual imagery and layout formatting.

- Retrieval-Augmented Generation (RAG) methods to enhance factual accuracy and relevance of generated slides.

The findings indicate a growing trend in the automation of slide creation, especially in academic and corporate environments. However, a notable gap exists in systems that combine document format flexibility, multimodal processing (text + images), advanced summarization, and automated visual augmentation — a gap **AutoSlides** is designed to fill.

2.3 Inference Drawn

From the patent analysis and literature review, several critical insights have emerged that shape the core value proposition of **AutoSlides**:

- **LLM-driven systems** show promising results in slide generation, especially when combined with audience-awareness and structured content modeling (e.g., GNNs, ILP). However, they often lack adaptability to varied document formats and domain-specific content.

- **Content structuring approaches** like ILP or GNN-based non-linear sequencing improve coherence but struggle with scalability and multimodal data inclusion.

- **RAG techniques** enhance factuality and relevance, yet add complexity in data handling and architecture design.

- Most existing solutions are siloed, focusing either on slide text, layout design, or content relevance, without a unified workflow.

- There is **a clear lack of end-to-end platforms** capable of taking in complex documents (e.g., with tables, equations, figures), processing them intelligently, and generating visually coherent, context-aware, and audience-tailored presentations.

2.4 Comparison with the Existing System

Feature	Existing Systems	AutoSlides Enhancements
Document Input Flexibility	Limited to plain text or basic PDF parsing (Tome.ai, Gamma.ai)	Supports rich formats including research papers with tables, figures, and equations
Content Summarization	Basic summarization, often losing technical depth or context	Context-aware summarization using fine-tuned LLMs with structural preservation
Audience-Awareness	Rarely implemented or requires manual tuning	Incorporates tone, formality, and audience profile (e.g., expert vs general) into slide design
Visual Slide Design	Templates or minimal automation	AI-generated visuals and smart layout optimization based on slide content
Multimodal Processing	Very limited or non-existent	Integrates both text and image-based content, enabling richer slide generation
Technical Precision	Struggles with equations, tables, and domain-specific data	Maintains structural integrity and offers support for specialized academic and professional formats

CHAPTER 3: REQUIREMENT GATHERING FOR THE PROPOSED SYSTEM

3.1 Introduction to Requirement Gathering

Requirement gathering is a critical stage in the software development lifecycle that ensures alignment between the system’s functionality and the actual needs of its users. For the AutoSlides system, this phase focuses on understanding the challenges users face in manually creating presentations from extensive educational or professional documents, such as research papers, textbooks, and technical reports.

AutoSlides is engineered to bridge this gap using advanced Natural Language Processing (NLP) and machine learning techniques to automate slide generation. Unlike basic summarization tools, AutoSlides recognizes the unique structure and design requirements of presentations. It extracts key content, summarizes intelligently, integrates relevant images, and formats them into a coherent slide deck, reducing the time and effort required for slide creation.

This chapter outlines the functional and non-functional requirements of the AutoSlides system, including its software architecture, tools, technologies, and constraints influencing its development.

3.2 Functional Requirements

The AutoSlides platform incorporates various modules to automate and customize the slide generation process. The key functional requirements are:

- **Document Upload and User Preferences**
  Users should be able to upload documents in formats like PDF or DOCX and specify preferences such as the number of slides, layout styles, and whether to include images.

- **Text and Image Extraction**
  The system must accurately extract text (headings, paragraphs) and visuals (charts, diagrams) using tools like PyMuPDF, PDFPlumber, or OCR modules.

- **Content Preprocessing and Segmentation**
  Raw extracted text should be cleaned and segmented into components like titles, bullet points, and body paragraphs for structured slide formation.

- **Summarization Using NLP**
  The system should use models like BERT, GPT, or T5 to summarize content, identify

key ideas, and remove redundant information, generating slide-ready text.

- **Slide Structuring and Formatting**
  Slides should be auto-generated with organized headings, bullet points, and consistent formatting according to user-specified preferences.

