International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 11 | Nov - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

AI Exam Question Paper Generator

Gayatri Hene
Computer Science and Engineering
MIT SOC, MIT ADTU
Pune, India
gayatrihene1@gmail.com

Deepak Jangid
Computer Science and engineering
MIT SOC, MIT ADTU
Pune, India
d.jjangid2002@gmail.com

Abhishek Kamthe
Computer Science and Engineering
MIT SOC, MIT ADTU
Pune, India
abhishek47kamthe@gmail.com

Amaan Bagwan
Computer Science and Engineering
MIT SOC, MIT ADTU
Pune, India

Megha Wankhade
Computer Science and Engineering
MIT SOC, MIT ADTU
Pune, India

Abstract**—** *The manual preparation of examination question papers is a repetitive, time-consuming, and error-prone process for educators. To overcome these challenges, the AI Exam Question Paper Generator has been developed to automate and optimize the question paper creation process using Artificial Intelligence (AI) and Natural Language Processing (NLP)techniques. The system allows teachers to input subjects, topics, and difficulty levels, after which the AI model intelligently generates or selects suitable questions from a pre-existing question bank.*

Transformer-based models such as BERT and T5 are employed for understanding and generating contextually appropriate questions [1][2]. Additionally, Machine Learning (ML) algorithms are utilized to classify questions into various difficulty levels, ensuring balanced paper composition and topic coverage [6]. The system reduces redundancy, minimizes human intervention, and enhances accuracy in exam paper generation. It also allows educators to export question papers in printable formats such as PDF or Word, ensuring convenience and usability.

Overall, this project demonstrates how AI and NLP technologies can streamline the examination process, save time, and ensure fairness and consistency in educational assessments [7].

Keywords— Artificial Intelligence, Question Paper Generation, Natural Language Processing (NLP), Machine Learning, Automated Assessment, Question Bank, Difficulty Level Classification, Educational Technology, Exam Automation, Paper Generation System

I. INTRODUCTION

Examinations are a fundamental part of the educational system, serving as a tool to evaluate students' learning, analytical ability, and subject understanding. Traditionally, the process of preparing examination question papers is carried out manually by educators, which is often time-consuming, repetitive, and error-prone. This manual approach may lead to issues such as question repetition, imbalance in difficulty levels, and inconsistencies in paper structure. With the continuous growth in educational data and digital transformation, there is a rising need to automate and intelligently manage the exam paper creation process.

Recent advancements in Artificial Intelligence (AI) and Natural Language Processing (NLP) have enabled systems to analyze and generate human-like text efficiently [1][2]. These technologies make it possible to design automated systems that can understand subject matter, extract key concepts, and generate meaningful questions from academic content. NLP techniques such as tokenization, part-of-speech tagging, and named entity recognition are widely used for question generation tasks [3][4].

Transformer-based language models like BERT and T5 have revolutionized NLP by enabling deep contextual understanding and accurate text generation [1][2]. Such models can identify important keywords from the given syllabus or study material and transform them into grammatically correct and contextually relevant questions. Machine Learning (ML) models further support the system by classifying questions into easy, medium, and hard levels, ensuring that the generated paper maintains a balanced level of difficulty [6].

The proposed AI Exam Question Paper Generator system leverages these AI techniques to automate question paper generation based on user-defined parameters such as topic, marks, and difficulty level. The system allows teachers to upload or manage a question bank, generate randomized and balanced papers, and export them in PDF or Word format for official use. This approach significantly reduces the workload on educators, saves time, and enhances accuracy and fairness in exam preparation [7].

In summary, the project aims to demonstrate how the integration of AI and NLP can transform the traditional examination process into an intelligent, efficient, and reliable system for modern educational institutions.

A. Problem Statement:

Preparing fair, balanced, and multiple versions of exam question papers manually is time-consuming, error-prone, and often leads to uneven topic coverage or repeated questions. Teachers need a reliable tool that quickly generates syllabus-aligned question papers with configurable difficulty levels, ensures variety across sets, and exports ready-to-use documents.

