

An Intelligent Educational Assessment and Automated Question Paper Generation Framework Using Machine Learning and Generative AI

Mr. Suyog V Tate Patil,

Sakshi Sitaram Kamble, Karan Shankar Kolekar, Aditya Ramesh Jagtap, Sujal Santosh Adate

Department of Computer Science and Engineering (Data Science)

Dr. D. Y. Patil Pratishthan's College of Engineering, Salokhenagar, Kolhapur, Maharashtra, India

Abstract

Creating well-balanced question papers, difficulty control, marking and analysing students' performance are mostly manual and non-integrated in most institutions. Previous work has focused on each of these components of this workflow, with the automatic question generation or performance prediction components being studied separately; few studies have explored them together in a system. This paper proposes an integrated framework that combines (i) Approved Syllabus Content-based Generative AI Question Generation (GAIQG) with teacher's review, (ii) Structured question bank with subject-wise, topic-wise and difficulty-wise organisation, (iii) Blueprint-driven automatic question-paper generation formulated as a constrained selection problem, (iv) Automatic assessment and topic-wise analytics, and (v) Machine Learning models for learning-level classification and adaptive testing based on Item Response Theory. The architecture is explained as a nested structure including access control based on roles and a teacher-in-the-loop gate. As a feasibility check for the analytics components, we run two experiments on synthetic data. A SVM (RBF) classifier achieves 78.0% on test set and macro-F1 of 0.786 in three-level learning classification on simulated cohort of 1500 students. The ability-estimation error (RMSE 0.40) is the same with adaptive item selection as it is with 29 items (62% fewer). The results obtained are preliminary and will be validated with real institutional data in the following phase.

Keywords: educational assessment, question paper generation, generative AI, machine learning, automated assessment, question bank, adaptive testing, student performance analysis.

I. INTRODUCTION

Assessment is at the heart of the teaching and learning process but the burden of assessment is great. Teachers choose questions from books and past papers, ensure balance of topics and marks, adjust difficulty level based on their experience, read scripts and use their experience of the class to interpret results and identify learning gaps. Interviews with teachers and students revealed that these steps are repetitive, tedious, and hard to maintain for multiple subjects, topics and student groups (Phase-I survey). The assessments are also generic; that is, students are given the same questions, no matter their strengths or weaknesses.

Fortunately, with recent advancements in Machine Learning (ML) and Generative AI (GenAI), parts of this workflow can be automated. Large language models can generate questions from source text [4, 8]; supervised learning can categorize students by their learning levels [10]; and psychometric models like item response theory (IRT) can enable tests to adapt to each student [11, 14]. These abilities are typically explored individually, though. A teacher still needs to traverse between a question-generation tool, a spreadsheet to store the question bank and a separate analytics tool and content is generated without a review step.

This paper proposes and specifies an integrated framework that closes this gap and reports preliminary feasibility experiments. The main contributions are:

- **An integrated architecture** that connects question generation, question bank, paper assembly, assessment, analytics and reporting with a teacher-review gate and role-based access control (Section 4).

- **A formulation of blueprint-based paper generation** as a constrained selection problem over a difficulty-tagged question bank (Section 4.4).
- **A learning-analytics and adaptive-testing design** combining supervised learning-level classification with IRT-based item selection (Sections 4.6 and 4.7).
- **Preliminary synthetic-data experiments** with graphs that show the analytics pipeline works end to end and indicate the potential gain of adaptive testing (Section 6).

The rest of the paper is organised as follows. Section 2 reviews related work and the research gap. Section 3 formulates the problem. Section 4 presents the framework. Section 5 gives the implementation plan. Section 6 reports the preliminary experiments. Sections 7 to 9 cover the evaluation plan, limitations and ethics, and conclusions.

II. RELATED WORK AND RESEARCH GAP

A. Automatic question generation

The majority of research focused on automation of question generation (AQG) for educational purposes is reviewed by Kurdi et al. [1] and they identified that most of the research focused on question generation while difficulty control and quality evaluation of the generated questions are weak. Last and Danon [2] outline the normal AQG pipeline of questions construction, ranking and evaluating. Both studies inspire us to assign a difficulty level, as well as a validation check, to each generated question, instead of stopping at generation.

B. Artificial intelligence in education

Zawacki-Richter et al. [3] gave an overview on the applications of AI in higher education, revealing that assessment and evaluation, adaptive systems and profiling of students are common applications but that teachers are seldom engaged in the design of such systems. This helps to reinforce our teacher-in-the-loop design, where approved teachers are still accountable for academic decisions, and for content generated.