- **Image Integration**
  Extracted images or contextually relevant visuals must be included to enhance the presentation, positioned accurately within corresponding slides.

- **Template and Layout Management**
  Users can choose from predefined templates or apply custom styles. The system ensures consistent styling throughout the slide deck.

3.3 Non-Functional Requirements

To ensure high performance and usability, the AutoSlides system must fulfill the following non-functional requirements:

- **Scalability**
  The platform should support multiple concurrent users and handle large documents efficiently.
- **Security**
  Uploaded documents and generated content must be handled securely, ensuring user privacy and preventing unauthorized access.

- **Availability**
  AutoSlides should be accessible at all times, providing uninterrupted service to users needing on-demand slide generation.
- **Usability**
  The user interface must be intuitive and responsive, making it easy for users to upload documents and customize presentations.
- **Performance Optimization**
  Slide generation should be fast and responsive, with minimal delays even for large or complex documents.
- **Cross-Platform Compatibility**
  The system should be accessible via web browsers across devices, ensuring a seamless experience regardless of platform.

3.4 Software, Technology, and Tools Utilized

To achieve its goals, the AutoSlides system uses a modern technology stack:

Frontend Development

- Framework: ReactJS
- Styling: Tailwind CSS or Material UI
- Libraries: react-dropzone for drag-and-drop file upload, axios for API communication

Backend Development

- Platform: Node.js
- Server Framework: Express.js
- File Parsing Libraries: PyMuPDF or PDFPlumber for text extraction from documents
- NLP Libraries: Hugging Face Transformers (BERT, T5, GPT)
- Image Processing: Tesseract OCR or similar for image and chart recognition

Machine Learning and NLP

- Models: Pre-trained transformer models for summarization and text classification
- Custom LLM Integration: Fine-tuned LLMs to improve slide structuring, bullet generation, and title formatting

Slide Generation

- Export Engine: Integration with libraries to export slides in PPT or PDF format
- Design and Layout: Templates and style engines to enforce consistent design aesthetics

Hosting and Deployment

- Cloud: Firebase for hosting or AWS/GCP for scalability
- Version Control: GitHub

3.5 Constraints

AutoSlides must address several constraints to ensure reliability and practical deployment:

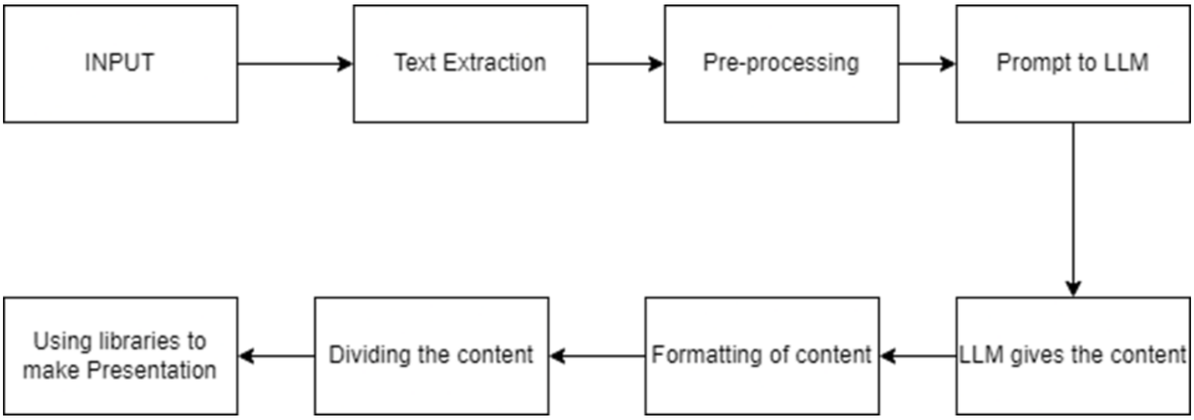
- **Document Complexity**
  Academic or technical documents may contain complex formatting, equations, or non-standard structures that require advanced parsing techniques.