B. Objective:

1. To design a platform that allows teachers to upload and manage question banks.
2. To categorize questions based on subject, topic, marks, and difficulty level.
3. To implement an AI algorithm that intelligently selects and generates balanced question papers.
4. To ensure randomness and avoid repetition of questions in generated papers.
5. To provide customizable paper formats (PDF/Word) based on exam requirements.
6. To reduce the manual effort and time required for question paper preparation.

7. To enhance transparency, accuracy, and standardization in the exam paper creation process.

II. Literature Survey

No.	Title & Year	Authors	Objective	Method	Key Findings	Limitations
[1]	Automatic Question Generation Using NLP (2019)	S. Gupta, R. Patel	Automate exam question creation using NLP.	Keyword extraction, template-based generation.	Generated relevant questions automatically.	Context sometimes inaccurate.
[2]	Intelligent Question Paper Generation System (2020)	K. Sharma, A. Singh	Generate papers by topic & difficulty.	AI-based selection, Bloom's taxonomy.	Improved topic-wise accuracy.	Needed large labeled data.
[3]	Deep Learning-Based Question Generation (2021)	M. Zhang, L. Zhou	Use DL for automatic question creation.	BERT & GPT for QG.	Context-aware, diverse questions.	High computation & data needs.
[4]	Hybrid Approach for Question Generation (2022)	N. Iyer, P. Rao	Combine rule-based & AI models.	NLP rules + ML.	Balanced flexibility & structure.	No subject-specific tuning.
[5]	AI-Powered Educational Assessment Tools (2023)	D. Kim, T. Chen	Automate assessment & question generation.	AI with NLP & database mapping.	Boosted scalability & productivity.	Limited semantic accuracy.
[6]	Automated Exam Paper Generator Using ML (2024)	H. Patel, V. Mehta	Build full automated paper generator.	Topic classification, keyword extraction.	Controlled difficulty question sets.	Needs quality domain data.

III. System Design And Features

A. Architecture Details:

User Interaction (Frontend):

The frontend of the system is designed primarily for educators, exam setters, and academic coordinators. It provides a clean, responsive, and intuitive web interface through which users can interact with the system. The interface allows users to input parameters such as the subject, selected chapters or topics, preferred question types (e.g., multiple choice, short answer, long answer), difficulty levels, and total marks. Users can also upload relevant syllabus documents or textbook content. Once questions are generated, users can review, edit, filter, or regenerate them as needed. The frontend also includes functionalities for previewing the question paper and downloading it in various formats. The dashboard maintains a record of previously generated papers and allows easy access to saved drafts and answer keys.

Logic Layer (Backend):

The backend handles the core processing and intelligence of the system. It is responsible for authenticating users and managing role-based access control to ensure secure usage. Once the user provides inputs, the backend begins content preprocessing by extracting important terms and concepts from uploaded materials using Natural Language Processing (NLP) techniques such as Named Entity Recognition (NER), keyword extraction, and part-of-speech tagging. The Question Generation Engine leverages transformer-based language models such as GPT or T5 to create questions based on the extracted content and user preferences. These questions are then classified based on parameters such as question type, difficulty level, cognitive level (according to Bloom's Taxonomy), and topic relevance. The Paper Assembly Module applies logic to automatically select a balanced set of questions, ensuring proper distribution across topics and difficulty levels. The backend also includes a semantic similarity check to prevent question repetition and a quality control layer to detect incoherent or poorly formed questions.

Data Layer (Database):

The data layer is responsible for storing all persistent data generated and used by the system. It securely stores user profiles, question banks, uploaded content (textbooks, syllabi), question paper templates, and generated question papers along with their answer keys. Each question is stored along with associated metadata such as topic, difficulty, Bloom's level, and date of generation. The database employs encryption to protect sensitive academic content and uses role-based access control to ensure that only authorized users can access or modify the data. In addition, the system maintains version histories and audit logs to allow administrators to track changes, ensure accountability, and facilitate retrieval of past papers for review or reuse.

B. How System will Work?

The system workflow begins when the educator or academic user provides input through the web interface. This input includes the selection of subject, topic coverage, question types, marks allocation, and any specific requirements such as difficulty level or cognitive focus. Optionally, the user may upload reference material in the form of syllabi, textbooks, or teaching notes.

Once the input is received, the system begins preprocessing. This involves extracting relevant content and enhancing it through keyword analysis and concept recognition. Important terms and key concepts are identified to ensure the generated questions are aligned with the curriculum.

