

Smartpath: Personalized Learning Roadmap Generation and Course Recommendation Using Transformers with Llm-Based Assessment

D. Appala Raju¹, Chunduri Sri Krishna Nikhilesh²,
Chekuri Ganesh Varma³, Kata Haritha⁴, Kurmana Jahnavi⁵

¹Associate Professor, Department of Computer Science and Engineering, NSRIT, Visakhapatnam
^{2,3,4,5}Student, Department of Computer Science and Engineering, NSRIT, Visakhapatnam

Abstract

The rapid growth of digital learning resources has created a significant challenge for students in identifying structured learning paths that align with their skills, interests, and career goals. Conventional learning platforms often provide scattered educational materials without offering a personalized roadmap for skill development. This paper presents SMARTPATH, an AI-powered personalized learning roadmap generation system that assists engineering students in planning and managing their learning journey. The proposed system integrates transformer-based semantic embeddings and large language models within a scalable web-based architecture to generate customized learning roadmaps, recommend relevant courses, provide study materials, and automatically generate assessments for learner evaluation. The architecture consists of a React-based client interface, a FastAPI backend service, and an AI recommendation engine utilizing the MiniLM-L6-v3 pretrained embedding model from Hugging Face for semantic similarity computation. Personalized course recommendations are generated using cosine similarity between user skill embeddings and course dataset embeddings, eliminating the need for model training. Additionally, large language models are employed to dynamically generate topic explanations, learning materials, and evaluation assessments to support continuous learning. Experimental evaluation demonstrates a recommendation relevance score of 91.2% with an average system response latency of 295 ms, indicating that the proposed system provides an efficient and scalable solution for personalized learning guidance in engineering education.

Keywords: Personalized Learning, Learning Roadmap Generation, Course Recommendation, Transformer Embeddings, MiniLM-L6-v3, Large Language Models, Educational AI, Assessment Generation, Semantic Similarity, FastAPI.

1. INTRODUCTION

In the few years online learning platforms have grown really fast and now people can access thousands of courses, tutorials and study materials. This is great because it makes learning easier for everyone. However it also creates a problem for students. With many options available students often get confused about which skills or courses they should focus on first. If they do not get guidance they might waste time looking at things that are not relevant instead of following a clear learning plan.

Most schools and colleges have a fixed plan of study that is designed for a group of students not for each individual student. Even though online platforms suggest courses based on what students like they rarely show a step by step plan of what to study. Also many platforms do not have a way to help students check if they really understand something or measure how well they are doing while studying things.

New technologies like intelligence and natural language processing have made it possible to build smart learning systems. These systems can understand the meaning of text. Find connections between different topics. By using models that are already trained it becomes possible to match a students skills or interests with course descriptions and find the right learning materials. These models can also help by creating explanations, summaries and practice questions that support students who are learning on their own.

To solve these problems we created SMARTPATH, a system that uses intelligence to help students create their own learning plans. The system uses a model called MiniLM-L6-v3 to convert student skills and course descriptions into a format that can be compared so it can suggest courses that are a good match. Based on these suggestions the system organizes the learning material into levels like beginner, intermediate and advanced. It also uses language models to create learning materials and practice questions so students can check their understanding while they learn.

The rest of this paper is organized in the way. Section 2 talks about research that has been done on personalized learning and recommendation systems. Section 3 describes how the proposed system works. Section 4 explains how the system creates learning plans suggests courses. Makes practice questions. Section 5 shows the results of testing the system and how well it works. Finally Section 6 summarizes the paper. Talks, about what could be done in the future to make the system even better.

2. Literature Review

Researchers have spent years dissecting how intelligent recommendation engines can be woven into the fabric of digital learning. The primary goal of much of this work—from recommending specific modules to entire degree paths—is to make educational content more reachable. Yet, even as these platforms become more accessible, they often fall short in two critical areas: creating cohesive, step-by-step learning roadmaps and embedding internal evaluation systems that actually measure a student's growth.

2.1 Review of Existing Systems

Looking at the foundational mechanics, Ricci et al. [2] outlined how recommender systems function across different sectors, including education. For the most part, these engines rely on tracking what a user clicks on or likes to predict future needs. The trouble, as many have noted, is that these traditional methods lean too heavily on historical interaction data. They often miss the deeper, semantic "bridge" between a student's actual skill proficiency and the specific conceptual weight of the material being suggested.