C. Large language models in education

The opportunities of large language models in creating educational materials, tailoring educational experiences, and assisting in assessment are explored by Kasneci et al., [4] and Tlili et al. [5] alongside potential threats of factual inaccuracies, bias, and reliance. Transformer models [7] and few-shot ability [8] are used to make question generation practical; retrieval-augmented generation [9] can be used as a method for grounding outputs with approved course material. We adopt retrieval grounding and human review to address the reliability concerns raised in these studies.

D. Ethics and responsible use

Holmes et al. [6] call for a community-wide ethical framework for AI in education covering fairness, transparency, accountability and human oversight. These principles are translated into concrete requirements in Section 8: audit logging, role-based access, controlled test data and no automatic final academic decisions.

E. Research gap

At a qualitative level, the provision of what is typical in each line of work is summarized in Table 1. Question generation tools do not control the difficulty of the questions, nor do they produce complete papers; performance-prediction systems do not produce assessments; and general-purpose chat tools do not have a structured question bank, blueprint control and access control. No individual reviewed line of work envelopes generation, trouble management, papers assembly, analysis, analytics and adaptability. This gap is plugged with the proposed framework.

Table 1. Qualitative comparison of typical capabilities. "Proposed" reflects design intent; implementation and validation are Phase-II work. Individual existing systems vary.

Capability	Typical AQG systems	Typical performance-prediction systems	General LLM chat tools	Proposed framework
Question generation from content	Yes	No	Yes	Yes
Difficulty-level management	Partial	No	Partial	Yes
Structured question bank	Partial	No	No	Yes
Blueprint-based paper assembly	No	No	Partial	Yes
Automated scoring and topic-wise analysis	No	Partial	No	Yes
ML learning-level classification	No	Yes	No	Yes
Adaptive assessment	No	Partial	No	Yes (planned)
Teacher review and role-based access	Partial	Partial	No	Yes

III. PROBLEM FORMULATION

Let S be a set of subjects and T the topics of a subject, with syllabus content D authorised by the institution. A question q in the question bank Q is described by its subject, topic $t(q)$, marks m_q , Bloom level $l(q)$ [12], difficulty class $d(q) \in \{\text{Easy, Medium, Hard}\}$ and a quality score $v_q \in [0, 1]$ that combines relevance, validity and teacher approval. A blueprint specifies the total marks M , the topic weights w_t and the difficulty proportions p_d of the desired paper.

The problem has four parts. (i) Populate Q with valid questions from D , using GenAI under teacher review. (ii) Assemble a paper $P \subseteq Q$ that satisfies the blueprint. (iii) Score responses and analyse topic-wise and difficulty-wise performance. (iv) Classify each student's learning level and select subsequent questions that are informative for that student. The design goal is to reduce manual effort while keeping the teacher in control of every academic decision.

IV. PROPOSED FRAMEWORK

A. System architecture

The framework follows a layered design (Fig. 1). Students, teachers and administrators use role-specific interfaces that reach the system only through a secure API gateway responsible for authentication, role-based access control (RBAC) and audit logging. Behind the gateway, five application services cooperate. The GenAI service drafts questions; the teacher review gate approves, edits or rejects them; the

question bank service stores approved questions with their metadata; the paper generation engine assembles papers; the assessment service scores responses; and the analytics service runs ML models and produces reports. Data are kept in a relational database, a content repository with a vector index for retrieval, and a model registry with audit logs.

