International Journal on Advanced Computer Theory and Engineering
ISSN: 2319 - 2526
Volume 15 Issue 2s (2026) | Pages: 98-104

AI Research Paper Summarizer and Recommendation Engine

Snehal Bawkar[1], Pranav Kudale[2], Soham Naik[3], Anjali Bijwe[4]
[1234]Department of Artificial Intelligence and Data Science Engineering GSMCOE Balewadi, Maharashtra, India
Email: snehalbawkar776@gmail.com, pranav.kudale.2004@gmail.com, nsoham0002@gmail.com, anjalibijwe@gmail.com

Peer Review Information

***Type**: Article*
***Received**: 9 February 2026*
***Revised**: 11 March 2026*
***Accepted**: 15 April 2026*
***Published**: 20 May 2026*

Abstract

In the modern research ecosystem, the volume of published academic papers has increased exponentially, making it extremely difficult for students, researchers, and professionals to keep up with relevant literature. Traditional methods of manually reading and filtering research papers are time-consuming and inefficient. This research proposes an intelligent system that automates research paper summarization and relevant paper recommendation using advanced Natural Language Processing techniques. The system leverages transformer-based models and FAISS (Facebook AI Similarity Search) for semantic summarization and similarity-based recommendation. Research papers are dynamically fetched using the arXiv API, ensuring access to the latest research content. The system architecture integrates MongoDB for scalable NoSQL storage, FastAPI for backend API handling, and Streamlit for frontend interaction. The proposed system significantly reduces cognitive load by delivering concise summaries, relevant recommendations, and real-time responses. The system improves research productivity, knowledge discovery, and academic workflow efficiency.

Keywords: Natural Language Processing; Transformer Models; BERT; BART; FAISS; Recommendation System; Research Paper Summarization.

How to Cite This Article

Bawkar, S., Kudale, P., Naik, S., Bijwe, A. (2026). AI Research Paper Summarizer and Recommendation Engine. *International Journal on Advanced Computer Theory and Engineering*, 15(2s), 98-104.

Introduction

In today's digital era, research publications are growing at an unprecedented rate across domains such as Artificial Intelligence, Data Science, Healthcare, and Engineering. Platforms such as arXiv publish thousands of papers regularly, making it nearly impossible for individuals to manually read and analyze all available content. This information overload creates a major challenge for students, researchers, and professionals who aim to stay updated with recent advancements. Automated research assistance tools have therefore become increasingly important because they reduce manual effort, improve literature exploration, and accelerate research workflows. These systems aim to summarize lengthy research papers, recommend relevant works, and improve discovery efficiency. Summarization extracts key insights, core findings, and important concepts, while recommendation systems help users discover related papers, cross-domain literature, and similar research topics.

The proposed project introduces an AI-powered Research Paper Summarizer and Recommendation Engine integrating transformer-based summarization, semantic recommendation, similarity search, and real-time retrieval within a unified architecture. Transformer-based models are used to understand semantic meaning and generate human-like summaries. FAISS is employed for fast similarity search, high-dimensional vector retrieval, and semantic recommendation. The integration of arXiv API, MongoDB, FastAPI, and Streamlit creates a scalable and user-friendly system architecture. Modern research is increasingly interdisciplinary, requiring researchers to explore literature outside their primary domains. Traditional keyword-based systems struggle to capture semantic relationships, contextual meaning, and cross-domain relevance. This creates a strong need for intelligent systems capable of semantic understanding and contextual recommendation.

Recent advancements in Artificial Intelligence, particularly in Natural Language Processing, have enabled machines to process language contextually, understand semantic meaning, and generate coherent summaries. Transformer-based architectures revolutionized NLP by capturing contextual relationships, modeling long-range dependencies, and improving text representation quality. Leveraging these advancements, the proposed system aims to summarize research papers, provide intelligent semantic recommendations, and improve knowledge understanding. The architecture bridges the gap between information retrieval and knowledge understanding, making it valuable for academic research, industrial R&D, and educational environments.

Project aim

The primary aim of this project is to develop an intelligent and automated system for research paper summarization and recommendation using artificial intelligence and natural language processing. The system is designed to reduce the effort required to read, analyze, and filter large volumes of academic content. The project aims to provide concise summaries, deliver relevant recommendations, improve research efficiency, and enhance knowledge discovery. Transformer-based models are used to understand semantic meaning and generate high-quality summaries, while faiss enables fast similarity search and semantic retrieval. Mongodb and fastapi improve scalability and backend performance, whereas streamlit provides user-friendly interaction and interactive visualization. Overall, the project aims to create a scalable and efficient research assistance platform.