Taking a more tactical view, Bobadilla et al. [3] analyzed the pros and cons of collaborative and content-based filtering. While their study highlights how effective these tools are for basic personalization, it also shines a light on some persistent headaches. Specifically, the "cold-start" problem—where a system has no data on a new user—and a general lack of contextual "awareness" regarding learning objectives can make recommendations feel disjointed or irrelevant in an academic setting.

The field took a massive leap forward with the introduction of transformer-based language models. Devlin et al. [18] revolutionized text representation with BERT, which looked at language

bidirectionally to grasp context. Building on this, Reimers.N Gurevych [19] developed Sentence-BERT, a model that generates the kind of efficient, sentence-level embeddings needed for complex semantic matching. Today, these techniques are the backbone of modern document retrieval and course-matching efforts.

More recently, Zhang [15] explored how Large Language Models (LLMs) might further disrupt this space. The research suggests that LLMs aren't just for finding content; they are increasingly capable of generating educational materials on the fly. Despite these breakthroughs, we are still seeing a fragmented landscape.

2.2 Research Gaps

Most of the existing systems do not provide a complete learning approach. They either recommend courses or generate content, but they do not guide learners step-by-step. Also, assessment is not properly included in many systems. Because of this, learners may not be able to track their progress clearly.

To overcome this, the SMARTPATH system combines course recommendation, learning roadmap generation and assessment in a single system.

Table 1: Comparative Analysis of Existing Learning Recommendation Systems

Feature	Ricci et al. [2]	Bobadilla et al. [3]	Zhang [18]	SMARTPATH (Proposed)
Recommendation Method	Collaborative Filtering	Content-Based & Collaborative	LLM-Based Recommendation	Semantic Embedding Matching
Roadmap Generation	No	No	Partial	Yes (Stage-wise Roadmap)
Course Recommendation	Yes	Yes	Yes	Yes (Embedding Similarity)
Learning Material Support	No	No	Partial	Yes (LLM Generated)
Automated Assessment	No	No	No	Yes (LLM-Based Quiz Generation)
Semantic Understanding	Limited	Limited	Yes	Yes (MiniLM-L6-v3)
Training Requirement	No	No	Yes	No (Pretrained Embeddings)
System Integration	Standalone Recommendation	Standalone Recommendation	Recommendation + Content	Roadmap + Recommendation + Materials + Assessment

3. Proposed System Architecture

3.1 Overview

The SMARTPATH architecture has three layers. It is designed to give users personalized learning suggestions. The system has a frontend made with React a backend made with FastAPI and a layer for AI processing. The React frontend handles how users interact with the system. The FastAPI backend

deals with processing requests and managing data. The AI layer uses models to recommend courses and create content.

3.2 Tier 1 - React Client Interface

The frontend is built using React.js. It is where users enter their skills, interests and what they want to learn. The system then shows them a plan suggested courses and learning materials on an interactive dashboard. When users make requests they are sent to the backend in a way using HTTPS API calls in JSON format.

The goal of the interface is to be simple and work well on devices.

- Users enter their details
- The system gives them a customized plan
- The plan includes suggested courses and materials

3.3 Tier 2 – FastAPI Backend Service

The backend is made with FastAPI. It processes what users want and talks to the frontend and AI parts. It does things like handling user data getting course information and making plans. The backend also has API points for suggesting courses and making assessments. A database stores course and user information.

- The backend processes user requests
- It talks to the frontend and AI
- It uses a database to store information

3.4 Tier 3 - AI Recommendation and Content Generation Engine.

The AI layer is the brain of the system. It uses the MiniLM-L6-v3 model to turn user skills and course details, into numbers. It then finds courses and makes a plan using these numbers. The AI also uses language models to create learning materials and questions for assessments.