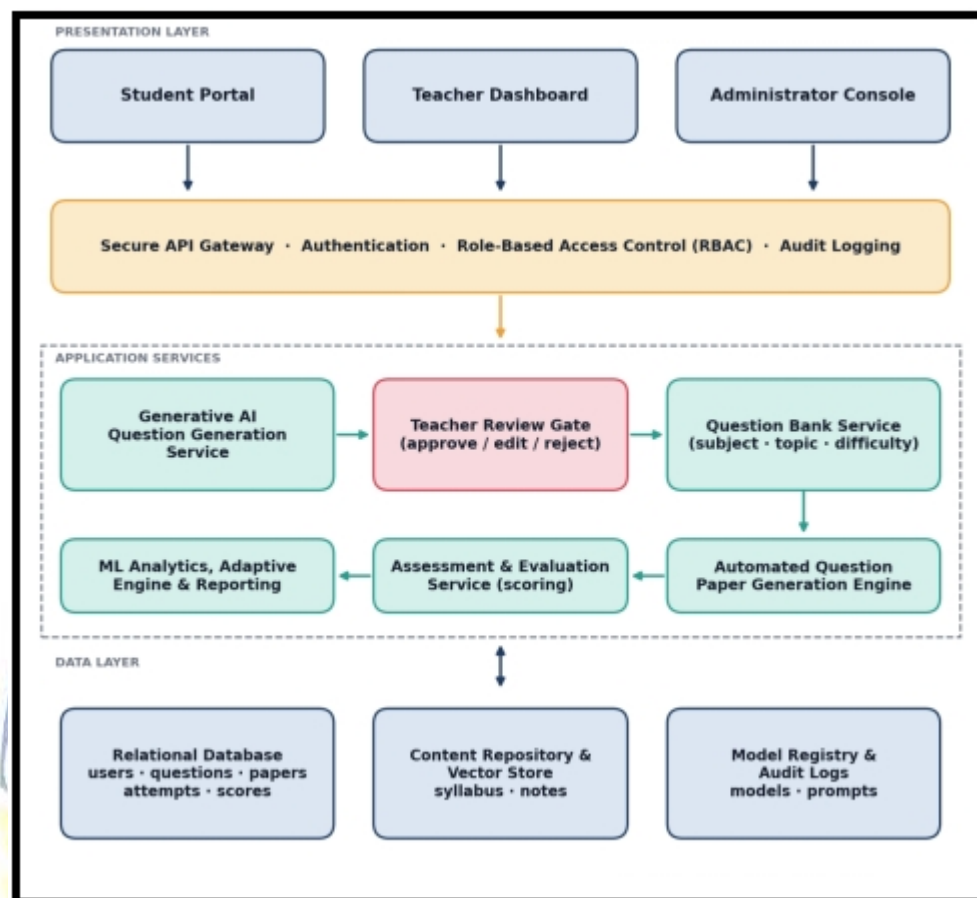


Fig. 1. Layered architecture of the proposed framework. Generated questions reach the question bank only after passing the teacher review gate.

B. Question bank and difficulty management

Each question is stored with subject, topic, marks, Bloom level, question type, answer key, source reference and difficulty class. Difficulty is assigned in two stages. A cold-start label is produced from the Bloom level, question type and a classifier over question features and is confirmed by the teacher. Once students attempt the question, the empirical facility index (the proportion of correct responses) and, when enough data exist, the IRT difficulty parameter b replace the initial label. This lets the bank improve with use and supports the difficulty-wise analytics described later.

C. Controlled Generative AI question generation

Question generation is a controlled pipeline rather than a free-form chat (Fig. 2). Approved syllabus content is chunked and mapped to topics and Bloom levels. For each request, a controlled prompt specifies topic, difficulty, marks and question type, and relevant chunks are retrieved to ground the

model [9]. Automated checks then test relevance to the source, duplication against existing questions using embedding similarity, and consistency of the proposed answer with the source text. Questions that pass are shown to an authorised teacher, who approves, edits or rejects each one; rejected items trigger regeneration with adjusted constraints. Only approved questions enter the bank. This design directly addresses the risk that GenAI produces incorrect, ambiguous or inappropriate questions [4], [5].

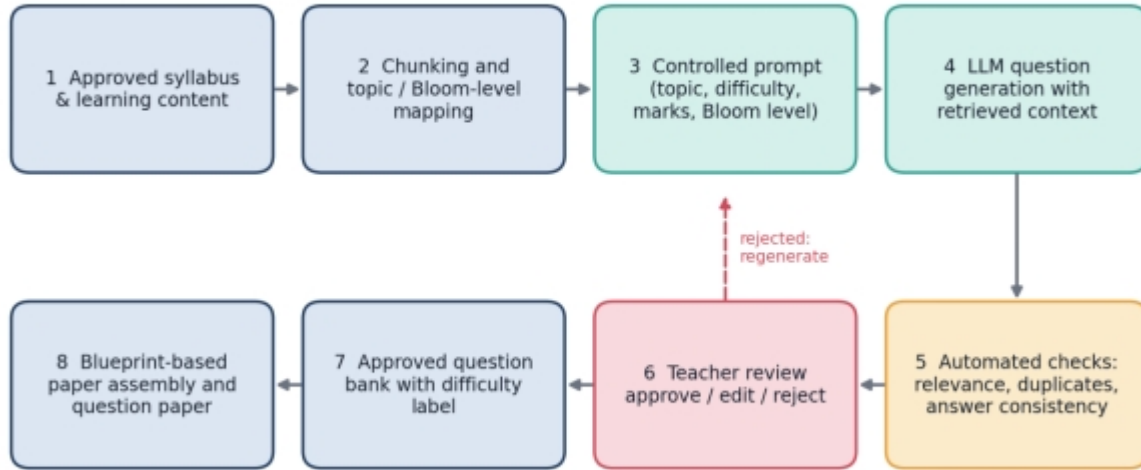


Fig. 2. Controlled question-generation and paper-assembly workflow with automated checks and teacher review.

