A Hybrid Agent-Based Framework for Intelligent Academic Program Recommendation in Engineering Education

Dr. Shatha Jawad, National University

Dr. Shatha Jawad has more than 25 years of experience in teaching and more than three years as a software engineer. She had UNESCO Fellowship in the field of Information and Communication Technologies, in 2002. Her Ph.D. is in computer engineering. In 2024, she received the Full-Time Faculty Excellence in Scholarship Award for the School of Technology and Engineering at National University, following her receipt of the Full-Time Faculty Excellence in Teaching Award in 2023. She is a member of the Institute for Learning-enabled Optimization at Scale (TILOS) which has an NSF grant that began on November 1, 2021, for five years. TILOS is a National Science Foundation-funded Artificial Intelligence (AI) Research Institute led by the University of California-San Diego and includes faculty from the Massachusetts Institute of Technology, the University of Pennsylvania, the University of Texas at Austin, Yale University, and the National University.

A Hybrid Agent-Based Framework for Intelligent Academic Program Recommendation in Engineering Education

Abstract

Selecting an academic program that aligns with a student's interests, abilities, and preferred learning style is a crucial yet often challenging decision during the early stages of college. Many students struggle to identify a program that matches their personal strengths, which can lead to disengagement, program switching, or even withdrawal from higher education. To address this issue, this research develops an Artificial Intelligence–driven recommendation system designed to assist students in selecting programs that best fit their individual profiles. The system was developed for the Department of Engineering, Data, and Computer Sciences, which offers seven specialized degree programs that share foundational subjects but differ in focus and learning outcomes.

The proposed system employs a LangChain agent to integrate semantic retrieval, embedding-based scoring, and external knowledge access through Wikipedia in order to evaluate student knowledge areas, soft skills, personal attributes, and career aspirations. Program information and student profiles are encoded into dense semantic embeddings using OpenAI's *text-embedding-3-small* model and indexed using FAISS to support efficient similarity-based retrieval. Experimental evaluation using representative student profiles demonstrates that the proposed approach consistently outperforms random program selection, achieving a positive average gain in similarity scores and producing recommendations that align with expected academic advising outcomes.

Introduction

When students begin their university journey, some focus primarily on their future careers, while others emphasize their current knowledge and personal strengths. In many cases, students believe they understand which academic program best aligns with their goals. However, they may not realize that several programs often overlap in foundational content and differ mainly in focus, learning outcomes, or career orientation. As a result, students may initially select a program that appears suitable, only to discover after several months or a year that it does not fully align with their expectations or abilities. Some students attempt to change programs, while others may leave the university altogether, contributing to dissatisfaction and reduced retention.

In the Department of Engineering, Data, and Computer Sciences, seven undergraduate programs are offered: Computer Science, Construction Management, Cybersecurity, Manufacturing Design Engineering Technology (MDET), Electrical & Computer Engineering, Data Analytics, and Information Technology (IT). Some of these programs share common foundational courses but diverge in specialization and professional outcomes. The developed system leverages program learning outcomes and structured program information to support a personalized academic program recommendation process that assists students in making more informed decisions.

To enable efficient semantic comparison between student profiles and program descriptions, the proposed system employs Facebook AI Similarity Search (FAISS), an open-source library designed for fast and scalable approximate nearest-neighbor (ANN) search over high-dimensional vector embeddings. FAISS provides in-memory indexing and efficient similarity search, making it well-suited for semantic retrieval tasks on moderately sized datasets without the overhead of deploying a full-scale vector database. Prior work has demonstrated that FAISS can significantly improve the efficiency and accuracy of vector similarity search in embedding-based information retrieval systems [1], [2].

In addition to vector-based retrieval, the system integrates large language models (LLMs) and agent-based orchestration to support higher-level reasoning and contextual understanding. Recent advances in generative AI and LLMs, including models provided through OpenAI, have enabled the transformation of unstructured textual information into dense semantic embeddings that capture contextual meaning beyond exact keyword matching. Frameworks such as LangChain provide a structured mechanism for combining LLMs with external tools, data sources, and reasoning workflows. LangChain has been successfully applied in conversational systems, knowledge-query applications, and learning-support platforms, demonstrating its ability to orchestrate complex reasoning and retrieval pipelines [3]-[5].

In the proposed system, a LangChain agent autonomously determines when to apply embedding-based similarity calculations and when to invoke external knowledge sources, such as Wikipedia, to augment incomplete or ambiguous student profiles. This hybrid approach combines the quantitative precision of embedding-based similarity scoring with the contextual reasoning capabilities of large language models (LLMs).

The remainder of this paper is organized as follows. The next section reviews related work relevant to academic program recommendation and agent-based AI systems. The Methodology section then presents the detailed design and architecture of the proposed system. Subsequently, the Evaluation section analyzes the system's performance using quantitative and qualitative metrics. Finally, the Conclusion and Future Work section summarizes the main findings and discusses potential directions for extending this research.