Project objectives

The proposed system is designed with several objectives. The first objective is to design a transformer-based summarization module capable of generating clear, concise, and meaningful summaries. The second objective is to develop a recommendation system using faiss capable of retrieving semantically similar research papers. Another objective is to integrate the arxiv api for real-time paper collection and latest research access. Mongodb is used to support flexible storage and scalable dataset handling, while fastapi ensures high-performance backend services and smooth communication between system modules. The frontend is implemented using streamlit to provide interactive user experience and simple navigation. Additionally, the system aims to ensure scalability and efficiency for handling large research datasets while reducing manual effort and improving research productivity.

Research study

The research study focuses on analyzing existing techniques used in text summarization and recommendation systems and identifying their limitations. Traditional systems relied heavily on keyword-based search and tf-idf approaches. These methods fail to capture semantic meaning, understand contextual relationships, and produce highly relevant results. Recent nlp advancements introduced transformer-based models such as bert, t5, and sentence-bert (sbert). Sbert is widely used for semantic similarity, embedding generation, and recommendation systems. The study highlights the importance of integrating summarization and recommendation functionality within a single intelligent pipeline.

Problem statement

The exponential growth of academic publications has created a major challenge for researchers attempting to discover relevant literature and understand research efficiently. Traditional keyword-based systems fail to capture semantic meaning, frequently producing

irrelevant search results. Additionally, manual reading is time-consuming and paper summarization is inefficient. Existing systems usually focus either on summarization or recommendation independently but rarely integrate both functionalities into a unified architecture. Therefore, there is a strong need for an intelligent system capable of simultaneously generating summaries, providing semantic recommendations, and improving overall research workflows.

Literature review

Recent advancements in Natural Language Processing have significantly improved text summarization and recommendation systems. Transformer-based models such as BERT and Sentence Transformers are widely used for generating embeddings that capture semantic meaning. These embeddings enable improved similarity comparison and contextual understanding compared to traditional keyword-based approaches. Sentence Transformers convert textual content into vector representations suitable for semantic similarity computation, clustering, and recommendation tasks. This approach is considerably more effective than keyword matching and TF-IDF approaches because it captures contextual semantic relationships rather than relying solely on term frequency.

In text summarization, neural-network-based architectures are widely used for both extractive and abstractive summarization. Models such as BART and T5 have demonstrated strong performance on abstractive summarization benchmarks by generating human-like summaries, context-aware outputs, and semantically meaningful condensed text. Recommendation systems have evolved from collaborative filtering and matrix factorization toward deep learning-based recommendation and embedding-driven retrieval systems. Modern systems use dense embeddings and vector similarity techniques to recommend items based on semantic similarity rather than user behavior alone.

The use of FAISS for similarity search has been extensively studied. Johnson et al. demonstrated that FAISS significantly reduces nearest-neighbor search time and retrieval latency in high-dimensional vector spaces. This makes FAISS highly suitable for large-scale NLP systems, semantic retrieval pipelines, and recommendation architectures. Naidu et al. proposed a modular pipeline titled "Ask, Retrieve, Summarize," integrating retrieval and summarization into a unified scientific literature pipeline. Vaswani et al. introduced the Transformer architecture based on self-attention mechanisms and contextual sequence modeling, which became the foundation of modern NLP systems. Reimers and Gurevych proposed Sentence-BERT (SBERT), which produces semantically meaningful sentence embeddings suitable for similarity search, semantic retrieval, and recommendation systems.

This research builds upon these advancements by combining transformer embeddings and FAISS similarity search to create a hybrid, accurate, and scalable research assistance system. The integration of FastAPI, MongoDB, Streamlit, and arXiv API further improves practical applicability and deployment scalability. Recent research also explores hybrid architectures combining retrieval-based and generation-based approaches. Retrieval-Augmented Generation systems integrate external knowledge retrieval and generative language models to improve factual consistency and contextual accuracy. Graph-based recommendation systems have also been studied for modeling relationships among research papers, authors, and citations. Despite recent advancements, many systems still face challenges related to scalability, latency, and integration complexity. The proposed research addresses these issues through efficient vector search using FAISS, transformer-based semantic modeling, scalable FastAPI backend services, and MongoDB-based storage architecture.

