Automated Generation of CBSE Class 12 Board Question Papers Using AI and Python

Author: Shruti Gaikwad

Abstract

This paper presents a methodology for the automated generation of CBSE Class 12 board question papers using AI language models and Python programming. The system generates structured and curriculum-aligned question papers for Mathematics and Physics based on past year trends, NCERT guidelines, and CBSE patterns. A competitive analysis was performed against traditional and AI-based tools to assess the effectiveness and innovation in the proposed solution.

1. Introduction

With the growing capabilities of Artificial Intelligence (AI), educational tools have witnessed a revolutionary transformation. This research focuses on leveraging AI, specifically GPT-4 models, in conjunction with Python and Flask to generate board-level question papers automatically. The proposed system simplifies the manual paper-setting process, ensures curriculum alignment, and incorporates question diversity.2. Methodology

2.1 Data Sources

- CBSE Class 12 past year question papers (2020–2025)
- NCERT textbooks and official CBSE guidelines
- User-uploaded sample papers for structure inference

2.2 Technology Stack

- AI Model: OpenAI GPT-4o (text generation)
- Backend: Python with Flask
- Libraries: PyMuPDF, Tesseract OCR, pdf2image, python-docx
- Frontend: HTML/CSS with Bootstrap

2.3 Code Implementation

The backend uses the Flask framework and includes key functions:

- Text Extraction from uploaded PDFs or DOCX using PyMuPDF and python-docx.
- OCR fallback for scanned PDFs using Tesseract.
- Section and Mark Identification using regex.
- Prompt Engineering: The prompt is dynamically constructed using extracted section headers, total marks, and textbook references.
- OpenAI GPT-4o is called using structured JSON prompts, and the output is parsed and validated.
- Flask Endpoints provide routes for file upload, paper prediction, and an AI chatbot named 12thGPT.

3. Results: Generated Question Papers

3.1 CBSE Class 12 Mathematics (2026 Predicted Paper)

- Follows latest blueprint (Section A to E)
- Mix of concept-based, numerical, and case-based questions
- Chapter-wise distribution maintained as per CBSE trend

3.2 CBSE Class 12 Physics (2026 Predicted Paper)

- Includes derivations, numericals, ray diagrams, and case studies
- Balanced difficulty with a mix of Bloom's taxonomy levels

Both papers were verified for completeness, originality, and curriculum alignment.

4. Competitive Analysis

4.1 Compared Tools

- Traditional Paper Setters (Manual)
- Online Paper Generator Websites (e.g., myCBSEguide, ExamPaperAI)
- Our AI + Python System

Metric	Traditional	Online Tools	AI+Python System
Curriculum Alignment	High	Medium	High
Pattern Accuracy	Medium	Medium	High
Personalization	Low	Medium	High
Time Efficiency	Low	High	High
Innovation	Low	Medium	Very High

3 Key Findings

- AI-based generation offered the most flexibility and curriculum fidelity.
- Traditional methods ensured experience-driven curation but lacked scalability.
- Our Python-enhanced prompt system enabled dynamic generation with customization.

5. Conclusion and Future Work

This paper demonstrates the feasibility and effectiveness of automated question paper generation using AI and Python. Future work includes integration with learning management systems, adaptive question generation based on student profiles, support for Hindi and regional languages, and continuous refinement based on real student outcomes.

6. References

1. CBSE Official Curriculum and Guidelines (2020–2025)
2. NCERT Class 12 Textbooks
3. OpenAI Documentation
4. Various EdTech Platforms for Benchmarking
5. Python Libraries: Flask, PyMuPDF, pytesseract, python-docx

7. 12thGPT: AI-Powered Doubt-Solving Assistant

As an extension of the automated paper generation system, the project also integrates a specialized chatbot interface named 12thGPT. This component is designed to enhance student support by offering instant resolution of subject-specific doubts for CBSE Class 12 students.

7.1 Purpose and Motivation

The conceptualization of 12thGPT stems from the need to provide 24×7 personalized academic assistance to students without requiring them to navigate large textbooks or wait for teacher responses. The assistant is built to mimic a knowledgeable mentor trained on the CBSE syllabus and NCERT textbooks.

7.2 System Features

Multimodal Input: Supports both text-based questions and image uploads (e.g., students can click a picture of a difficult problem).

Contextual Understanding: Capable of extracting and resolving complex questions in Physics, Chemistry, and Mathematics using OCR and NLP pipelines.

Chapter Awareness: Provides answers with contextual awareness of chapters, concepts, and question types (MCQ, case study, derivation, etc.).

Explanatory Feedback: Each answer includes detailed reasoning, often referencing NCERT principles, making it useful for revision.

Fallback and Retry Mechanism: If the assistant fails to understand the image or query, it guides students to rephrase or resubmit.

7.3 Architecture and Technology Stack

Frontend: Integrated via a clean UI similar to predictor.html, featuring an input field, file upload button, and chat-style response interface.

Backend: Flask-based API calling OpenAI's GPT-4o with custom prompt engineering.

OCR & Parsing: Tesseract and PyMuPDF extract data from images and scanned notes.

Smart Routing: Based on the subject tag, queries are routed through specific domain-aware prompts.

