iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

TEXT2COURSE: AN END-TO-END LLM-POWERED FRAMEWORK FOR COURSE GENERATION

A. Srikanth, P. Manogna, V. Ashwini
Final Year Students, Department of Artificial Intelligence & Data Science,
J.B. Institute of Engineering and Technology (UGC Autonomous), Hyderabad, Telangana, India.

ABSTRACT
The existing systems have failed to offer personalised interactive learning experiences from static content in textbooks and documents. It is difficult for users to gain insights, create practice materials and evaluate their comprehension level while the available platforms lack multimodal and adaptive learning capabilities. TEXT2COURSE is an end-to-end LLM-powered framework capable of converting user input and uploads into interactive course material. In particular, this system automatically creates summaries, questions answers pairs, MCQs, explanatory content and graphics, thereby fostering an active learning experience. This framework incorporates triple verification that enables reliable content generation through input validation, model-based cross-referencing, and alignment of generated outputs within context. Its framework entails various modules of content processing, generation, visualisation, multimedia integration, and fast content access via QR codes. With the help of NLP and generative AI, the proposed framework facilitates adaptive learning experiences in a context-aware manner. Content personalisation according to user inputs is also possible in this regard. Text2course framework is built on React, Next.js, Tailwind CSS, and Python frameworks that ensure its scalability, responsivity, and high performance. Experiments demonstrate improved learning efficiency owing to automated creation of study materials and enhanced engagement through interactive and multimodal content.
Keywords: Large Language Models, Automated Course Generation, Intelligent Tutoring Systems, Natural Language Processing, Adaptive Learning, Multimodal Learning, Quiz Generation, Educational Technology.

INTRODUCTION
Rapid adoption of digital education has increased the accessibility of learning resources; however, most of the currently available technologies are still insufficient to turn passive learning materials into active experience for users. Most often, students rely on textbooks, PDF, and notes in which they should make significant efforts to identify main ideas, produce questions and answer them, and ensure comprehension of the studied material. Besides, there are no tools and methods to customise this process according to individual needs of the user.

Current e-learning solutions lack a wide range of functionalities necessary to increase interactivity and adaptability. Though some programs may have options for quiz completion or watching videos, few platforms use various approaches to teach, such as diagrams, summaries, real-time testing, etc. Lack of an intelligent way to respond to any queries makes these platforms less effective since users cannot benefit from them fully. TEXT2COURSE is an end-to-end solution based on large language models (LLMs) that can transform any input text, textbook, document into structured, personalised, and interactive learning resources.It operates as an intelligent tutor that produces summaries, explanations, quiz questions, and assessments. The system converts learning materials from passive forms into an active learning environment.

One of the innovations of the suggested system lies in its triple verification method that guarantees the validity and relevancy of the produced content. Specifically, the process of producing content involves validation of the input query or document, cross-validation via the language model, and context alignment of the output data. Thanks to this three-step procedure, the quality of the created learning materials increases. The system comprises a set of interrelated modules that together create a comprehensive learning experience. These include the module of content processing that deciphers input messages, the module of generation that provides answers to questions as well as explanations, and the module of visualisation that uses diagrams and emoji flowcharts to make difficult information clear. Also, there is the multimedia module with videos for learning and a QR module for interacting with generated content.Modern technologies such as natural language processing, generative artificial intelligence, and

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

adaptive learning techniques have been used for the development of the system to offer customised and context-aware learning solutions. The system was built using the technologies of React, Next.js, Tailwind CSS, and Python to ensure scalability and efficient functioning. Through automation of generation of structured content and offering multimodal solutions, TEXT2COURSE can enhance learning efficiency considerably.