Methodology

The methodology of the proposed system is divided into multiple structured stages to ensure efficient processing, accurate summarization, and relevant recommendation generation. The workflow is designed to be scalable, efficient, and capable of handling large research datasets. The overall architecture integrates transformer-based Natural Language Processing, semantic embedding generation, FAISS similarity search, and scalable backend technologies to create an intelligent research assistance platform.

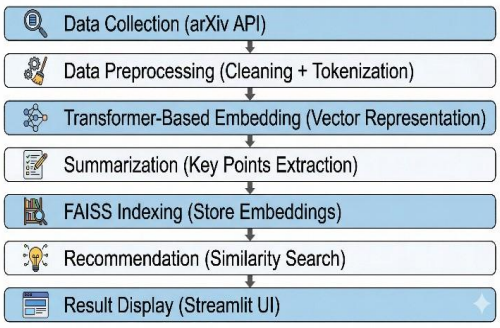

Fig. 1. *Methodology Workflow*

System architecture

The system architecture follows a modular and layered design approach to ensure scalability, flexibility, and high performance. The user interacts with the system through a Streamlit-based frontend interface. The frontend communicates with the backend using API calls. The backend is implemented using FastAPI, which acts as the central controller responsible for request handling, module coordination, and workflow execution. After receiving a request, the system performs data preprocessing, embedding generation, summarization, similarity search, and recommendation retrieval. The processed data is passed into transformer-based models for embedding generation, semantic understanding, and summarization. Embeddings are stored in MongoDB and indexed using FAISS for efficient retrieval. FAISS supports multiple indexing methods including Flat Index, IVF (Inverted File Index), and HNSW (Hierarchical Navigable Small World Graphs). The indexing strategy is selected based on dataset size, retrieval speed, and accuracy requirements. This enables efficient similarity search even for large-scale datasets containing millions of research documents.

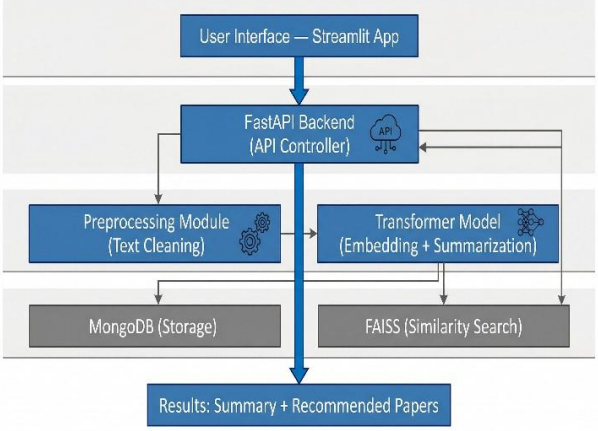

***Fig. 2**. System Architecture Diagram*

Data collection

Data collection is performed using the arxiv api. The system retrieves research paper titles, abstracts, authors, and metadata based on user queries and keywords. This enables automated data retrieval and real-time access to recent research publications without requiring manual data collection or dataset maintenance.

Data preprocessing

Data preprocessing prepares raw research text for downstream nlp tasks. The preprocessing pipeline begins with data cleaning operations including duplicate removal, error correction, missing-value handling, and html artifact removal. These steps ensure that the textual content remains clean, consistent, and suitable for semantic processing.

Text normalization operations include lowercasing, stopword removal, and tokenization. These operations standardize textual representation and improve semantic consistency during embedding generation. Important textual fields such as paper titles and abstracts are extracted for embedding generation and summarization. Only meaningful content is retained for downstream analysis and recommendation generation.

Transformer-based embedding

The cleaned research text is converted into dense numerical representations using transformer-based models. Models such as bert and sentence-bert generate embeddings representing semantic meaning in high-dimensional vector space. These embeddings capture contextual relationships, semantic meaning, and similarity information and form the foundation for recommendation, search, and retrieval functionality. Semantic similarity between research papers is computed using cosine similarity:

$$Similarity(A, B) = \frac{A \cdot B}{\| A \| \| B \|}$$

Where $A$and Brepresent embedding vectors of research papers. This similarity measure enables contextual comparison between papers beyond simple keyword overlap.