D. Automated question paper generation

Given a blueprint, paper generation selects questions from the bank. Let $x_q \in \{0, 1\}$ indicate whether question q is included, $Q(t)$ the questions of topic t and $Q(d)$ those of difficulty d . The selection problem is

$$\text{maximise } \sum_{q \in Q} v_q x_q \quad (1)$$

$$\text{subject to } \sum_{q \in Q} m_q x_q = M \quad (2)$$

$$| \sum_{q \in Q(t)} m_q x_q - w_t M | \leq \delta \text{ for every topic } t \quad (3)$$

$$| \sum_{q \in Q(d)} m_q x_q - p_d M | \leq \delta \text{ for every difficulty level } d \quad (4)$$

$$x_q + x_{q'} \leq 1 \text{ whenever } \text{sim}(q, q') > \tau \quad (5)$$

Equation (1) prefers high-quality questions; (2) enforces the total marks; (3) and (4) keep topic and difficulty distributions within a tolerance δ of the blueprint; and (5) prevents near-duplicate questions in the same paper, where sim is embedding similarity and τ a threshold. The problem is a small integer program that can be solved exactly with an ILP solver or approximately with a greedy randomised heuristic when many variants of a paper are required. Randomisation over feasible solutions produces multiple parallel paper sets for the same blueprint. The generated paper is reviewed and finalised by the teacher before use.

E. Assessment, scoring and performance analysis

Student responses are validated and scored automatically for objective items (multiple choice, true/false, short answer with a key); subjective items are flagged for teacher marking, with GenAI-suggested rubrics as an optional aid. From the response records the system computes topic-wise, difficulty-wise and question-wise scores, class averages and score distributions, and highlights strengths, weaknesses and unattempted areas. Results are presented on the Student Dashboard (own history and progress) and the Teacher Dashboard (class, subject and topic views), with exportable reports.

F. Machine Learning for learning-level classification

The analytics service trains supervised models on validated assessment data to classify each student as Low, Medium or High learning level and to expose the factors behind the classification (Fig. 3). Features include topic-wise and difficulty-wise accuracy, previous scores, assignment completion, attendance, time per question, number of attempts and score variability across topics. Candidate models are logistic regression, support vector machines, decision trees, random forests [10] and gradient boosting. Models are evaluated with stratified splits and cross-validation using accuracy, macro-averaged F1 and quadratic weighted kappa, and feature importance is reported so that teachers can see which skills drive a prediction. Outputs are decision-support information and never automatic academic decisions.

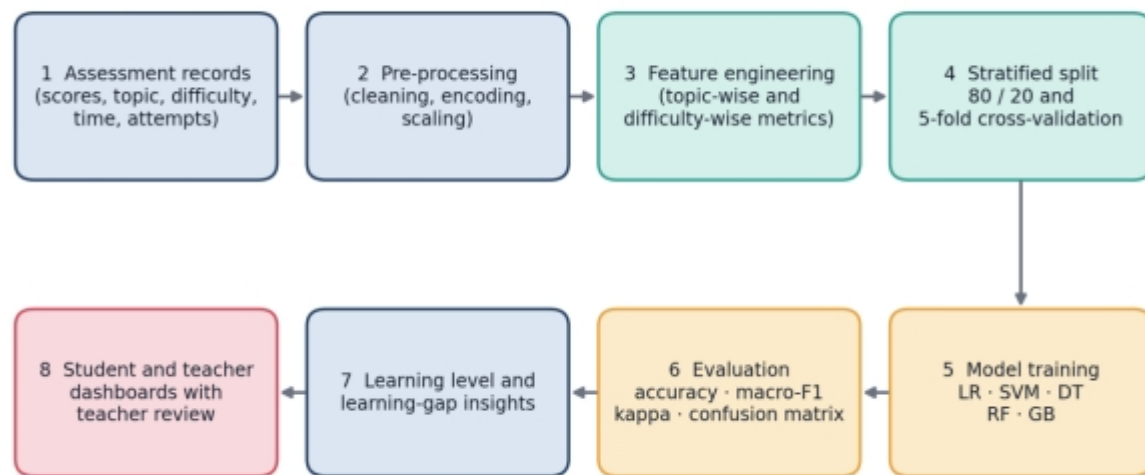


Fig. 3. Machine Learning pipeline for learning-level classification and learning-gap insights.