- **NLP Limitations**
  Summarization models may occasionally miss context or prioritize less relevant information. Manual review and iterative improvement of models are needed.
- **Image Relevance**
  Automatically selected images may not always match the intended context, requiring fallback strategies or manual user input.
- **Export Limitations**
  Some slide elements (like animations or interactive components) may not translate well across all output formats (e.g., PPT vs. PDF).
- **Browser Compatibility**
  The web interface must function consistently across different browsers and screen resolutions.
- **Compute Resources**
  Running large NLP models can be resource-intensive. Efficient backend architecture and optional GPU acceleration are necessary for performance.
- **User Customization Complexity**
  While flexibility is a strength, too many customization options might overwhelm novice users. Preset templates and guided workflows should be provided.

CHAPTER 4: PROPOSED DESIGN

4.1 Block diagram of the system

Fig. AutoSlides System Architecture



4.2 Detailed Design

●**Input Acquisition:** Collect inputs from the user, including the PDF document to convert and the desired number of slides for the final presentation.

●**Text Extraction:** Extract raw text content from the PDF, capturing headings, paragraphs, tables, and other relevant information.

●**Pre-processing:** Clean and prepare the extracted text by removing unwanted characters and ensuring readability to make it suitable for LLM processing.

●**Prompting the LLM:** Construct a predefined prompt with details like the number of slides, the structured format required, and the cleaned text input for generating slide content.

●**Content Generation:** Use the LLM to generate content for the slides, including slide headers and bullet points, formatted as per the instructions in the prompt. Images are also generated with in this module.

●**Content Division:** Parse the generated content into individual slide headers and corresponding body text, organizing them into distinct sections, and keeping images as per standard format.

●**PPT Generation:** Create the final presentation by generating slide layouts, inserting the slide headers, images and content, and saving the file in PPT format for user review and hence downloading the presentation.

CHAPTER 5: IMPLEMENTATION OF THE PROPOSED SYSTEM

5.1 Methodology Employed

A. **Web Interface Development (React)**

The AutoSlides platform features a responsive web interface developed using React. It allows users to easily upload documents, set parameters (number of slides, design preferences), and preview generated slides in real time. The design emphasizes user-friendliness and rapid interaction.

B. **Backend Processing and LLM Integration (Python & Node.js)**

The backend is built with Python and Node.js to handle:

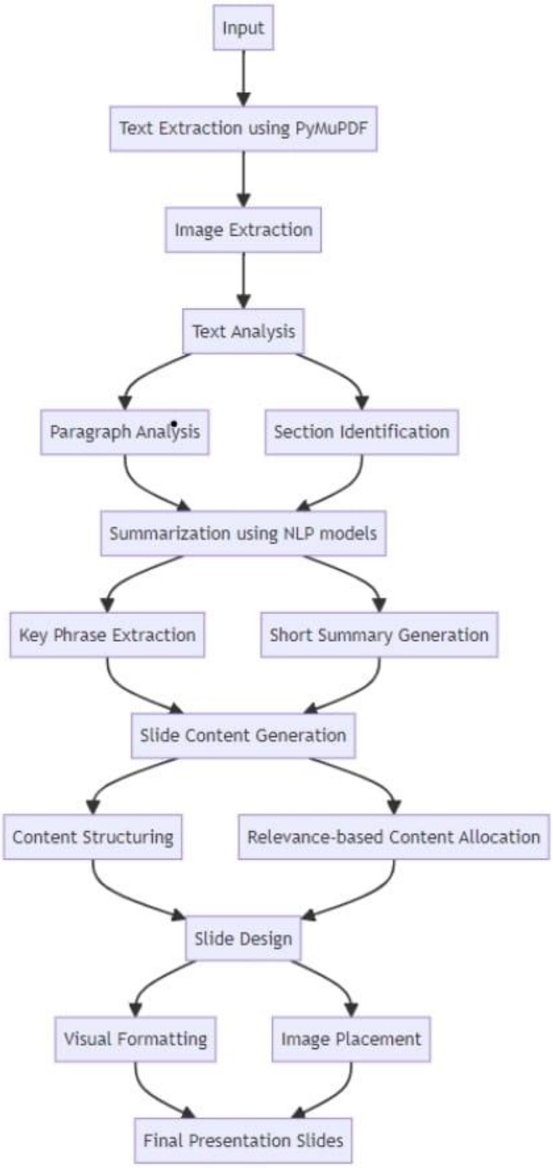
- Document parsing and text extraction using libraries such as PyMuPDF.
- Integration with a large language model (LLM) that generates concise slide content.
- Real-time communication with the frontend for progress updates.