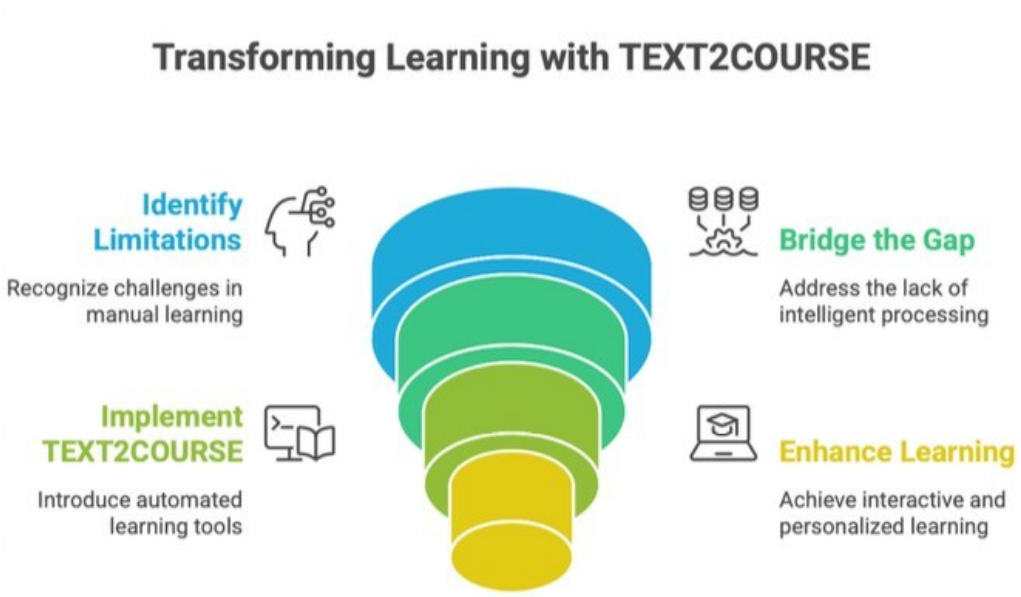

Figure 1: Text2Course System Overview

OBJECTIVES

The primary objective of this research is to develop and deploy a comprehensive and intelligent, end-to-end framework for transforming static content and user queries into educational courseware using large language models.

The key objectives of the proposed study include the following:

- The development of an LLM-based solution for automatically generating summaries, explanations, and learning modules from textbooks, PDF documents, and user queries.
- The use of a triple check system to ensure the accuracy, reliability, and contextual relevance of the generated educational content.
- The design of module-based components, including the content processing module, content generation module, visualisation module, multimedia module, and QR module.
- Automatic generation of different types of assessments, including multiple choice, descriptive question type, and answers evaluation.
- Multi-modal education by including features like visual diagrams, emoji flowcharts, and video explanations.
- Support for different kinds of inputs like text files and PDF documents.
- Adaptive learning to personalise content on the basis of user interactions.
- The deployment of a scalable full-stack solution by leveraging technologies such as React, Next.js, Tailwind CSS, and Python.
- QR-based access to generated content for quick access and sharing

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

METHODOLOGY

The methodology for the TEXT2COURSE framework involves building an efficient and flexible pipeline that will take the raw educational input to generate structured and interactive educational outputs. The process involves user inputs provided in the form of natural language queries or uploaded documents such as PDF and txt documents. The input will be used to extract semantic knowledge using NLP techniques. The use of large language models helps generate the output based on summaries, explanations, and assessments. In order to improve the validity of the generated output, a three-level validation method has been proposed.

The input validation will ensure that the input received is valid and will help in avoiding irrelevant outputs being generated. At the same time, there will be a cross-validation step where the output is validated against the contextual information present in the language model. Moreover, the framework includes several modules such as visualisation, multimedia, and interaction via QR. The different modules collaborate with each other and help in achieving the goal of transforming the static inputs into structured course materials.