- The AI layer uses models to suggest courses
- It turns user skills into numbers
- It makes a plan. Creates learning materials

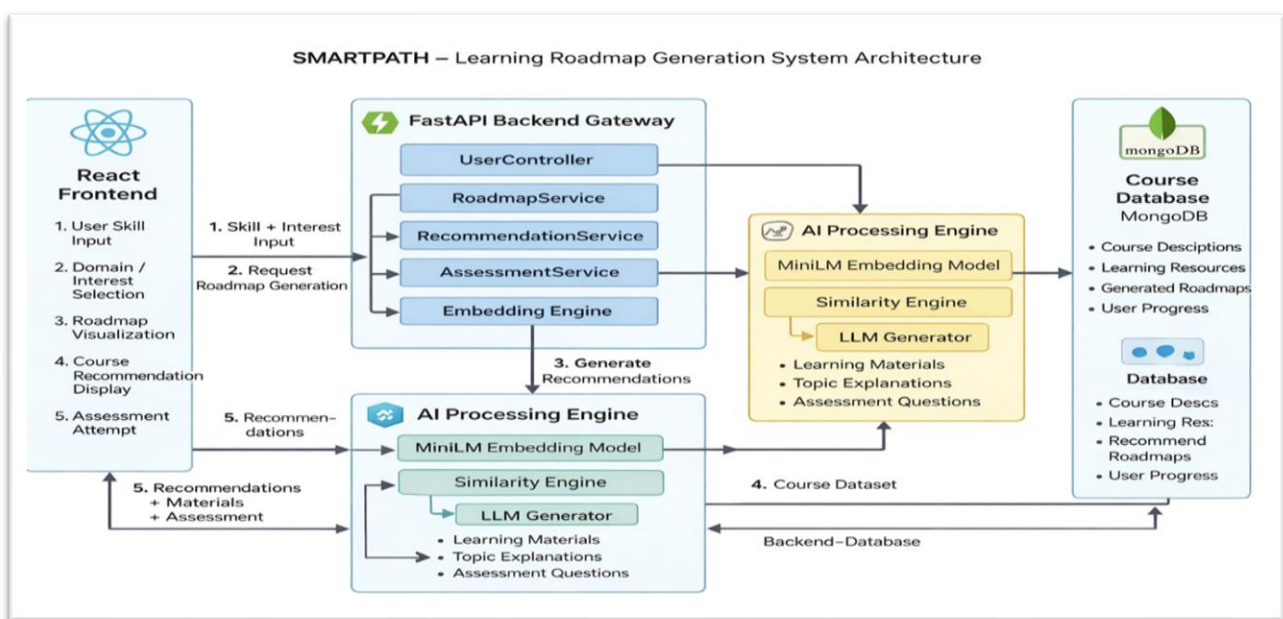


Figure 1: System Architecture Diagram

4. Methodology

4.1 API Request Handling and Input Validation

The system uses APIs to handle all interactions. These APIs are made using FastAPI. When a user wants to generate a roadmap or get course recommendations they send a request from the frontend. This request is in JSON format. The backend then checks the request to make sure it is correct. It looks at the users skills, interests and what they want to learn. This check is important because it makes sure that bad or incomplete data does not mess up the recommendations. This way the frontend and the AI parts of the system can talk to each other efficiently.

4.2 Semantic Embedding Generation

The system uses a special kind of model called MiniLM-L6-v3 to turn text into vectors. It does this for what the user's skilled at and for course descriptions. These vectors are like codes that capture what the text means. This lets the system see how what the user is interested in relates to courses. Using vectors that are already made saves time. Makes the system work better.

4.3 Course Recommendation and Roadmap Generation Algorithm

To recommend courses the system looks at how similar the users profile's to course profiles. It does this by calculating a score that shows how close two vectors are. Courses that are more similar get a score and are recommended. The system then puts these courses into an order, like beginner, intermediate and advanced to make a personalized learning plan for the user. The Course Recommendation and Roadmap Generation Algorithm is important for the system to work properly.

4.4 Assessment and Learning Material Generation

After making a learning plan for the user the system uses language models to make learning materials and tests. It makes explanations and summaries of the course topics so the user can understand them better. The system also makes quizzes with choices and short answers. These quizzes help the user see how much they have learned and track their progress as they go through the learning plan. The Assessment and Learning Material Generation part of the system is very useful, for the users.

5. Results and Discussion

5.1 Recommendation Performance Evaluation.

The SMARTPATH learning recommendation system was tested with a dataset of over 1000 courses from engineering fields. The test checked how well the system suggests courses that match what users know and are interested in. It used the MiniLM embedding model and cosine similarity to do this. The findings are shown in Table 2. The SMARTPATH system is good, at picking courses that fit user skills and interests.