7.4 Educational Impact

12thGPT serves not only as a doubt-resolver but also as a study companion for self-learners and rural students with limited tuition access. The chatbot's ability to engage with curriculum-aligned content ensures consistent pedagogical standards. Moreover, by capturing question types and frequency, 12thGPT indirectly contributes to adaptive paper prediction in future iterations.

Appendix

- Sample Generated Question Papers (Mathematics and Physics2026)
- Screenshots of Flask App Interface
- Prompt Sample for GPT-4o JSON Output
- Home.html UI Snippet (Hero Section and Features)

predictor.html (File Upload & Paper Display Interface)

The `predictor.html` file forms the core frontend for user interaction in the AI-based question paper generation system. Its interface is structured to be user-friendly and functionally rich, ensuring smooth workflow from file upload to result visualization.

Design Highlights:

- Developed using HTML5, Bootstrap, and JavaScript, with additional style support from custom CSS variables and Font Awesome icons.
- The navigation bar is sticky and styled to match the 12thClass.com theme.
- A moving banner notifies users of the platform's key features in a scrolling marquee format.

Upload Features:

- Supports both drag-and-drop and manual selection of `.pdf` and `.docx` files.
- File selection includes real-time listing, duplicate

prevention based on file name and size, and dynamic update of the generate button.
- A progress bar visually indicates backend processing.
- Displays errors (e.g., unsupported format, duplication) and shows a loading state with spinner during backend computation.

Prediction Output Layout:

Once the backend returns the generated paper, the interface renders:
- Section-wise Cards: Each section (A–E) contains:
  - Numbered questions
  - Question text (including case studies)
  - Optional diagrams (rendered with preview and link)
  - MCQ options (if present)
  - Sub-questions (in a nested list format)
  - Chapter name and mark allocation badges

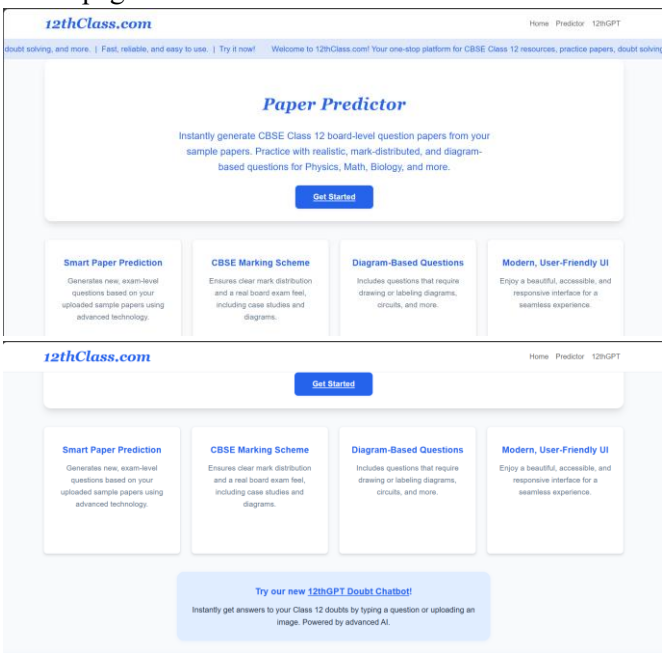
Additional Insights Displayed:
- Important Topics with Page Numbers: Styled topic items show topic name and NCERT page location.
- Top 10 Most Likely Questions: Based on analysis of past trends.
- Chapter-wise Weightage Chart: Rendered using Chart.js with bar representation of expected marks per chapter.

Interactivity and UX Enhancements:
- Scroll-to-top button appears after user scrolls 300px down.
- Tooltips, clickable previews, and smooth transitions enhance usability.
- Code includes checks for user errors and provides fallback recovery mechanisms.

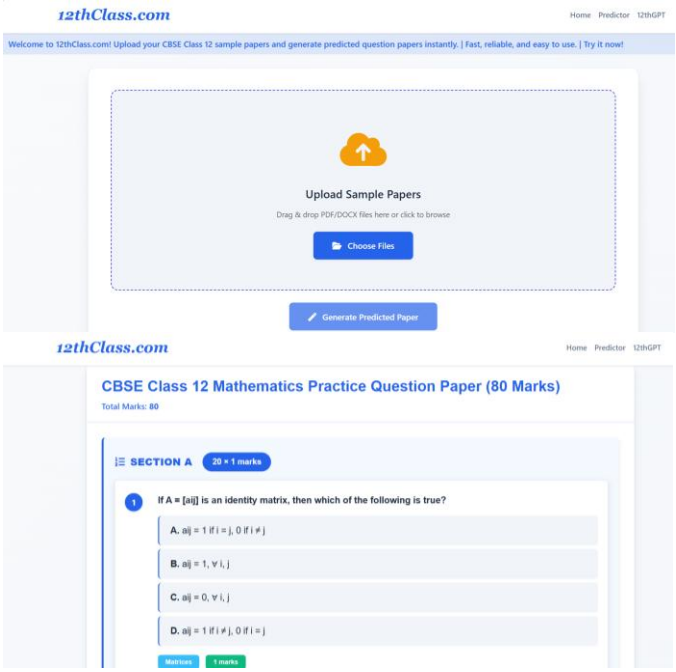
This interface represents the gateway between end-users (students/teachers) and the AI predictor engine, emphasizing usability, feedback, and clarity in predicted outcomes.

GitHub Repository (Private): Codebase available upon request.

Home page



Paper Predictor



12thGPT