

Research Agent – AI-Powered Academic Assistant

Problem Statement

Researchers, students, and R&D professionals spend a significant amount of time performing repetitive and manual tasks:

- Understand complex research queries.
- Retrieve and summarize relevant papers.
- Manage references and citations.
- Assist in writing research papers by drafting sections.
- Organize and generate research reports.
- Enhance overall research productivity and innovation.

Objectives of the Research Agent

1. Literature Search Automation
2. Summarization of Research Papers
3. Reference and Citation Management
4. Content Drafting
5. Report Generation
6. Hypothesis Generation and Suggestion
7. Conversational Interface

Core Technologies Used

Mandatory:

- IBM Cloud Lite Services (IBM Watson NLP, IBM Cloud Functions, IBM Cloudant, IBM App Connect)
- IBM Granite Foundation Models

Optional:

- Node.js/Flask API
- React or IBM Carbon Design System
- ArXiv or Semantic Scholar API

Architecture Overview

User Interface (Web App / Chatbot)

- > Query Understanding (IBM Granite)
- > Search Handler (IBM Cloud Functions)
- > Paper Retrieval (External APIs via App Connect)
- > Summarization + Metadata Extraction (IBM Watson NLP / Granite)
- > Storage (IBM Cloudant)
- > Report Generator / Hypothesis Generator (IBM Granite LLMs)

Functionality Modules

1. Query Understanding: Understand user intent and extract keywords.
2. Literature Retriever: Searches databases and returns papers.
3. Summarizer: Extracts key elements using NLP.
4. Reference Manager: Manages citations.
5. Draft Generator: Writes section drafts.
6. Report Builder: Compiles content into structured reports.
7. Chat Interface: Supports conversational research.

Example Use Case

User Input: 'Summarize the latest research on quantum computing in cryptography'

Agent Actions:

- Extract keywords
- Search databases
- Retrieve papers
- Summarize using IBM Granite
- Generate citations
- Display summary and export options

Development Stack

- Frontend: React.js / IBM Carbon Design
- Backend: Node.js / Python Flask
- LLM/NLP: IBM Granite Foundation Models
- Paper Retrieval: Semantic Scholar, ArXiv API
- Storage: IBM Cloudant
- Logic: IBM Cloud Functions
- Integration: IBM App Connect

Benefits

- Saves time in literature review
- Reduces citation errors
- Improves research quality
- Enables faster innovation

Security & Ethics Consideration

- Data privacy maintained
- Bias control via explainable AI
- Verified citations

FAQs

Q: Can it work with non-technical subjects?

A: Yes, it supports multi-domain understanding.

Q: Can I export reports?

A: Yes, in PDF or DOCX.

Q: Does it support plagiarism checks?

A: Can be added via third-party APIs.