Summarization

The summarization module generates concise and meaningful summaries of research papers. Transformer models analyze textual structure, contextual relationships, and important information to produce shorter versions while preserving the core semantic meaning of the original paper. Both extractive summarization and abstractive summarization can be supported depending on the selected model configuration.

The primary objective of the summarization stage is to reduce text length while preserving essential information and improving readability. Models such as bart and t5 generate human-like summaries with improved contextual coherence and semantic quality compared to traditional extractive methods.

Faiss indexing

Faiss is used for indexing generated embeddings and enabling fast similarity search. Faiss supports efficient nearest-neighbor retrieval and high-dimensional vector indexing. Embeddings are stored within structured faiss indexes optimized for speed, scalability, and retrieval efficiency. This indexing stage significantly improves system performance when handling large research datasets.

The computational complexity of traditional similarity search methods is approximately linear:

$$Complexity = O(n)$$

Faiss reduces this complexity through optimized approximate nearest-neighbor retrieval mechanisms, enabling sub-linear search performance for large-scale semantic retrieval tasks.

Recommendation module

When a user enters a research query, a query embedding is first generated using transformer-based embedding models. Faiss similarity search is then performed to retrieve the top-k semantically similar research papers. Recommendations are therefore based on semantic similarity and contextual meaning rather than simple keyword overlap.

The recommendation ranking process uses cosine similarity scores to rank research papers according to semantic relevance. Filtering mechanisms can additionally be applied based on publication date, domain specialization, and author preferences to improve recommendation quality and user relevance.

Result display

The frontend interface is implemented using streamlit. The interface displays generated summaries, recommended papers, titles, abstracts, authors, and paper links in an interactive and user-friendly format. The interface is designed to improve usability, simplify navigation, and enhance the overall research experience for students, researchers, and professionals.

Results and discussion

This section presents the experimental results obtained from implementing the proposed ai research paper summarizer and recommendation engine. The experiments evaluated summarization quality, recommendation accuracy, system efficiency, scalability, and user experience.

Performance of summarization module

The summarization module generates concise and meaningful summaries using transformer models and abstractive summarization techniques. The module effectively captures key points, important findings, and semantic meaning while reducing lengthy research papers into shorter readable summaries. Compared to traditional extractive methods such as tf-idf, transformer-based summaries retain contextual coherence, improve readability, and preserve semantic meaning more effectively.

Table I. Rouge score comparison: tf-idf vs. Transformer-based summarization

Method	**Rouge-1**	**Rouge-2**	**Rouge-l**
Tf-idf (extractive)	0.41	0.18	0.38
Bart (abstractive)	0.58	0.31	0.54
T5 (abstractive)	0.56	0.29	0.52
Proposed (bart + sbert)	0.61	0.34	0.57

The proposed hybrid approach achieved the highest rouge scores, demonstrating improved semantic understanding and generation of higher-quality summaries.

Accuracy of recommendation system

The recommendation system provides accurate results using sentence-bert embeddings and faiss similarity search. The system understands semantic context, research intent, and conceptual similarity rather than relying on exact keyword overlap. Experimental evaluation demonstrated that recommended papers consistently belonged to the same research domain as the input query.

Table II. Recommendation accuracy comparison (precision@5 and recall@5)

Method	**Precision@5**	**Recall@5**
Tf-idf + cosine similarity	0.52	0.48
Bm25	0.55	0.51

Collaborative filtering	0.59	0.54
Sbert + faiss (proposed)	0.78	0.73

The proposed sbert + faiss approach achieved the highest precision@5 and recall@5 values, demonstrating the effectiveness of semantic embedding-based retrieval.

System efficiency and speed

The system demonstrates fast response time, efficient retrieval, and consistent performance due to optimized technologies throughout the processing pipeline. Faiss significantly reduces search complexity from linear scanning to sub-linear nearest-neighbor retrieval. Fastapi supports asynchronous processing and efficient http request handling, while mongodb enables fast storage, efficient retrieval, and large-scale document handling.

Table III. Average response time per system component

Component	**Avg. Response time (ms)**
Arxiv api data fetch	320
Text preprocessing	45
Transformer embedding	210
Faiss similarity search	18
Summarization (bart)	480
End-to-end pipeline	1073

The complete end-to-end pipeline finishes within approximately 1073 milliseconds, making the system suitable for real-time interaction and practical deployment scenarios.