C. **Slide Generation and Layout Formatting**

Once the content is generated, a dedicated module formats the slides according to predefined templates. It ensures the proper placement of text and images, adhering to visual consistency. This module also incorporates a dynamic image generation tool to create illustrations when required.

5.2 Algorithms and Flowcharts

Fig. 8 AutoSlides Slide Generation Flowchart



The flowchart outlines the following steps:

1. **Input Acquisition:** User uploads a document and selects presentation parameters.
2. **Text Extraction & Preprocessing:** The document is parsed and cleaned.
3. **Content Generation:** The LLM generates slide headers and bullet points.
4. **Slide Design:** The generated content is arranged into the slide template.
5. **Output Generation:** The final presentation is compiled and made available for download.

5.3 Dataset Description

The AutoSlides project uses a custom-curated dataset that includes:

- **Training Data:** A collection of academic and professional documents along with corresponding slide decks, used to fine-tune the LLM for presentation generation.
- **Test Data:** Manually selected PDFs and text documents representing diverse subject areas to evaluate the system’s performance.
- **Feedback Data:** User feedback on generated presentations is continuously collected to refine content accuracy and formatting quality.

CHAPTER 6: TESTING OF THE PROPOSED SYSTEM

6.1 Introduction to Testing

Testing ensures that AutoSlides meets the requirements of accuracy, usability, and reliability in generating high-quality presentations. This chapter describes the overall testing approach and the key metrics used to evaluate the system.

6.2 Types of Tests Considered

- **Functional Testing:** Verifies that each module (input, processing, output) performs as expected.
- **Performance Testing:** Assesses the system's speed in processing documents and generating slides.
- **Usability Testing:** Involves end-user trials to evaluate interface intuitiveness and overall user satisfaction.
- **Integration Testing:** Ensures seamless communication between the web interface, backend modules, and LLM integration.

6.3 Various Test Case Scenarios Considered

- **Case 1:** Uploading a multi-page academic PDF and verifying that the content is correctly segmented into slides.
- **Case 2:** Testing with documents containing images, tables, and complex formatting.
- **Case 3:** Stress testing by simulating multiple concurrent user requests.
- **Case 4:** Validating the accuracy of slide content and consistency of slide design across different document types.

6.4 Inference Drawn from the Test Cases

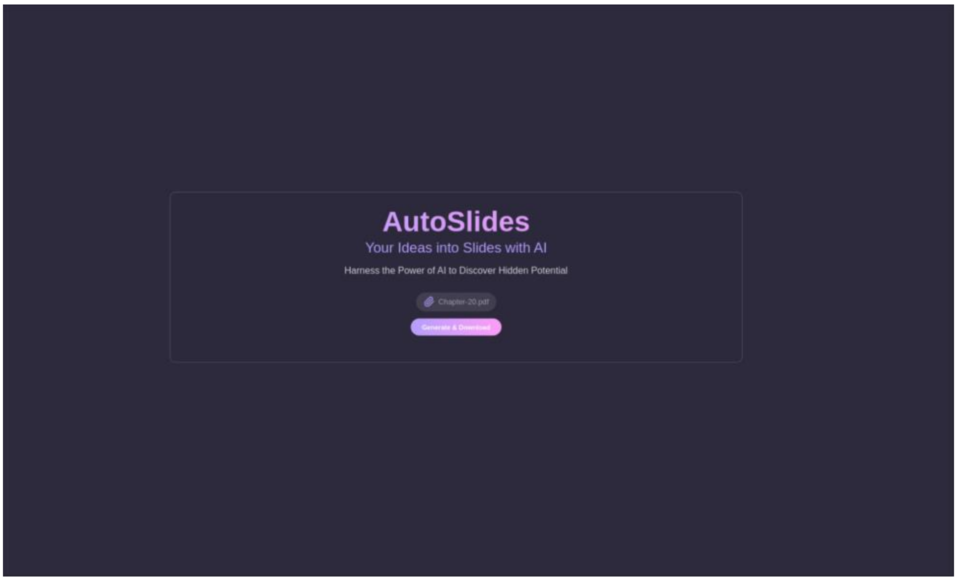
Testing revealed that AutoSlides:

- Consistently generates presentations that accurately reflect the key points from the source documents.
- Maintains a uniform design layout even when processing varied document formats.
- Meets performance benchmarks with acceptable processing times under peak loads.
- Benefits from iterative improvements driven by user feedback, particularly in formatting and content accuracy.

CHAPTER 7: RESULTS AND DISCUSSION

7.1 Screenshots of User Interface (GUI)

Fig. 9 AutoSlides Homepage & Document Upload Interface

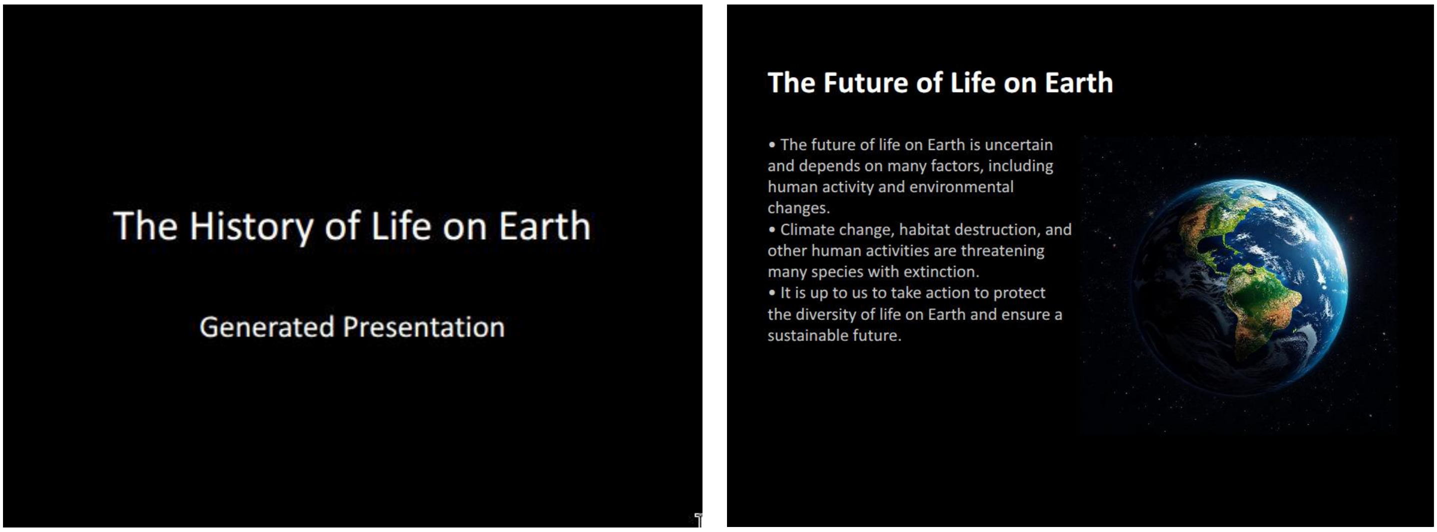


AutoSlides

Your Ideas into Slides with AI

Harness the Power of AI to Discover Hidden Potential

Fig. 10 Slide Preview and Editing Screen



The History of Life on Earth

Generated Presentation

The Future of Life on Earth

• The future of life on Earth is uncertain and depends on many factors, including human activity and environmental changes.
• Climate change, habitat destruction, and other human activities are threatening many species with extinction.
• It is up to us to take action to protect the diversity of life on Earth and ensure a sustainable future.