Related Work

Many studies have explored the use of artificial intelligence (AI) to support students throughout their academic journey in higher education. Recent studies have increasingly examined AI and large language model (LLM)–based chatbots as academic support tools, examining their ability to provide tutoring, answer student inquiries, and offer personalized guidance. Al-Abri [6] conducted a multi-dimensional evaluation of ChatGPT as a virtual tutor, demonstrating its effectiveness across multiple academic support functions over a semester-long deployment. Similarly, Stöhr et al. [7] analyzed students' perceptions and usage patterns of AI chatbots across genders, academic levels, and disciplines, highlighting both opportunities and challenges in their adoption. A broader review by Peyton et al. [8] surveyed university chatbot systems designed for student support, illustrating the evolution of chatbots from simple FAQ systems to more advanced conversational agents capable of addressing diverse student needs.

In parallel, other studies have investigated the role of AI in academic success monitoring and early intervention. Learning analytics and machine learning techniques have been employed to develop early-warning systems that identify at-risk students and enable timely academic support.

Akçapınar et al. [9] proposed a learning analytics–based early-warning framework, while Embarak and Hawarna [10] introduced an AI-driven RADAR system enhanced with explainable AI to improve transparency in risk detection. Additionally, AI has been leveraged to personalize learning experiences through adaptive learning platforms, which dynamically adjust instructional content and learning pathways based on learner data. Tan et al. [11] provide a comprehensive review of AI-enabled adaptive learning platforms, emphasizing their potential to improve learning outcomes and student engagement at the university level.

These studies collectively demonstrate the growing role of AI in supporting higher-education students, motivating the development of intelligent systems that extend beyond tutoring and monitoring toward academic decision support.

Peng et al. [12] survey large language model (LLM)- powered agent frameworks in recommender systems, providing a high-level taxonomy of existing approaches. The work offers contextual background but does not implement or evaluate a specific recommendation system. In contrast, the present study develops and evaluates an LLM-driven agent for academic program recommendation in an applied educational setting. Recent study has explored the use of large language models (LLMs) to enhance traditional recommender systems by providing semantic understanding and reasoning beyond simple similarity matching. The book Building Recommender Systems Using LLMs [13] presents foundational concepts related to LLMs, traces the evolution from classical recommendation techniques to LLM-powered systems.

Akiba and Fraboni [14] explore the use of large language models, such as ChatGPT, for AI-supported academic advising, demonstrating that LLMs can provide clear and supportive responses to common student advising questions and thereby improve accessibility and scalability. In a different direction, Ma, Hahsler, and Moore [15] propose a recommender system architecture for university curriculum advising based on structured student and curriculum information; however, their approach relies primarily on deterministic logic. Other studies have incorporated traditional machine learning techniques, such as the work by Atalla et al. [16], which applies machine learning and graph-based analysis to generate personalized study plans and demonstrates improved predictive performance on institutional data. Similarly, Jawad et al. [17] present an AI-based system for engineering program selection using a supervised deep learning model trained on structured student attributes, though the reliance on fixed feature representations limits its ability to perform dynamic reasoning or incorporate external knowledge. In contrast, the present work introduces a hybrid LLM-driven agent architecture that integrates semantic embeddings, vector similarity search, and external knowledge to support dynamic, context-aware academic program recommendation with natural-language explanations.

Methodology

As shown in Figure 1, the proposed AI-based academic program recommendation system is designed to support students in selecting an academic program that aligns with their individual profiles. The system accepts a student profile consisting of knowledge areas, soft skills, personal attributes, and career goals, and produces a personalized program recommendation accompanied by an explanatory rationale and optional similarity scores.

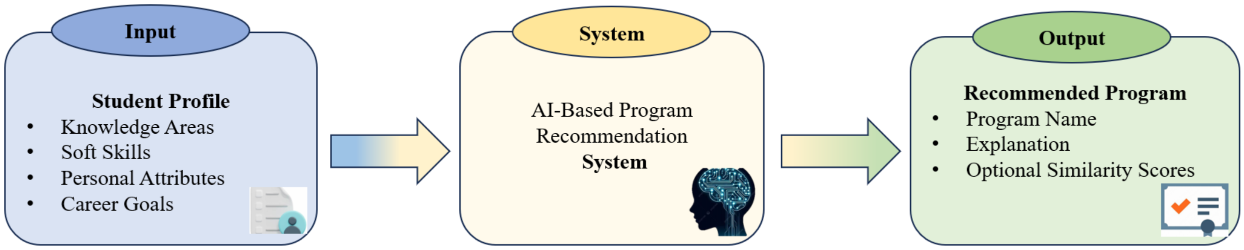

Figure 1- High-level architecture of the proposed AI-based academic program recommendation system.