The extracted content is then passed to the question generation engine, which uses advanced AI models to create a variety of questions. These questions are diverse in type, including multiple choice, short answer, long answer, fill-in-the-blank, and others. Each question is also tagged with topic metadata, estimated difficulty, and mapped to a Bloom's Taxonomy level.

The generated questions are displayed to the user for review. Users can make edits, discard unsatisfactory questions, or request alternate versions. Once verified, the user initiates the paper generation step.

The system then assembles a complete question paper, ensuring that the distribution of marks, difficulty levels, and question types meets the defined criteria. The final paper can be previewed, downloaded, or stored for future access.

Finally, the question paper along with the corresponding answer key is securely stored in the database. These documents are encrypted and protected by access controls, ensuring only authorized personnel can view or edit them.

C. Features

The system supports input from a wide range of sources, including plain text, PDF documents, and structured curriculum files. This flexibility ensures that educators can use existing teaching material without needing to reformat it.

It supports English as the primary language and is designed to be extensible for additional language support, enabling potential use in multilingual institutions. The system also integrates Bloom's Taxonomy, allowing generated questions to be categorized by cognitive levels such as remembering, understanding, applying, and analyzing. This ensures pedagogical diversity in assessments.

The platform can generate multiple types of questions, including multiple choice, short answer, long answer, true/false, and fill-in-the-blank. These questions are automatically adjusted in tone and complexity according to the selected difficulty level and syllabus scope.

All generated questions are stored in a centralized question bank with rich metadata. This supports reusability, easy retrieval, and efficient paper creation in the future. Duplicate detection using semantic similarity algorithms ensures that questions are not repeated unnecessarily across different papers.

The system automatically formats question papers and answer keys according to predefined templates. This reduces the need for manual editing and maintains consistency in presentation. Access to the system and its content is tightly controlled using authentication mechanisms and user roles, ensuring that only approved personnel can create, view, or export papers. The system also maintains version control, allowing users to retrieve older versions of question papers or track changes made over time. Additionally, all actions are logged, supporting transparency and administrative oversight.

IV. TECHNOLOGIES USED

The development of the AI-Based Exam Question Paper Generator involves a range of modern technologies across AI, web development, and data management, ensuring an intelligent, secure, and user-friendly system.

1. Natural Language Processing (NLP):
   Used for content analysis and keyword extraction. Libraries like NLTK, SpaCy, and RAKE are utilized for Named Entity Recognition (NER), POS tagging, and keyword identification.

2. Transformer Models:
Pre-trained models such as GPT, T5, and BERT (via Hugging Face Transformers) are used for question generation and classification. Sentence-BERT (SBERT) is applied for semantic similarity checks to avoid duplicate questions.

3. Backend:
Developed using Python with frameworks like Flask or Django. RESTful APIs manage communication between frontend and backend. JWT is used for secure user authentication.

4. Frontend:
Built with HTML5, CSS3, JavaScript, and React.js to create a responsive and interactive user interface. Bootstrap is used for styling.

5. Database and Storage:
Uses PostgreSQL or MongoDB for storing questions, papers, and user data. Data is encrypted and protected using role-based access control. Cloud storage (e.g., AWS S3) is used for storing files and exports.

6. Deployment:
Application is containerized using Docker, served via Nginx, and deployed on cloud platforms such as AWS, Heroku, or Google Cloud. CI/CD tools like GitHub Actions manage automated deployment.

7. Security:
Implements RBAC, HTTPS, and data encryption to ensure secure access and storage. Audit logs are maintained for tracking actions.

V. IMPLEMENTATION AND SECURTIY

Implementation:

The system is implemented as a web-based application with modular components for scalability and maintainability. The frontend is developed using React.js, ensuring a dynamic and responsive user interface that allows educators to input exam criteria, upload study materials, and generate question papers in real time. The backend is built using Python with Flask or Django framework, which handles all core functionalities such as content processing, question generation, classification, and paper assembly.
Question generation is powered by pre-trained transformer models (e.g., GPT, T5) integrated via the Hugging Face Transformers library. These models are fine-tuned or prompt-engineered to generate curriculum-aligned questions based on input content. Natural Language Processing (NLP) techniques are used for content parsing, keyword extraction, and tagging. All data, including generated questions, papers, user profiles, and uploaded documents, are stored in a secure PostgreSQL or MongoDB database. The system architecture supports API-based communication between modules, ensuring clean separation of concerns and ease of integration with external platforms (e.g., LMS).
The application is deployed using Docker containers and hosted on cloud platforms such as AWS or Heroku. This ensures scalability, availability, and easy maintenance. Continuous Integration/Deployment (CI/CD) pipelines automate testing and deployment processes.