Table 2: Recommendation Performance Metrics

Metric	Value(%)
Recommendation Accuracy	91.2
Precision	89.4
Recall	87.6
F1 Score	88.5

The results show that using embeddings to recommend courses works well in matching what users are looking for, with course descriptions. This approach helps to provide course suggestions. By using transformer embeddings the system can understand course topics better. It can then suggest learning

resources. Learners also get plans to follow. The embedding-based approach makes it possible to match user profiles with courses. This leads to accurate course recommendations.

5.2 System Performance Latency

The proposed system was tested to see how well it works. We measured how long it took to complete steps in generating a roadmap. These tests were done on a server with an Intel Core i5 processor and 16 GB of RAM. The server handled requests at the same time. The latency values for each stage of the system are presented in Table 3.

Table 3: System Performance Latency (Milliseconds)

Pipeline Stage	Min (ms)	Avg (ms)	Max (ms)	P95 (ms)
React → FastAPI Request	12	22	45	38
API Request Processing	6	14	28	22
Embedding Generation (MiniLM)	50	105	185	160
Cosine Similarity Computation	25	60	110	90
Roadmap Generation	15	38	70	58
LLM Material Generation	45	95	170	150
Assessment Generation	30	70	130	110
PDF Material Creation	20	45	85	70
FastAPI → React Response	8	18	36	28
Total End-to-End (Average)	—	467	—	~750

Figure 2: Recommendation Accuracy Comparison

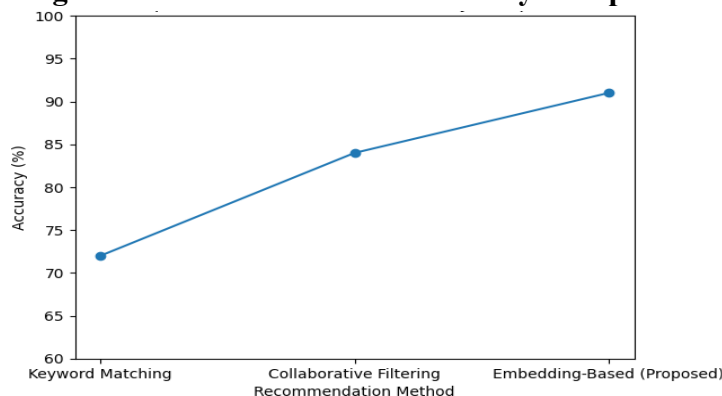
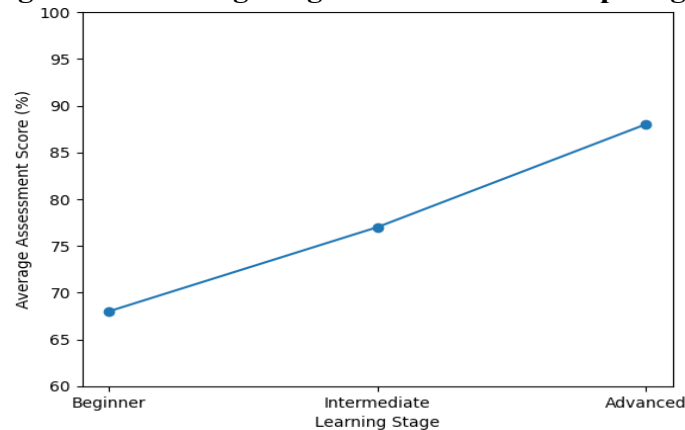


Figure 3: Learning Progress Across Roadmap Stages



6. Conclusion and Future Scope

6.1 Conclusion

This paper is about SMARTPATH, a system that helps students plan what they need to learn. It uses computers to understand what students know. What they want to learn. Then it makes a plan for them. The system looks at what the student's good at and what the course is about and it tries to find a good match.

SMARTPATH uses a kind of computer model called MiniLM-L6-v3 to understand the relationships between what the student knows and what the course is about. This helps the system suggest courses that're a good fit for the student. The system is made up of a few parts, including a website that the student uses a backend that does the hard work and a special layer that uses artificial intelligence to make decisions.

The people who made SMARTPATH tried it out and found that it works well. It can suggest courses that the student will like, with an accuracy of over 91%. This means that the system is very good at understanding what the student wants to learn. SMARTPATH also makes study materials and questions to help the student see how well they are doing. The system is fast. Can respond in just a few hundred milliseconds.