The system is implemented using LangChain, an open-source framework that enables the integration of large language models (LLMs), such as GPT-4 class models, with external computational tools and structured program data. LangChain provides the orchestration layer through which reasoning, retrieval, and scoring operations are coordinated within the proposed architecture.

As illustrated in Figure 2, the internal architecture of the proposed AI-based program recommendation system consists of several interconnected components. Program data are first encoded using a semantic embedding model and indexed in a FAISS vector store to enable efficient similarity-based retrieval. A LangChain agent orchestrates semantic retrieval, optional Wikipedia-based knowledge augmentation, and a scoring module to generate program recommendations.

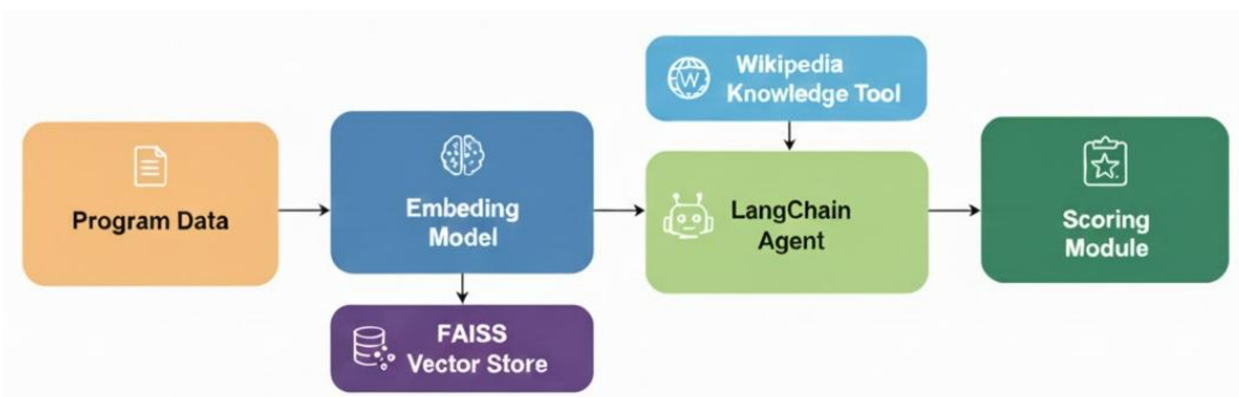

Figure 2- Internal architecture of the proposed AI-based academic program recommendation

As discussed in the Introduction, the system was developed for the Department of Engineering, Data, and Computer Sciences, which offers seven specialized degree programs that share foundational subjects but differ in focus and learning outcomes. Figure 3 illustrates the embedding-based semantic retrieval pipeline employed in the proposed system. Program descriptions are converted into textual representations and encoded into dense vector embeddings using the OpenAI text-embedding-3-small model.

For each academic program, structured information including learning outcomes, technical focus areas, emphasized soft skills, and associated career pathways is concatenated into a unified textual representation prior to embedding. Similarly, each student profile is transformed into structured text by combining the declared knowledge areas, soft skills, personal attributes, and career goals into a single descriptive input. Both program and student representations are encoded using this embedding model. Cosine similarity is then computed between the student embedding and the corresponding program embeddings to obtain the similarity values used in the weighted scoring process.

The resulting embeddings are indexed in a FAISS vector index to support efficient nearest-neighbor retrieval. Given a student profile, the LangChain agent formulates a semantic query to retrieve relevant program candidates, which are subsequently used in the recommendation and scoring process.