G. Adaptive assessment with item response theory

To personalise assessment, each question has IRT parameters: discrimination a_j and difficulty b_j . Under the two-parameter logistic model [11], the probability that a student of ability θ answers item j correctly is

$$P_j(\theta) = 1 / [1 + \exp(-a_j(\theta - b_j))] \quad (6)$$

The information an item provides about ability is

$$I_j(\theta) = a_j^2 P_j(\theta) [1 - P_j(\theta)] \quad (7)$$

and is largest when the item difficulty is close to the student's ability (Fig. 4). The adaptive engine keeps an expected-a-posteriori (EAP) ability estimate,

$$\hat{\theta} = \sum_k \vartheta_k L(\vartheta_k) g(\vartheta_k) / \sum_k L(\vartheta_k) g(\vartheta_k) \quad (8)$$

where L is the likelihood of the responses so far and g a prior over a grid of ability values θ_k . After every response the estimate is updated and the next item is the unused one with maximum information at the current estimate [14]. Adaptive selection needs calibrated item parameters, so it is scheduled after enough response data have been collected, and exposure control is added so that popular items are not over-used.

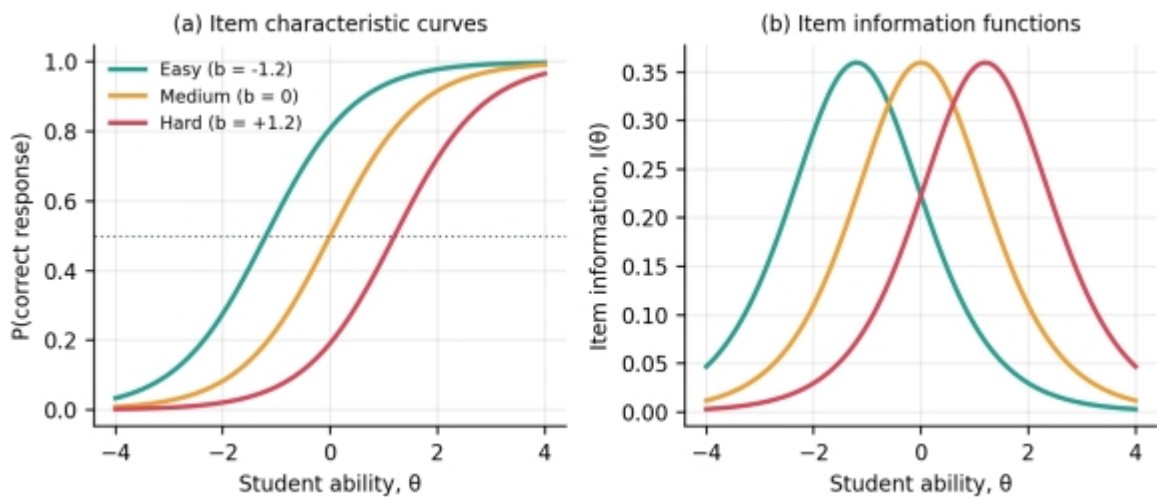


Fig. 4. Two-parameter IRT model for three items of equal discrimination ($a = 1.2$): (a) item characteristic curves; (b) item information functions. Each item is most informative near its own difficulty.

Table 2. Mapping of project objectives to framework modules.

ID	Objective	Framework module	Section
O1	Intelligent educational assessment workflow and assessment records	Assessment and evaluation service	4.5
O2	Student and teacher dashboards	Presentation layer, reporting	4.1, 4.5
O3	Automated assessment processing and performance analysis	Assessment service, analytics	4.5
O4	Question bank and automated paper generation	Question bank, paper engine	4.2, 4.4
O5	ML models for performance analysis and learning gaps	ML analytics service	4.6, 4.7
O6	Controlled GenAI question generation with teacher review	GenAI service, review gate	4.3

V. IMPLEMENTATION PLAN

Phase-I (July to August 2026) covered problem identification, a school and college survey with teachers and students, requirement finalisation, architecture design and study of the required technologies. Implementation follows in Phase-II (Fig. 5). Core services (authentication, database, and student portal and question bank) are built first because all later modules depend on structured data. The teacher dashboard, evaluation and reports follow then ML analytics and adaptive components, which need validated response data, and finally GenAI generation, integration, security hardening, testing and deployment.