7.2 Performance Evaluation Measures

- **Processing Time:** Average time from document upload to presentation generation.
- **Content Accuracy:** Percentage of key points correctly extracted and formatted.
- **User Satisfaction:** Feedback scores from beta testers on ease of use and presentation quality.

7.3 Input Parameters / Features Considered

User-Provided Inputs:

- **Document Upload:** Accepts PDFs and text files.
- **Number of Slides:** User-defined slide count to balance detail and brevity.
- **Design Preferences:** Theme selection and layout style.

System-Generated Inputs:

- **Extracted Key Points:** Derived using NLP techniques.
- **Formatted Slide Content:** Automatically organized into headers, bullet points, and visuals.
- **Real-Time Preview:** Dynamic generation of slides allowing users to make edits before finalizing.

7.4 Graphical and Statistical Output

The system generates statistical data on:

- **Extraction Accuracy:** Graphs showing precision and recall of key point extraction.
- **Generation Speed:** Performance charts comparing processing times across various document lengths.
- **User Interaction:** Usage statistics and feedback distribution to guide future enhancements.

7.5 Comparison of Results with Existing Systems

AutoSlides stands out by:

- Offering a fully automated and highly customizable slide-generation process.
- Integrating state-of-the-art LLM technology to ensure accurate summarization and presentation.
- Providing a user-friendly interface with real-time feedback, compared to more rigid and manual systems.
- Delivering superior visual consistency and design aesthetics through dynamic template formatting.

7.6 Inference Drawn

The evaluation confirms that AutoSlides successfully streamlines the presentation creation process by:

- Reducing manual effort and time in slide generation.
- Delivering high-quality presentations that retain essential content and design integrity.
- Outperforming traditional methods in both speed and user satisfaction while paving the way for further enhancements.

CHAPTER 8: CONCLUSION

8.1 Limitations

While AutoSlides represents a significant advancement, it faces several challenges:

- **Formatting Complexities:** Documents with highly complex layouts or unusual formatting may result in less-than-perfect slide generation.
- **Content Nuance:** The system may sometimes oversimplify detailed content, requiring manual edits for specialized presentations.
- **Scalability:** Processing very large documents or handling multiple simultaneous requests may affect performance.
- **Customization Limits:** Current design templates may need further refinement to meet all user aesthetic preferences.

8.2 Conclusion

AutoSlides effectively leverages advanced NLP and LLM technologies to transform documents into professional presentations. By automating the extraction, summarization, and layout processes, it saves significant time and effort while ensuring content accuracy. The project has demonstrated the potential for intelligent presentation generation, making it a valuable tool for educators, professionals, and students alike.

8.3 Future Scope

Future work on AutoSlides will focus on:

- **Enhanced Customization:** Expanding template options and allowing deeper user control over design elements.
- **Advanced Formatting:** Integrating AI-driven adjustments for complex document structures.
- **Broader File Support:** Extending compatibility to include more file types such as Word documents and spreadsheets.
- **User Feedback Integration:** Continuously refining the system based on real-world user data and feedback.
- **Cloud Scalability:** Optimizing for higher loads and real-time collaboration features.

References

[1] A. Shreewastav, B. Acharya, N. Paudel, and Y. Humagain, “Presentify: Automated Presentation Slide Generation from Research Papers using NLP and Deep Learning (May 2024),” Authorea Preprints, May 2024, doi: 10.36227/techrxiv.171561007.70987286.

[2] K. Shaj, S. S. John, R. Philip, and A. George, “Learning Based Slide Generator,” Int. J. Eng. Res. Technol., vol. 9, no. 7, Jul. 2020, doi: 10.17577/IJERTV9IS070415.