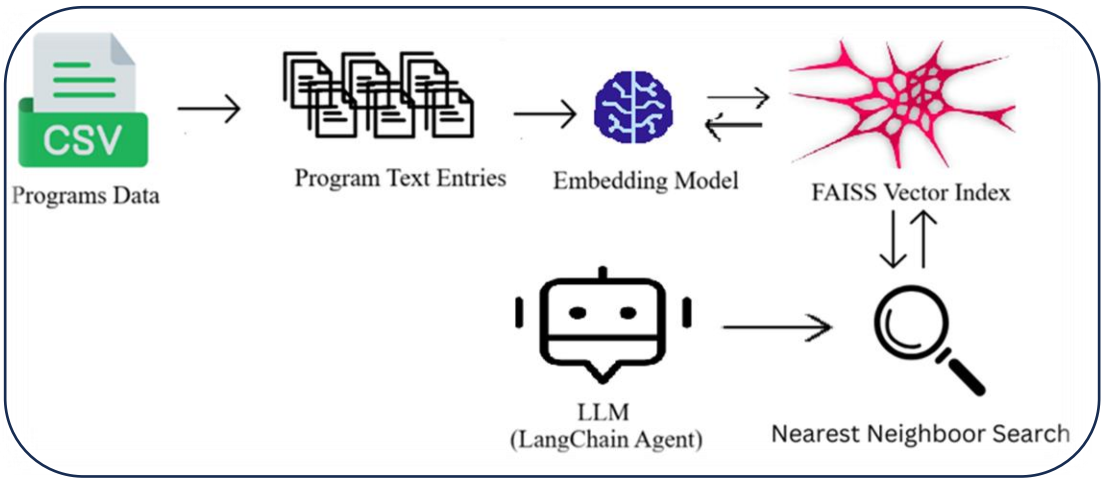

Figure 3- Embedding-Based Semantic Retrieval Pipeline Using FAISS

The decision-making logic of the system is governed by a LangChain agent, as illustrated in Figure 4. At the core of this agent is a large language model (LLM), which serves as the primary reasoning engine. The LLM is embedded within the LangChain agent and operates in conjunction with a structured system prompt that defines the decision-making process.

Rather than functioning as a standalone component, the agent represents a unified structure composed of the LLM, the prompt, an iterative reasoning loop, and tool-calling mechanisms. The LLM continuously evaluates the student profile and intermediate results to determine whether additional actions, such as semantic retrieval or external knowledge lookup, are required.

Within the reasoning loop, the LLM assesses whether the student profile contains unknown or ambiguous terms that may benefit from external knowledge. If such terms are detected, the agent issues a live request to Wikipedia's API to retrieve concise background information. The retrieved knowledge is then incorporated into subsequent reasoning steps. If no external knowledge is required, the agent proceeds directly to program retrieval and scoring.

Following candidate retrieval, the system applies a weighted similarity scoring module to select the most suitable academic program. For each candidate program, cosine similarity is computed between the student profile and program attributes across four dimensions: knowledge areas, soft skills, personal attributes, and career goals. These similarity scores are combined using a predefined weighting scheme of 40% for knowledge areas, 10% for soft skills, 10% for personal attributes, and 40% for career goals. The weighting parameters can be adjusted to reflect different student priorities or advising objectives. In the current implementation, this weighting distribution was empirically selected based on preliminary experimentation, as it yielded the most consistent alignment between student profiles and recommended programs.

The weighted scoring mechanism ensures that both academic preparedness and long-term career alignment are emphasized in the recommendation process. The program with the highest overall similarity score is selected as the final recommendation. These numerical scores are subsequently used by the LLM to generate a natural-language explanation, providing transparency and interpretability for the recommendation outcome.