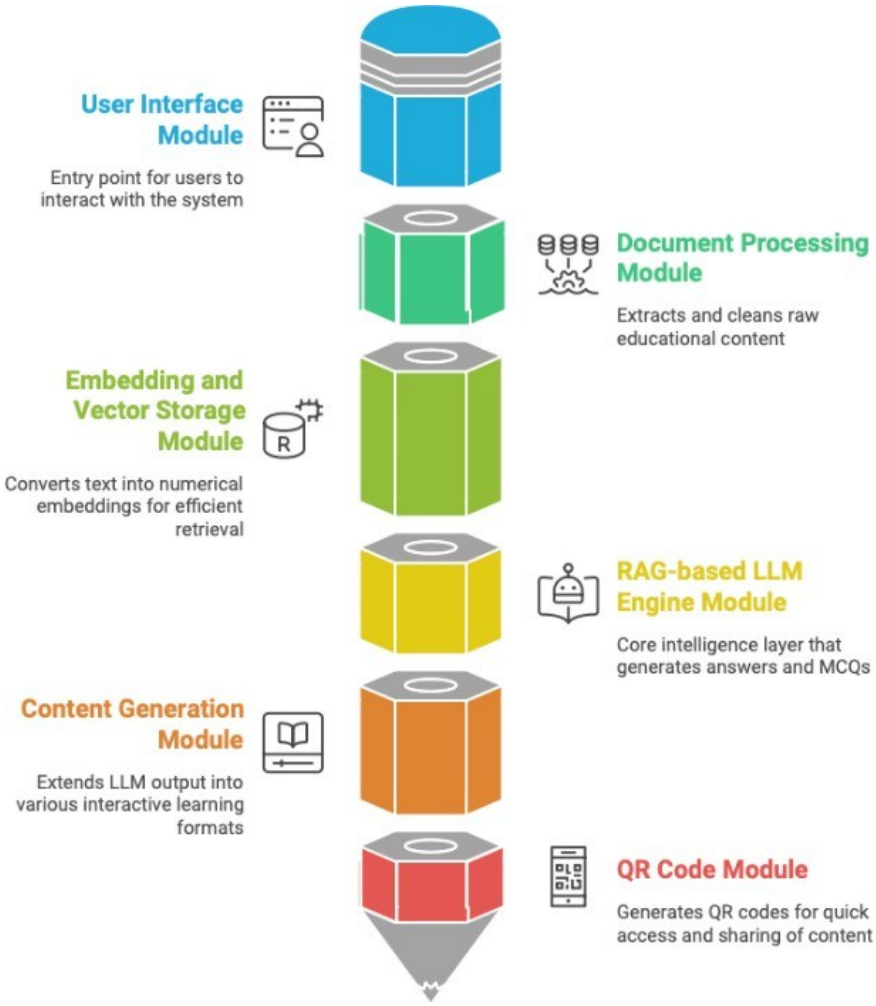

Figure 2: Text2Course Model Organization

A. User Interface Module:

The first module is where the system begins and serves as the interaction gateway to allow users to easily communicate with the system. Users will be able to type their queries or upload files like pdf files and text files using this module that facilitates communication between the user and other parts of the system.

B. Document Processing Module:

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

This module is utilised to extract and clean up the data after receiving the input from the user. It will remove any noise and unnecessary data that can interfere with the extraction process. This module uses natural language processing techniques in order to identify the important data for analysis and generation of outputs.

C. Embedding and Vector Storage Module:

Here, the extracted texts are transformed into numeric values called embeddings. These embeddings carry important semantics and relationships between words or sentences in the document. The data in this stage is stored in a vector database in order to facilitate the process of finding relevant data for queries.

D. RAG-based LLM Engine Module:

This module acts as the intelligence hub of the system. It uses a Retrieval-Augmented Generation (RAG) mechanism. In this module, the information is extracted from the vector database, and it is combined with the capabilities of large language models to produce contextually relevant responses. The answers, explanations, and multiple-choice questions are generated using this module based on retrieved information.

E. Content Generation Module:

In this module, the outputs of the LLM engine are processed further. It involves converting answers to more useful learning outputs in the form of summaries, descriptions, questions, etc. In this way, it provides learning content for the users which will help the users to learn effectively.

F. QR Code Module:

In the final module, QR code links to the generated outputs will be created. These QR codes will allow the user to easily get access to the educational material that has been created.

In this way, it becomes easier for the user to study through any medium as per his convenience.

G. System Architecture and Technology Stack

The System Runs On These Technologies:

Component	Technology
Frontend Framework	Next.js with React
UI Structure	HTML5
Styling	CSS3 (with optional Tailwind CSS)
Backend Runtime	Node.js
Backend Framework	Node.js (Express.js for APIs)
AI & ML Processing	Python (NLP, LLM integration, RAG pipeline)
Document Processing	Python (PDF/Text extraction libraries)
Embedding & Storage	Python (Vector embeddings, vector database)
API Communication	REST APIs (Node.js ↔ Python integration)
Content Generation	Python (LLM-based generation modules)
QR Code Generation	Node.js / Python (QR libraries)

Table 1: Technology Stack ofText2Cours

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

RESULTS AND DISCUSSION

The evaluation of the performance of the TEXT2COURSE system reveals that the framework works efficiently and stably, as demonstrated by the relatively high total pass rate and the ability to provide consistent and timely educational material using the developed framework. The core components such as authentication, PDF manipulation, answer generation, quiz generation, visualisation, and QR-code generation were evaluated comprehensively. Most modules worked successfully more than 80% of the time, demonstrating the effectiveness of using Node.js, Python, and processing based on LLMs when developing educational applications.