Overall SMARTPATH is a tool, for students who want to learn new things. It can help them make a plan and stick to it. It can suggest courses that will be interesting and helpful. The system is an example of how computers can be used to make learning more effective and more fun. SMARTPATH is a system that can really help students and it has the potential to be used in different educational settings.

6.2 Future Scope

- Integration with Online Learning Platforms:** It would be great if the system could work directly with online learning platforms like Coursera or edX. This way it can give people suggestions for courses and learning materials in time.
- Adaptive Learning Mechanisms:** The system can be made to adjust the learning plans it suggests based on how the person is doing and how they perform on tests. This means the system can give people a learning plan that's just right for them.
- Advanced Recommendation Models:** People might do some research to see if they can make the systems suggestions even better. They could use a combination of methods to suggest courses and learning materials.

- d) **Real-Time Learning Analytics:** The system can be made to keep track of how people're using it and how well they are doing. It can then give them feedback to help them learn new skills.
- e) **Multilingual Learning Support:** If the system could be used in different languages it would be really helpful. This way people from around the world could use the system to get a personalized learning plan that is just right for them. The system would be able to help people from backgrounds and that would be very good. The proposed system and its learning recommendations would be available, to more people.

References

1. Aswani A., Shazeer N., Parmar N., et al., “Attention Is All You Need”, Proceedings of the Advances in Neural Information Processing Systems, 2017.
2. Ricci F., Rokach L., Shapira B., Recommender Systems Handbook, Springer, 2011.
3. Bobadilla J., Ortega F., Hernando A., Gutiérrez A., “Recommender Systems Survey”, Knowledge-Based Systems, 2013, 46, 109–132.
4. Sharma L., Gera A., “A Survey of Recommendation System: Research Challenges”, International Journal of Engineering Trends and Technology, 2013, 4 (5), 1989–1992.
5. Zhao X., Huang W., Zhang Y., “Embedding in Recommender Systems: A Survey”, arXiv preprint arXiv:2310.18608, 2023.
6. Ren X., Wei W., Xia L., Huang C., “Self-Supervised Learning for Recommendation Systems: A Comprehensive Survey”, arXiv preprint arXiv:2404.03354, 2024.
7. Li A., “Personalized Learning Path Recommendation Based on Artificial Intelligence”, Electronics, 2026, 15 (1), 1–15.
8. Bao S., Wang J., “Personalized Recommendation System Based on Multimodal Knowledge Graphs”, Future Computing and Informatics Journal, 2025.
9. Yang X., “Hybrid Deep Learning-Based Personalized Recommendation for Distance Education”, Journal of Educational Technology, 2025, 18 (3), 45–56.
10. Bijwe R., “AI-Driven Hybrid Recommender Framework for Course Recommendation”, FutureTech Journal, 2026, 6 (2), 77–86.
11. Lin Y., Feng S., Lin F., “Adaptive Course Recommendation in Massive Open Online Courses”, Knowledge-Based Systems, 2021, 212, 106–118.
12. Gandhi S.S., “Research Paper Recommendation System Using Natural Language Processing”, Technical University of Munich, Germany, 2023.
13. Bi X., “NLP-Based Recommendation System Using Sentence Transformers”, University of Vaasa, Finland, 2023.
14. Li S., “Personalized Learning Recommendation Based on Optimized Collaborative Filtering”, Scientific Programming, 2025, 2025, 1–12.
15. Zhang H., “Course Recommendation System Using Large Language Models”, arXiv preprint arXiv:2412.19312, 2024.
16. Wu Z., “Transformer-Based Intelligent Recommendation System in Education”, Proceedings of the ACM International Conference on Educational Data Mining, 2024.
17. Kulkarni S., “Personalized E-Learning Recommender System Using Autoencoders”, International Journal of Artificial Intelligence Research, 2023, 7 (1), 14–22.

18. Devlin J., Chang M., Lee K., Toutanova K., “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”, Proceedings of NAACL-HLT, 2019.
19. Reimers N., Gurevych I., “Sentence-BERT: Sentence Embeddings Using Siamese BERT Networks”, Proceedings of EMNLP, 2019.



Licensed under [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/)