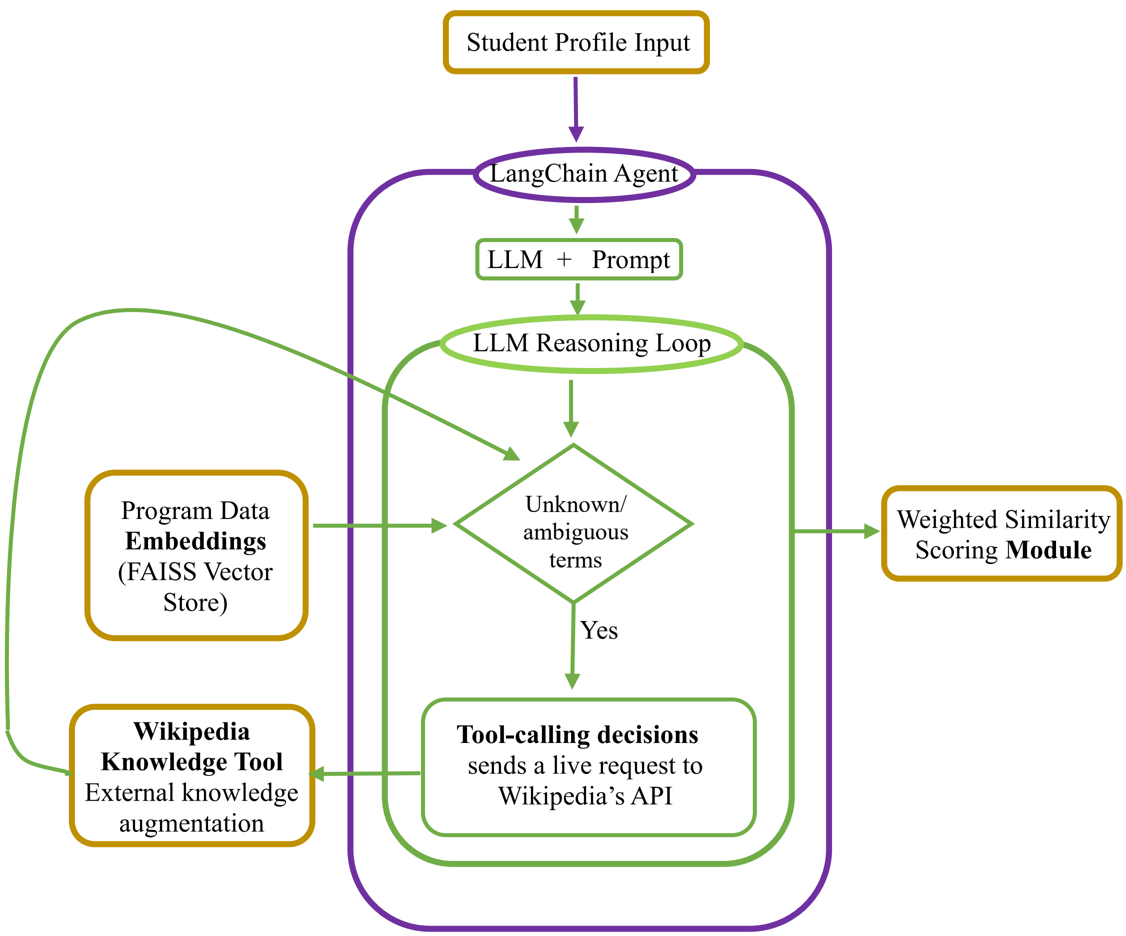

Figure 4- LangChain Agent–Driven Decision Flow for Academic Program Recommendation

Evaluation

The developed system was evaluated on its ability to recommend one of seven academic programs offered by the Department of Engineering, Data, and Computer Sciences. The evaluation was conducted using synthetic yet realistic student profiles, designed to reflect typical academic backgrounds and career aspirations observed at the undergraduate level. Each student profile consists of four attributes: knowledge areas, soft skills, personal attributes, and career goals.

Since no labeled ground truth exists for determining a single "correct" academic program for a given student, the evaluation employs multiple complementary strategies to assess recommendation quality, alignment, and interpretability.

Metric 1: Weighted Similarity Score

The primary quantitative evaluation metric used in this study is the weighted similarity score.

For each student profile s and recommended program p, the system computes an overall alignment score defined as:

$$Score(s,p) = 0.4 * sim_{Knowledge} + 0.1 * sim_{soft} + 0.1 * sim_{personal} + 0.4 * sim_{careers}$$

where each sim represents the cosine similarity between the student profile and the corresponding program attributes in the embedding space.

This metric quantifies the degree of semantic alignment between a student and the recommended program across all four dimensions. Higher scores indicate stronger alignment and, consequently, greater suitability of the recommended program.

Illustrative Examples

To demonstrate the behavior of the proposed system, an example student profile and its corresponding recommendation are presented below.

Student Profile 1

- Knowledge Areas: Technology and engineering systems, CAD basics, building information modeling (BIM)
- Soft Skills: Design thinking, effective communication, analytical thinking
- Personal Attributes: Interest in business processes, strong work ethic, lifelong learning
- Career Goals: Automation and robotics engineer, manufacturing engineer, process improvement specialist

System Recommendation

Recommended Program: Manufacturing Design Engineering Technology (MDET).

Rationale:
This program aligns closely with your knowledge in technology and engineering systems, CAD basics, and building information modeling (BIM). It also emphasizes soft skills like design thinking and effective communication, which you possess. The career paths available, such as manufacturing engineer and automation and robotics engineer, match your career interests.

Similarity Score Breakdown

For clarity, detailed similarity values are reported to four decimal places in the breakdown, while summary values in tables are rounded to three decimal places for readability.

- Knowledge Similarity: 0.6535
  The program covers core technical areas such as CAD and engineering systems relevant to the student's academic background.

- Soft Skills Similarity: 0.7291
  The curriculum emphasizes design-oriented thinking and communication skills, closely aligning with the student's soft skills.

- Personal Attributes Similarity: 0.5785
  The program values hands-on learning and continuous improvement, which resonate with the student's personal attributes.

- Career Similarity: 0.808
  The program's career outcomes strongly align with the student's desired roles in manufacturing and automation.

- Overall Weighted Score: 0.7154

Based on both quantitative similarity scores and qualitative reasoning, the MDET program represents the most suitable recommendation for this student profile, providing a strong foundation for future careers in engineering and technology.

Top-3 Program Recommendations

In addition to identifying the single best-fit program, the proposed system is capable of ranking the top three academic programs for each student profile. These ranked recommendations provide additional transparency and allow students and academic advisors to explore alternative programs that also demonstrate strong alignment. The ranking is based on the weighted similarity score, where Rank 1 represents the highest alignment and Rank 3 represents the lowest alignment among the top candidates.

Table 1- presents the top three recommended programs for Student Profile 1

Rank	Program	Score
1	Manufacturing Design Engineering Technology (MDET)	0.715
2	Construction Management	0.581
3	Electrical & Computer Engineering	0.546

Student Profile 2

- Knowledge Areas: Mathematical thinking, basic knowledge of data structures and algorithms, and computer hardware
- Soft Skills: Problem-solving, effective communication, and attention to detail.
- Personal Attributes: Customer-centric mindset, problem-solving mindset, and interest in technology.