Candidate technologies under consideration are Python with a REST framework such as FastAPI or Django for the backend, PostgreSQL for relational data, a vector index (for example pgvector or FAISS) for retrieval, scikit-learn for ML models [13], a hosted or open-weight large language model for generation, and a web frontend for the portals. The final selection will be made during Phase-II design.

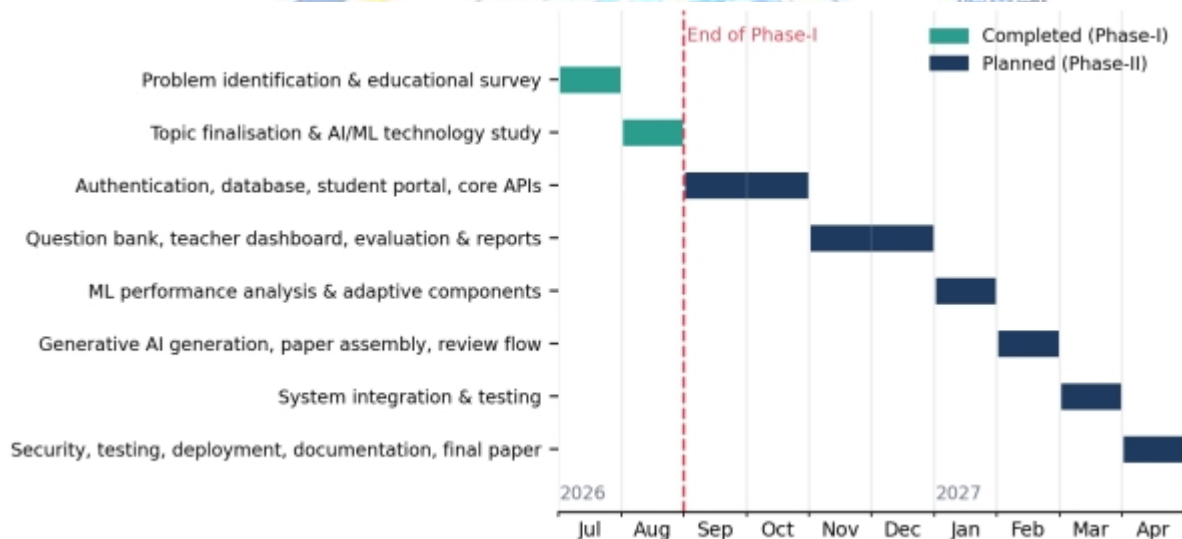


Fig. 5. Project timeline, July 2026 to April 2027. Phase-I activities are complete; Phase-II activities are planned.

VI. PRELIMINARY EXPERIMENTS ON SYNTHETIC DATA

Real institutional assessment data are not yet available, because the system has not been deployed. To check that the analytics design is workable and to give an early indication of the potential benefit of adaptive testing, we ran two experiments on **synthetic data**. These experiments verify the pipeline and illustrate expected behaviour; they are not evidence of performance on real students, and the numbers below will be replaced by results on real assessment data in Phase-II.

A. Learning-level classification

Data. We generated 1500 synthetic students, each with a latent ability drawn from a standard normal distribution. Nine features (Table 3) were generated as noisy functions of this ability. The learning level was derived from the ability plus noise (standard deviation 0.45), which makes the labels deliberately imperfect, and split into Low, Medium and High in a 30/40/30 ratio (450, 600 and 450 students). Data were divided into 1200 training and 300 test students with stratification.

Table 3. Features of the synthetic student dataset.

Feature	Description
Attendance (%)	Percentage of classes attended
Previous exam score	Score in the preceding examination (0 to 100)
Assignment completion (%)	Share of assignments submitted
Easy / medium / hard accuracy (%)	Percentage of correct answers at each difficulty level
Average time per question (s)	Mean response time
Attempts per topic	Mean number of attempts before completion
Topic score standard deviation	Variability of scores across topics

Method. Five classifiers from scikit-learn [13] were trained: logistic regression, an RBF-kernel SVM, a decision tree (depth 5), a random forest (300 trees) and gradient boosting. Each was assessed by five-fold stratified cross-validation on the training set and then on the held-out test set.

Results. Table 4 and Fig. 6 compare the models. Test accuracy ranges from 73.3% to 78.0%. The SVM (RBF) achieved the highest test macro-F1 (0.786) and ties with logistic regression for the best cross-validated accuracy (0.734) with a lower variance across folds. Differences between the top models are small relative to the fold-to-fold spread, so we do not claim that one model is clearly superior. The confusion matrix of the SVM (RBF) (Fig. 7) shows that every error is between neighbouring levels: 0 Low students were predicted as High and 0 High students as Low, with the 66 misclassifications all involving the Medium class. This is consistent with the high quadratic kappa of 0.804. Per-class F1 scores are 0.807 (Low), 0.750 (Medium) and 0.800 (High); the middle class is hardest, as expected for an ordinal target.