As for the issues observed while running the tests on the system modules, one can note that most of them are rather technical and have a limited impact on the overall system operation. For example, there were minor problems with UI validation in the authentication module, but PDF manipulation showed minor performance problems. RAG-based answer generation is efficient in generating answers based on context, but there are some difficulties with interpreting highly complicated queries. In their turn, quizzes and visualisations usually perform efficiently, although they may have some randomising and prompt comprehension issues.In conclusion, the outcome shows that the TEXT2COURSE system is quite reliable and effective. It is evident from the fact that the number of test cases passed is higher than the number of failed ones, thus proving how reliable the architecture of the program is, as well as how effectively implemented technologies work. In order to improve the system further, some improvements may include optimisation of edge cases, as well as accuracy when processing complex inquiries. Overall, despite some disadvantages, the program manages to fulfill its purpose effectively.

Figure 3: Achieving Text2CourseSystem Success

Test Category	Total Test Cases	PASS	FAILS	REMARKS
Authentication Module	10	9	1	Secure login working, minor UI validation issue in error message handling
PDF Upload & Processing	15	13	2	File upload stable, occasional delay with large PDFs
Answer Generation (RAG + LLM)	20	17	3	Good contextual answers, some ambiguity in complex queries
Quiz Generation Module	15	12	3	MCQs generated correctly, minor issues in answer randomness

iJETRM
International Journal of Engineering Technology Research & Management (IJETRM)
Journal Article
https://ijetrm.com/issue/

Image & Flowchart Generation	10	8	2	Good visual output, occasional mismatch in complex prompts

Table 2: Test Case Summary

ACKNOWLEDGEMENT

Here, the writers thank the faculty of their department for support and advice during the course of developing this project.

The writers especially wish to thank their project guide and department head for giving valuable technical inputs as well as access to the facilities of the institution required for the purpose of the experiment. The writers would also like to thank all the users who participated in the user acceptance testing process.

CONCLUSION

Thus, the TEXT2COURSE framework can be described as a unique solution that converts regular learning material into a set of interactive and personalised educational resources through an automated process. It successfully compensates for the disadvantages of typical learning platforms by using several advanced methods of generating structured study content.

One of the distinctive features of the system is the triple verification of all generated material, which allows ensuring the accuracy and consistency of the learning materials. A highly flexible structure with modules related to content processing, generation, visualisation, multimediatisation, and interaction based on QR code technology allows for a collaborative creation of educational resources in one platform. It is also essential to note the presence of multimodal elements that make the process of generating and analysing material more efficient. Specifically, the proposed framework includes such elements as diagrams and flowcharts that help understand concepts better and even some videos that may enhance the learning experience even further.

Thanks to modern programming languages such as React, Next.js, Tailwind CSS, and Python, the framework ensures efficiency and high speed of work.

REFERENCES

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville, Deep Learning, MIT Press, 2016.
[2] Tom B. Brown et al., “Language Models are Few-Shot Learners,” NeurIPS, 2020.
[3] Lewis et al., “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” 2020.
[4] Vaswani et al., “Attention Is All You Need,” NeurIPS, 2017.
[5] Christopher Manning, Foundations of Statistical Natural Language Processing, MIT Press, 1999.
[6] Jurafsky & Martin, Speech and Language Processing, Pearson, 2023.
[7] OpenAI, “GPT Models Documentation,” https://platform.openai.com, 2024.
[8] Hugging Face Documentation, “Transformers Library,” https://huggingface.co/docs, 2024.
[9] Pinecone Docs, “Vector Database for AI Applications,” https://www.pinecone.io/docs, 2024.
[10] FAISS Library Documentation, “Efficient Similarity Search,” https://faiss.ai, 2023.
[11] TensorFlow Developers, “Machine Learning Framework Guide,” https:// www.tensorflow.org, 2023.
[12] PyTorch Team, “Deep Learning with PyTorch,” https://pytorch.org, 2023.
[13] spaCy Documentation, “Industrial NLP Library,” https://spacy.io, 2024.
[14] NLTK Project, “Natural Language Toolkit,” https://www.nltk.org, 2023.
[15] Scikit-learn Developers, “Machine Learning in Python,” https://scikit-learn.org, 2023.
[16] Django Software Foundation, “Django Web Framework Documentation,” https: docs.djangoproject.com, 2024.
[17] Flask Documentation, “Micro Web Framework for Python,” https:// flask.palletsprojects.com, 2023.
68[18] PostgreSQL Global Development Group, “PostgreSQL Documentation,” https:// www.postgresql.org/docs, 2024.
[19] MongoDB Inc., “MongoDB Manual,” https://www.mongodb.com/docs, 2024.
[20] OpenCV Documentation, “Computer Vision Library,” https://docs.opencv.org, 2023.