- Career Goals: Information security auditor, web developer (frontend/backend), and frontend developer.

Recommended Program: Computer Science

Rationale:
This program aligns closely with your knowledge of mathematical thinking, data structures, and computer hardware. It also emphasizes problem-solving and effective communication, which are key soft skills you possess. The career opportunities in software development and web development match your career interests well.

Table 2- presents the top three recommended programs for Student Profile 2

Rank	Program	Score
1	Computer Science	0.674
2	Information Technology (IT)	0.629
3	Cybersecurity	0.590

Metric 2: Random Baseline Comparison

To assess whether the proposed system produces recommendations that are meaningfully better than chance, a random baseline comparison was performed. For each student profile s, the system's recommended program p^* and its weighted similarity score were compared against a baseline obtained by randomly selecting programs from the available set. For each profile, the random baseline score was computed as the average similarity over N=100 random program selections (sampled with replacement).

$$AvgRandom(s) = \frac{1}{N}\sum_{i=1}^{N} Score(s, p_i^{rand})$$

$$Gain(s) = Score(s, p^*) - AvgRandom(s)$$

A positive gain indicates that the system recommendation outperforms random program selection for the given student profile.

Table 3 summarizes the results of the random baseline evaluation for three representative student profiles. Across all evaluated profiles, the proposed system consistently achieved higher similarity scores than the random baseline. The average system score was 0.652, compared to an average random baseline score of approximately 0.528, resulting in an average gain of +0.124 over random selection.

Table 3- Random Baseline Comparison Results

Student Profile	Recommended Program	System Score	Random Avg (±σ)	Gain vs Random
Profile 1	Computer Science	0.674	0.547 ± 0.065	+0.127
Profile 2	Information Technology	0.629	0.518 ± 0.064	+0.111
Profile 3	Construction Management	0.653	0.519 ± 0.095	+0.135
Average	**-**	**0.652**	-	**+0.124**

Although outperforming random selection may appear expected for a structured semantic recommendation framework, the consistent positive gain across diverse profiles indicates that the embedding-based scoring mechanism captures meaningful relationships between student attributes and program characteristics. This comparison therefore validates that the system's recommendations are driven by structured semantic alignment rather than arbitrary selection.

The following illustrative case studies provide additional insight into how the system generates recommendations for individual student profiles.

Student Profile 1

- Knowledge Areas: Mathematical thinking, basic knowledge of data structures and algorithms, and computer hardware.
- Soft Skills: Problem-solving, effective communication, and attention to detail.
- Personal Attributes: Customer-centric mindset, problem-solving mindset, and interest in technology.
- Career Goals: Information security auditor, web developer (frontend/backend), and frontend developer.

Student Profile 2

- Knowledge Areas: Basic electronics, introductory programming knowledge, and IoT systems.
- Soft Skills: Adaptability, analytical mindset, and curiosity.
- Personal Attributes: Lifelong learning, strong work ethic, and leadership potential.
- Career Goals: IT support specialist, ERP specialist, and software developer.

Student Profile 3

- Knowledge Areas: Technology and engineering systems, manufacturing processes, and CAD basics.
- Soft Skills: Collaboration skills, analytical mindset, and problem-solving skills.
- Personal Attributes: Customer and client focus, curiosity and passion for technology, and attention to detail.
- Career Goals: Construction consultant, site manager (construction supervisor), and quantity surveyor.

Illustrative Case Studies System Recommendation

- *Profile 1* represents a student with strong foundations in programming, data structures, and hardware, along with career interests in web development and information security. The system recommended the Computer Science program, achieving a weighted similarity score of 0.674. The explanation highlights alignment in foundational computing knowledge, problem-solving skills, and career pathways, illustrating how the system integrates quantitative scoring with interpretable reasoning.

- *Profile 2* represents a student with strong Basic electronics, introductory programming, and IOT systems, along with career interests in IT support specialist, ERP specialist, and software developer. The system recommended the Information Technology, achieving a weighted similarity score of 0.629.

- *Profile 3* represents a student whose background emphasizes technology and engineering systems, manufacturing processes, and CAD, with career interests centered on construction consulting, site supervision, and quantity surveying. The system recommended Construction Management, achieving a weighted similarity score of 0.653. This recommendation is primarily driven by strong alignment with construction-oriented career outcomes (career similarity 0.77) and consistent matches in soft skills and personal attributes (both 0.62), indicating that the system captures both technical preparation and professional role fit.

These results demonstrate that the proposed recommendation approach captures meaningful semantic relationships between student profiles and academic programs, rather than producing arbitrary recommendations. The consistent positive gain across multiple profiles indicates that the system effectively leverages embedding-based similarity and LLM-guided reasoning to improve program selection quality beyond chance.