[3] M. Sravanthi, C. R. Chowdary, and P. S. Kumar, “Slidesgen: Automatic generation of presentation slides for a technical paper using summarization,” in Proc. 22nd Int. FLAIRS Conf., 2009, pp. 284–289.

[4] D. Kottachchi and T. N. D. S. Ginige, “Slide hatch: Smart slide generator,” in Proc. 2nd Global Conf. Adv. Technol. (GCAT), 2021, pp. 1–5.

[5]W. Chen, H. Hu, X. Chen, P. Verga, and W. W. Cohen, “Murag: Multimodal retrieval-augmented generator for open question answering over images and text,” arXiv preprint arXiv:2210.02928, 2022.

[6] S. Thomas, V. G. John, J. Chacko, M. Shajahan, and S. Sunny, “PPT Generation from Report,” Int. J. Emerging Res. Areas, vol. 3, no. 1, 2023, doi: 10.5281/zenodo.8012970.

[7] T.-J. Fu, W. Y. Wang, D. McDuff, and Y. Song, “Doc2ppt: Automatic presentation slides generation from scientific documents,” in Proc. AAAI Conf. Artif. Intell., vol. 36, no. 1, 2022, pp. 634–642.

[8] Y. Hu and X. Wan, “PPSGen: Learning-based presentation slides generation for academic papers,” IEEE Trans. Knowl. Data Eng., vol. 27, no. 4, pp. 1085–1097, 2014.

Citations

1. Ge J, Wang ZZ, Zhou X, et al. AutoPresent: Designing Structured Visuals from Scratch. arXiv. Published January 1, 2025. Accessed April 27, 2025. https://arxiv.org/abs/2501.00912

2. Kumar K, Chowdary R. SlideSpawn: An Automatic Slides Generation System for Research Publications. arXiv. Published November 20, 2024. Accessed April 27, 2025. https://arxiv.org/abs/2411.17719

3. Fu TJ, Wang WY, McDuff D, Song Y. DOC2PPT: Automatic Presentation Slides Generation from Scientific Documents. In: Proceedings of the AAAI Conference on Artificial Intelligence. 2022;36(1):1003-1011. Accessed April 27, 2025. https://arxiv.org/abs/2101.11796

4. Xie E, Xiong G, Yang H, et al. Leveraging Grounded Large Language Models to Automate Educational Presentation Generation. In: Proceedings of Large Foundation Models for Educational Assessment. Proceedings of Machine Learning Research. 2025;264:207-220. Accessed April 27, 2025. https://proceedings.mlr.press/v264/xie25a.html

5. Gupta T. Automatic Presentation Slide Generation Using LLMs. Master’s Thesis. San José State University; 2023. Accessed April 27, 2025. https://scholarworks.sjsu.edu/etd_theses/5444/

6. Shreewastav A, Acharya B, Paudel N, Humagain Y. Presentify: Automated Presentation Slide Generation from Research Papers using NLP and Deep Learning. TechRxiv. Published May 2024. Accessed April 27, 2025. https://www.techrxiv.org/doi/full/10.36227/techrxiv.171561007.70987286/v1

7. Geng M, Chen C, Wu Y, Chen D, Wan Y, Zhou P. The Impact of Large Language Models in Academia: from Writing to Speaking. arXiv. Published October 22, 2024. Accessed April 27, 2025. https://arxiv.org/abs/2409.13686

8. Luo Y, Li Y, Ogunyemi O, Koski E, Himes B. Leveraging Large Language Models for Academic Conference Organization. npj Digital Medicine. 2025;8(1):Article 149. Accessed April 27, 2025. https://www.nature.com/articles/s41746-025-01492-7

Appendix

a. List of Figures

b. List of tables

c. Paper Publications :-

1. **Draft of the paper published. (Paper/MP)**
2. **Plagiarism report of the paper published /draft**
3. **Certificate of the paper publication**
4. **Xerox of project review sheets**