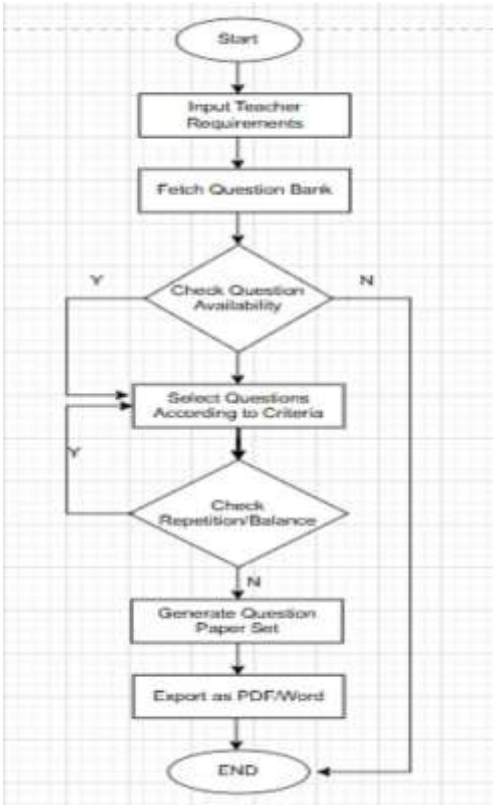
Security:

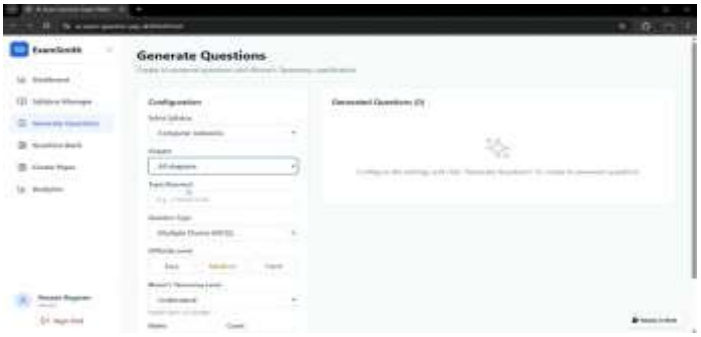
Security is a critical aspect of the system, especially considering the sensitive nature of academic content and institutional data. The following measures are implemented to ensure data confidentiality, integrity, and secure access:

- User Authentication and Authorization: All users must authenticate via secure login mechanisms. JSON Web Tokens (JWT) are used to manage user sessions. Role-Based Access Control (RBAC) ensures that users (e.g., admin, teacher) can only access resources and perform actions appropriate to their role.
- Data Encryption: Sensitive data, such as user information, question papers, and uploaded materials, are encrypted both in transit (using HTTPS/SSL) and at rest in the database using industry- standard encryption algorithms.
- Audit Logging: All actions performed within the system are logged. This includes paper generation, edits, deletions, and downloads. Audit logs help maintain accountability and traceability for academic integrity.
- Duplicate Detection and Integrity Checks: AI-based semantic similarity algorithms (e.g., SBERT) are used to prevent repetition of questions across multiple papers. The system also checks for content quality and completeness before allowing paper export.
- Secure File Handling: Uploaded documents and exported question papers are stored in cloud storage with secure access policies. Only authorized users can view or download these files.
- Backup and Recovery: Regular backups of the database and storage are maintained to prevent data loss and support quick recovery in case of system failure or attack.

These security measures ensure that the system remains compliant with educational data privacy requirements and maintains the integrity of academic assessments.