Table 4. Learning-level classification results on the synthetic dataset (test set of 300 students). CV = five-fold cross-validation on the training set; κ_w = quadratic weighted kappa.

Model	CV accuracy (mean $\pm$ SD)	Test accuracy	Test macro-F1	κ_w
Logistic Regression	0.734 $\pm$ 0.021	0.753	0.760	0.790
SVM (RBF)	0.734 $\pm$ 0.008	0.780	0.786	0.804
Decision Tree	0.694 $\pm$ 0.016	0.767	0.773	0.802
Random Forest	0.722 $\pm$ 0.017	0.757	0.764	0.788
Gradient Boosting	0.713 $\pm$ 0.019	0.733	0.741	0.771

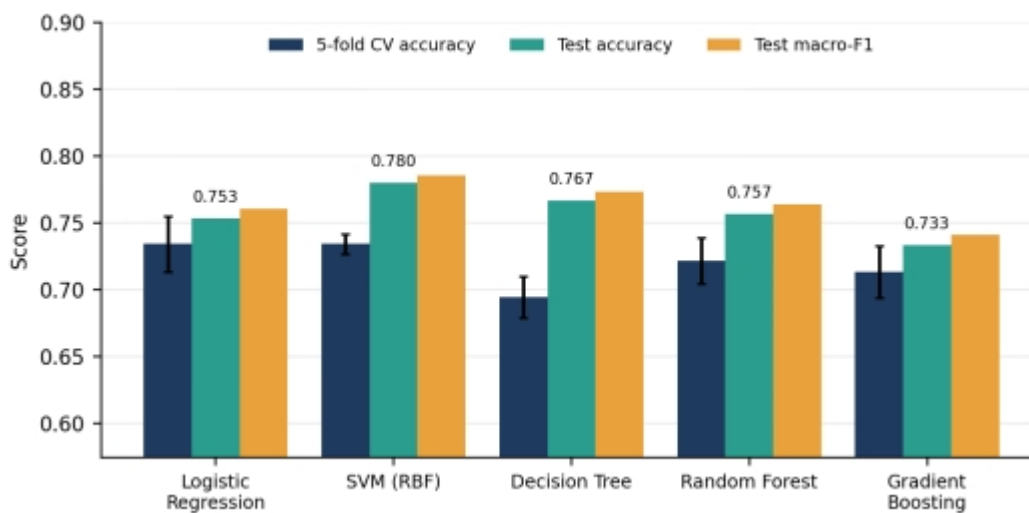


Fig. 6. Comparison of the five classifiers by cross-validated accuracy, test accuracy and test macro-F1 (error bars: one standard deviation across folds).

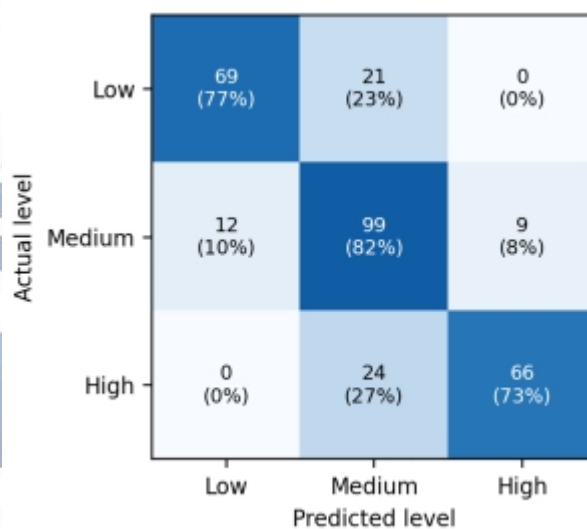


Fig. 7. Confusion matrix of the SVM (RBF) on the test set (counts and row percentages).

Permutation importance on the test set (Fig. 8) shows that the most influential features are medium-difficulty accuracy, easy-difficulty accuracy, hard-difficulty accuracy and previous exam score, while behavioural features such as time per question and assignment completion contribute little. This pattern arises partly because the synthetic generator was built with strong difficulty-wise signals, so it should not be read as an empirical finding. It does, however, illustrate the type of explanation the system will give teachers, and it underlines why difficulty-tagged questions (Section 4.2) matter: without difficulty labels, the most informative features could not be computed.

