

---

# CAPSTONE PROJECT

## Research Agentic Model

**Presented By:**  
**Himanshu Singh**  
**Shree Guru Gobind Singh Tricentenary University**  
**Department of Computer Science & Engineering**

# OUTLINE

- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm & Deployment
- Result (Output Image)
- Conclusion
- Future Scope

---

# Problem Statement

Researchers often struggle with:

- Navigating large volumes of academic literature
- Extracting meaningful insights from complex papers
- Generating structured summaries, citations, and content
- Lack of unified tools integrating reading, understanding, and content generation
- These challenges hinder productivity, innovation, and learning—especially for students and early-stage researchers.

# Proposed Solution

The proposed system aims to address the challenge faced by researchers in quickly understanding, summarizing, and reusing academic content for research writing. This involves leveraging Large Language Models (LLMs) and agentic planning to automate critical parts of the research process. The solution will consist of the following components:

## Data Collection:

- Allow users to upload academic papers (PDF format) containing abstracts or full text.
- Store uploaded documents securely in IBM Cloud Object Storage.

## Data Preprocessing:

- Extract text from PDF files using PyPDF2 for further processing.
- Sanitize input to remove formatting issues or non-relevant characters.
- Identify key sections like title, abstract, and metadata using rule-based extraction.

## LLM-based Processing:

- Use IBM Granite LLMs via Watsonx.ai Prompt Lab to perform multiple tasks:
  - Summarize the research text into 5–7 structured bullet points
  - Generate citations in APA, MLA, and IEEE formats
  - Draft sections like Introduction and Literature Review
- Use predefined prompt templates for consistent output and accuracy.

## Deployment:

- Build a web-based application using Streamlit for interactive UI.
- Deploy the app on Lightning AI Studio for fast, GPU-enabled inference.
- Integrate backend with IBM Cloud Object Storage and Watsonx.ai for seamless data flow.

## Evaluation:

- Test the system across various research papers and academic fields.
- Evaluate generated outputs for accuracy, formatting, and relevance based on user feedback.
- Perform continuous prompt refinement and implement prompt chaining or RAG in future iterations.

## Result:

- Users can upload PDFs and instantly receive:
  - Bullet-point summaries
  - Formatted citations
  - Auto-generated research section content
- Reduces manual effort and accelerates the research writing workflow.
- Offers a lightweight, no-login, cloud-powered research assistant tool.

---

# System Approach

## Technologies Used:

- **Frontend: Streamlit (Python)**
- **Backend: IBM Watsonx.ai (Granite LLMs)**
- **Storage: IBM Cloud Object Storage**
- **Orchestration: Agentic planning for reasoning tasks**
- **Deployment Platform: Lightning AI Studio (for GPU access)**
- **Libraries: requests, streamlit, PyPDF2, ibm\_boto3, dotenv, logging, traceback**

# Algorithm & Deployment

- Model: granite-3-3-8B LLMs from IBM (Prompt-driven)
- Input: PDF Abstract → Extracted text → Prompt Template
- Prompt Template Tasks:
  - Bullet point summary
  - Generate citations in 3 formats
  - Generate structured research section content
- Deployment:
  - RESTful interaction with Watsonx Prompt Lab
  - Frontend deployed via Streamlit and Lightning AI Studio
  - Object Storage API for file handling

# Result

- Successfully processes academic PDFs and outputs:
- Bullet-point summaries
- Correctly formatted citations
- Introduction/Lit Review content
- Provides real-time responses through web UI
- Modular design with easy update paths

The screenshot shows the IBM Agentic Research Assistant web interface. On the left is a sidebar with options: 'Upload Academic Paper (PDF/Text)' with a 'Browse files' button, 'Enable streaming responses' (unchecked), 'Rate Limit Information' (dropdown), and 'Clear Cache'. The main area is titled 'IBM Agentic Research Assistant' and has tabs for 'Research Prompt', 'Summary', 'Hypotheses', and 'Section Draft'. Below the tabs is a section 'Enter Research Input' with a 'Try Example' button and a text area for 'Academic Text or Abstract'. Below this are input fields for 'Title', 'Author(s)', 'Journal/Source', and 'Year'.

This screenshot shows the 'Section Draft' tab of the IBM Agentic Research Assistant. The sidebar is identical to the previous screenshot. The main area has a 'DOI/URL' field, a 'Research Question' text area, a 'Section' dropdown menu set to 'Introduction', and a 'Topic for Section Draft' text area. Below these is an 'API Status' section showing 'Rate Limit Status' as 'Normal' (with an 'Available' indicator) and 'Requests (Last Minute)' as '0' (with '5 remaining'). A 'Generate Output' button is located below the API status. At the bottom, there are 'Tips to avoid rate limits' listed as bullet points: 'Space out your requests (wait a minute between submissions)', 'Use the example first to ensure everything works', 'Keep your abstracts concise for faster processing', and 'Save your outputs to avoid repeating requests'.

---

# Conclusion

- The research assistant reduces effort in academic writing and research
- Empowers users to:
- Understand complex papers faster
- Get correctly formatted citations instantly
- Generate research content through automation
- Modular, cloud-based architecture ensures scalability and flexibility



# Future scope

- Add full paper analysis using LangChain + vector DB for RAG
- Integrate full-fledged chatbot for interactive Q&A over PDFs
- Add plagiarism checker & grammar enhancer
- Enable user-uploaded prompt templates
- Expand support for multilingual papers and journals

# IBM Certifications



# IBM Certifications

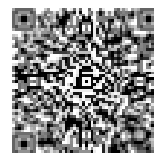
In recognition of the commitment to achieve  
professional excellence



## Himanshu Singh

Has successfully satisfied the requirements for:

### Journey to Cloud: Envisioning Your Solution



Issued on: Jul 20, 2025  
Issued by: IBM SkillsBuild

Verify: <https://www.credly.com/badges/1b9ee547-2b03-4cf6-9b5e-c48415814b7f>



# IBM Certifications

**IBM SkillsBuild**

Completion Certificate



This certificate is presented to

Himanshu Singh

for the completion of

**Lab: Retrieval Augmented Generation with  
LangChain**

(ALM-COURSE\_3824998)

According to the Adobe Learning Manager system of record

**Completion date:** 24 Jul 2025 (GMT)

**Learning hours:** 20 mins

**THANK YOU**