It should be noted that attributes such as soft skills and personal characteristics involve inherent subjectivity; however, the embedding-based representation helps capture semantic similarity rather than relying solely on exact keyword matching.

It is also observed that gain values vary across different recommended programs. Programs with broader or overlapping domains, such as Information Technology, Computer Science, and Data Analytics, may produce comparatively lower gains due to semantic similarity across their descriptions. This behavior is consistent with the embedding-based representation, which captures contextual relationships between related programs. Despite this overlap, the system is still able to identify the most suitable program for each student. Incorporating more detailed program descriptions and learning outcomes may further improve the gain for closely related programs.

In Appendix A, we report a representative subset of 24 student profiles selected from the 100 evaluated profiles to ensure coverage across all recommended programs and a range of similarity scores.

Conclusion and Future work

This work focuses on supporting academic advising and program recommendation for new students beginning their journey in STEM-related disciplines. The proposed system accepts a student profile consisting of knowledge areas, soft skills, personal attributes, and career goals, and generates a personalized academic program recommendation accompanied by an explanatory rationale and optional similarity scores. The system incorporates learning outcomes and relevant information from seven undergraduate programs- Computer Science, Construction Management, Cybersecurity, Manufacturing Design Engineering Technology (MDET), Electrical and Computer Engineering, Data Analytics, and Information Technology (IT)- offered by the Department of Engineering, Data, and Computer Sciences. Program descriptions are converted into textual representations and encoded into dense vector embeddings using the OpenAI text-embedding-3-small model. For a given student profile, a LangChain agent orchestrates semantic retrieval, optional Wikipedia-based knowledge augmentation, and a weighted scoring module to generate program recommendations.

Experimental evaluation using representative student profiles demonstrates that the proposed LangChain-agent-driven approach consistently outperforms random program selection, achieving a positive average gain in similarity scores and producing recommendations that align with expected academic advising outcomes. These results highlight the potential of LLM-driven agent architectures to enhance personalized educational guidance, reduce program-selection uncertainty, and support data-informed academic advising in higher education.

While the current implementation relies primarily on semantic embeddings and live Wikipedia-based knowledge augmentation, future work may integrate live web search services to further enrich student profiles and program context with more dynamic and industry-oriented information. Additionally, although the proposed evaluation demonstrates strong semantic alignment and consistency, future research may incorporate expert advisor annotations or longitudinal student outcomes to enable supervised evaluation and further validate the effectiveness of the recommendation system. Future research may also extend the system to larger and more diverse academic program catalogs, enabling evaluation across broader institutional settings.

Appendix A- Representative Student Recommendations

Representative subset of 24 student profiles selected from the evaluation dataset. Profiles are ordered by gain over the random baseline and were chosen to ensure coverage across all academic programs while emphasizing higher-gain recommendations.

Profile ID	Knowledge Summary	Career Goals Summary	Recommended Program	System Score	Gain vs Random
94	Materials, manufacturing, CAD	Construction projects	Construction Management	0.716	+0.168
5	CAD, environmental science, math	Site management	Construction Management	0.705	+0.154
55	Networking, programming basics	Cybersecurity roles	Cybersecurity	0.759	+0.170
98	Construction math, CAD	Quantity surveying	Construction Management	0.693	+0.173
17	Geometry, AutoCAD, manufacturing	Manufacturing engineer	MDET	0.701	+0.149
12	Mathematics, CAD, physics	Engineering technology	MDET	0.684	+0.136
4	Materials, physics, manufacturing	Manufacturing design	MDET	0.678	+0.133
97	Environmental science, CAD	Construction consulting	Construction Management	0.672	+0.128

8	Geometry, environmental science	Project planning	Construction Management	0.669	+0.121
96	Business tech, computing basics	IT project roles	IT	0.671	+0.114
77	Robotics fundamentals, automation	Robotics engineering	MDET	0.662	+0.111
84	Math, hardware, web basics	Software roles	Computer Science	0.696	+0.104
3	Web dev, business technology	IT support	IT	0.668	+0.095
73	Hardware, computing foundations	IT infrastructure	IT	0.661	+0.094
82	Programming fundamentals	Data analysis	Data Analytics	0.630	+0.093
19	Cybersecurity basics	Security analyst	Cybersecurity	0.627	+0.091
88	Databases, analytics	Data analyst	Data Analytics	0.625	+0.088
42	Math, electronics	Systems roles	Electrical & Computer Eng.	0.674	+0.086
11	OS basics, web dev	IT operations	IT	0.612	+0.082
65	Programming, math	Data analytics	Data Analytics	0.607	+0.074
1	Electronics, programming	IT support	IT	0.595	+0.071
56	CAD, materials	Engineering design	Electrical & Computer Eng.	0.583	+0.069
27	Cyber fundamentals	Security compliance	Cybersecurity	0.579	+0.045
30	Engineering basics	Technical roles	Electrical & Computer Eng.	0.568	+0.033

References

[1] M. Douze *et al.*, "The Faiss library," *IEEE Trans. Big Data*, vol. 11, pp. 1–15, 2025. doi: 10.1109/TBDATA.2025.3618474.