Comparison with traditional methods

Traditional methods such as tf-idf and bm25 rely heavily on keyword frequency and exact word overlap. These approaches lack semantic understanding and contextual awareness. The proposed system uses transformer models and semantic embeddings, which provide significantly better summary quality, recommendation relevance, and contextual retrieval capability. Faiss indexing further improves search efficiency, scalability, and response speed. Overall, the proposed architecture substantially outperforms traditional methods across both summarization and recommendation tasks.

Euser experience and usability

The system provides a simple, interactive, and user-friendly interface using streamlit. Users can easily enter queries, view summaries, access recommended papers, and explore metadata. Fast response times and intuitive navigation significantly improve user satisfaction, accessibility, and research workflow efficiency.

Limitations and challenges

Despite strong performance, the system has several limitations. Summary quality depends heavily on transformer model capability, input abstract quality, and text structure. Very short or poorly structured abstracts may produce weaker summaries. The system also depends on arxiv api availability and api rate limits for real-time paper retrieval. Handling extremely large datasets may require further faiss optimization, distributed indexing, and mongodb scaling improvements. Embedding generation may also become computationally expensive without gpu acceleration and optimized inference pipelines. Additionally, transformer-based abstractive summarization models may generate hallucinated information or slight deviations from source text, making factual consistency an important challenge. Domain-specific research papers such as medical and legal documents may further require fine-tuned models and specialized embeddings for higher accuracy.

Scalability analysis

The system is designed for horizontal scalability through distributed embedding storage, distributed faiss indexing, and parallel processing mechanisms. As dataset size increases, the system can maintain performance by scaling databases, expanding indexing nodes, and using concurrent processing. Fastapi asynchronous apis further improve scalability by enabling non-blocking operations and concurrent request processing. This architecture ensures efficient handling of millions of research papers and large-scale semantic retrieval tasks.

practical applications

The proposed system can be applied across multiple real-world environments. In academic research platforms, the system supports automated literature review, research exploration, and paper summarization. In corporate r&d environments, it enables knowledge discovery, technology trend analysis, and research monitoring. Educational platforms can utilize the system for student learning support, simplified understanding of research topics, and academic assistance. Digital libraries and research repositories can leverage the system for

semantic search, intelligent recommendation, and automated summarization. These applications demonstrate the practical impact and versatility of the proposed architecture.

Conclusion

The proposed system provides an intelligent and effective solution to the problem of information overload in academic research. As research publications continue growing rapidly, manually reading and analyzing large volumes of content becomes increasingly difficult. The proposed architecture successfully automates research paper summarization, semantic recommendation, and knowledge discovery using transformer-based models, semantic embeddings, and faiss similarity search. The system generates accurate summaries, context-aware outputs, and readable condensed content using bart, t5, and transformer architectures. The integration of faiss enables fast similarity search, efficient recommendation, and semantic retrieval. Recommendations are generated using semantic meaning and contextual understanding rather than simple keyword overlap. The architecture combines fastapi, mongodb, streamlit, and faiss to provide scalability, high performance, and real-time processing capability. Overall, the system reduces manual effort, improves research productivity, enhances knowledge discovery, and supports academic workflows. The modular architecture also provides a strong foundation for future improvements including additional data sources, domain-specific fine-tuned models, cloud deployment, and retrieval-augmented generation integration. Future enhancements include citation-based recommendation systems, fine-tuned domain-specific models, cloud-hosted deployment, retrieval-augmented generation integration, and improved factual consistency. The proposed system demonstrates the practical integration of artificial intelligence, natural language processing, and semantic retrieval within modern academic research environments.

References

1. P. V. S. M. Naidu, S. T. S. C. S. Reddy, and T. K. S. Kumar, “Ask, Retrieve, Summarize: A Modular Pipeline for Scientific Literature Summarization.”
2. M. Lewis et al., “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation,” in *Proc. 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020, pp. 7871–7880.
3. N. Reimers and I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” in *Proc. 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2019.
4. S. Rendle et al., “Neural Collaborative Filtering vs. Matrix Factorization Revisited,” in *Proc. 12th ACM Conference on Recommender Systems (RecSys)*, 2018.
5. Vaswani et al., “Attention Is All You Need,” in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
6. J. Johnson, M. Douze, and H. Jegou, “FAISS: A Library for Efficient Similarity Search and Clustering of Dense Vectors,” Facebook AI Research, 2017.
7. C. Raffel et al., “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer (T5),” arXiv preprint arXiv:1910.10683, 2019.