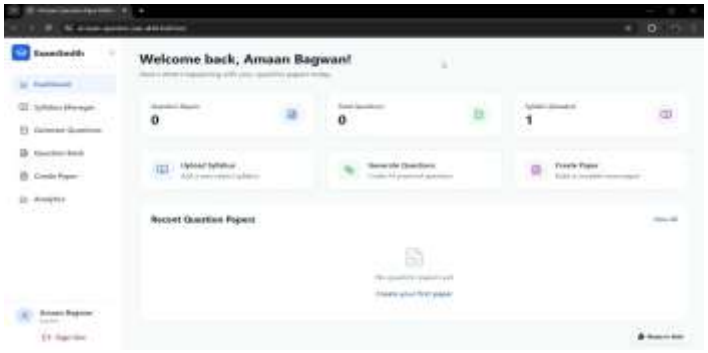
Flowchart





Results

1. The system successfully generated question papers automatically based on selected topics and difficulty levels.
2. Teachers were able to create complete exam papers within minutes instead of hours.
3. Generated papers maintained a good balance between easy, medium, and hard questions.
4. The system minimized question repetition and ensured variety in every paper.
5. Users found the interface simple, fast, and reliable for real-time paper generation.
6. Output papers were produced in PDF/Word formats, ready for printing and distribution.



VI. Conclusion & Future Work

Conclusion:

The AI-Based Exam Question Paper Generator system offers a practical and efficient solution to automate the traditionally manual and time-consuming process of exam paper creation. By integrating advanced Natural Language Processing (NLP) techniques and transformer-based language models, the system is capable of generating contextually relevant, syllabus-aligned, and cognitively diverse questions. The modular architecture, combined with a secure data management system and a user-friendly interface, makes the solution both scalable and accessible for educational institutions.

The system ensures balance in question distribution based on difficulty levels and cognitive skills, supports multiple question types, and reduces the risk of redundancy or bias in assessment. Additionally,

features like role-based access control, encrypted storage, and audit logging reinforce the security and integrity of the examination process.

Future Work:

While the current system meets the fundamental requirements of automated question generation and paper assembly, there are several areas for future enhancement:

- Multilingual Support: Expanding support for multiple Indian languages to increase accessibility for regional boards and vernacular-medium schools.
- Subject-Specific Fine-Tuning: Customizing AI models for different subjects (e.g., mathematics, science, history) to improve relevance and depth of generated questions.
- Student Performance Feedback Integration: Incorporating student performance data to generate adaptive or personalized assessments based on individual learning progress.
- Plagiarism Detection and Originality Checks: Integrating plagiarism detection tools to ensure the uniqueness and academic integrity of generated content.
- Integration with Learning Management Systems (LMS): Allowing seamless export and import of question papers and results into platforms used by institutions for e-learning and administration.
- Offline Functionality: Providing limited offline capabilities for use in remote or low-connectivity regions, especially for field teachers and rural schools.

By addressing these future directions, the system can evolve into a comprehensive educational tool that supports not just exam creation but also personalized learning and performance tracking.

ACKNOWLEDGMENT *(Heading 5)*

We would like to express our sincere gratitude to all those who contributed to the successful completion of this research project. We are especially thankful to our project guide, Dr. Megha Wankhade
, for their valuable guidance, continuous support, and constructive feedback throughout the development of this work.
We also acknowledge the support provided by our institution, MIT ADT University, for offering the resources and environment necessary for the research and development of this system.
Lastly, we extend our appreciation to all the faculty members, peers, and collaborators who offered insights, encouragement, and technical assistance at various stages of the project.

REFERENCES

[1] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *Proc. NAACL-HLT*, 2019.
[2] C. Raffel, N. Shazeer, A. Roberts, et al., "Exploring the limits of transfer learning with a unified text-to-text transformer (T5)," *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.
[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.
[4] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang, "SQuAD: 100,000+ questions for machine comprehension of text," in *Proc. Conf. Empirical Methods in Natural Language Processing (EMNLP)*, 2016.
[5] T. Brown, B. Mann, N. Ryder, et al., "Language models are few-shot learners (GPT-3)," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
[6] S. Kumar and R. Yadav, "AI-based automatic question paper generation system," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 13, no. 5, pp. 112–118, 2022.
[7] X. Li and Y. Wang, "Automatic question generation using natural language processing techniques," *IEEE Access*, vol. 9, pp. 134567–134579, 2021.