[2] S. Safar, B. R. Jose, J. Mathew, and T. Santhanakrishnan, "Recommendation systems with LLM-based semantic embeddings and FAISS similarity search," *Neurocomputing*, vol. 649, Art. no. 130753, Oct. 2025. doi: 10.1016/j.neucom.2025.130753.

[3] S. S. Subasree and N. K. Sreeja, "Leveraging LangChain Framework and Large Language Models for Conversational Chatbot Development," *Int. J. Intell. Syst. Appl. Eng.*, vol. 12, no. 20s, pp. 110–118, 2024.

[4] N.-L. Hsueh and W.-T. Wang, "Development and Evaluation of Learning Portfolio Query System Based on LangChain Framework," *Eng. Proc.*, vol. 92, Art. no. 40, Apr. 2025. doi: 10.3390/engproc2025092040.

[5] S. R. B. Narayan and N. Agarwal, "Introduction to LangChain," in *Mastering LangChain: A Comprehensive Guide to Building Generative AI Applications*, Berkeley, CA, USA: Apress, 2025, ch. 1, pp. 1–9. doi: 10.1007/979-8-8688-1718-2_1.

[6] A. Al-Abri, "Exploring ChatGPT as a virtual tutor: A multi-dimensional analysis of large language models in academic support," *Educ. Inf. Technol.*, vol. 30, no. 12, pp. 17447–17482, Dec. 2025. doi: 10.1007/s10639-025-13484-x.

[7] C. Stöhr, A. W. Ou, and H. Malmström, "Perceptions and usage of AI chatbots among students in higher education across genders, academic levels and fields of study," *Comput. Educ. Artif. Intell.*, vol. 7, Art. no. 100259, 2024. doi: 10.1016/j.caeai.2024.100259.

[8] K. Peyton, S. Unnikrishnan, and B. Mulligan, "A review of university chatbots for student support: FAQs and beyond," *Discov. Educ.*, vol. 4, no. 1, Art. no. 21, Jan. 2025. doi: 10.1007/s44217-025-00397-7.

[9] G. Akçapınar, A. Altun, and P. Aşkar, "Using learning analytics to develop early-warning system for at-risk students," *Int. J. Educ. Technol. High. Educ.*, vol. 16, no. 1, Art. no. 40, Dec. 2019. doi: 10.1186/s41239-019-0172-z.

[10] O. H. Embarak and S. Hawarna, "Enhancing Student Success with XAI-Powered RADAR: An Automated AI-driven System for Early Detection of At-risk Students," *Procedia Comput. Sci.*, vol. 231, pp. 151–160, 2024. doi: 10.1016/j.procs.2023.12.187.

[11] L. Y. Tan, S. Hu, D. J. Yeo, and K. H. Cheong, "Artificial Intelligence-Enabled Adaptive Learning Platforms: A Review," *Comput. Educ. Artif. Intell.*, vol. 8, Art. no. 100318, 2025. doi: 10.1016/j.caeai.2024.100318.

[12] Q. Peng *et al.*, "A Survey on LLM-powered Agents for Recommender Systems," in *Findings of the Association for Computational Linguistics: EMNLP 2025*, Suzhou, China, Nov. 2025, pp. 11574–11583.

[13] J. Wang, *Building Recommender Systems Using Large Language Models*. Switzerland: Springer Nature, 2025.

[14] D. Akiba and M. C. Fraboni, "AI-Supported Academic Advising: Exploring ChatGPT's Current State and Future Potential toward Student Empowerment," *Educ. Sci.*, vol. 13, no. 9, p. 885, Aug. 2023. doi: 10.3390/educsci13090885.

[15] Z. Ma, M. Hahsler, and P. Moore, "A Recommender System Architecture for University Curriculum Advising," in *Proceedings of the AAAI Symposium Series*, vol. 5, no. 1, 2025, pp. 235–241. doi: 10.1609/aaaiss.v5i1.35593.

[16] S. Atalla *et al.*, "An Intelligent Recommendation System for Automating Academic Advising Based on Curriculum Analysis and Performance Modeling," *Mathematics*, vol. 11, no. 5, Art. no. 1098, Feb. 2023. doi: 10.3390/math11051098.

[17] S. Jawad, R. P. Uhlig, M. N. Amin, P. Dey, and B. Sinha, "Using Artificial Intelligence in Academia to Help Students Choose Their Engineering Program," in *Proc. 2023 ASEE Annual Conf. & Expo.*, Baltimore, MD, June 2023.