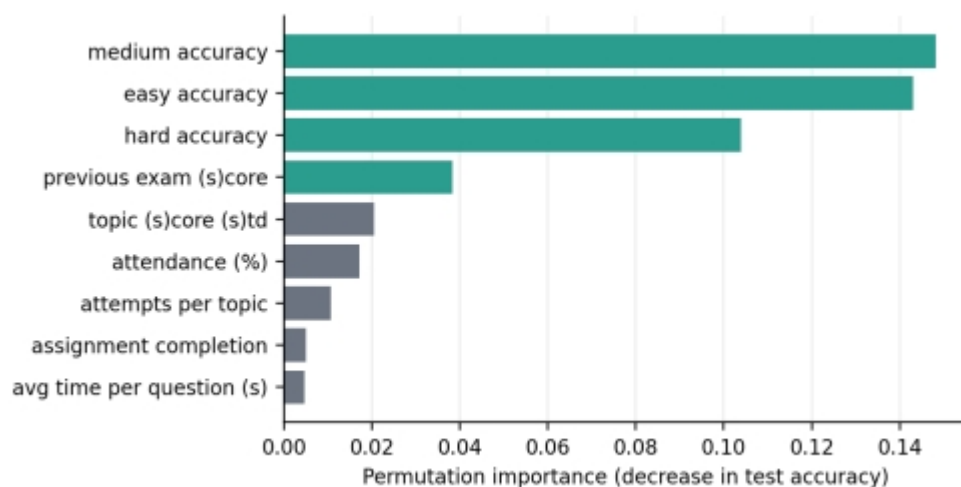


Fig. 8. Permutation feature importance of the SVM (RBF) (mean decrease in test accuracy over 20 repeats). The four highest-ranked features are highlighted.

B. Adaptive versus fixed assessment

Setup. We simulated a bank of 400 items with discrimination a drawn from a log-normal distribution (median 1.0) and difficulty b from a normal distribution with mean 0 and standard deviation 1.2, and 1000 simulated students with abilities from a standard normal distribution. Responses were generated from the 2PL model (Equation 6). Two strategies were compared: a fixed paper of randomly chosen items, and adaptive selection of the maximum-information item at the current EAP estimate (Equations 7 and 8). Ability-estimation error was measured as the root mean squared error (RMSE) between the estimated and true ability after each item, up to 30 items.

Results. Fig. 9 shows that adaptive selection lowers the error faster at every test length. After 10 items the RMSE is 0.404 for adaptive selection and 0.586 for the fixed paper, a reduction of about 31%. To reach an RMSE of 0.50, the adaptive strategy needs 6 items against 17 for the fixed paper; to reach 0.40 it needs 11 items against 29, that is 62% fewer. In practice this suggests that a personalised assessment could measure ability with a much shorter test, or with higher precision at equal length.

International Journal of Multidisciplinary
Allied Research Review and Practices



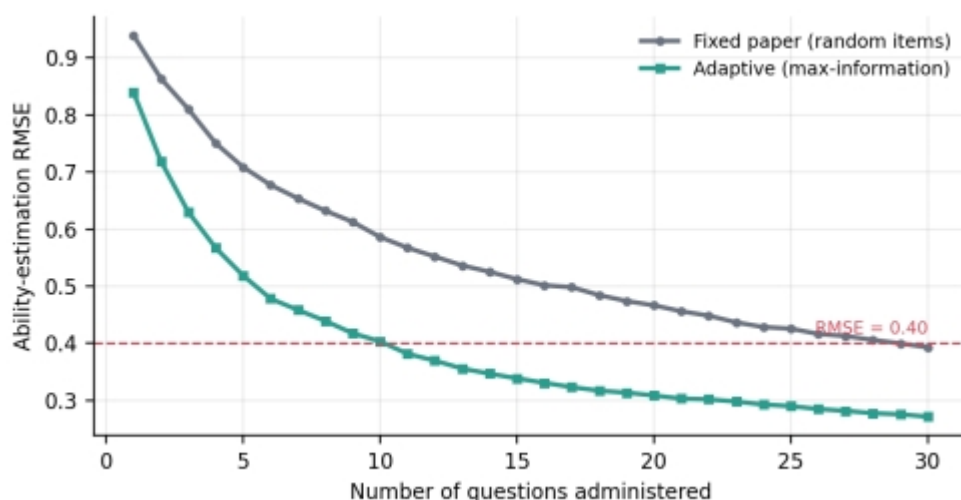


Fig. 9. Ability-estimation error against test length for fixed and adaptive assessment in the simulation (1000 simulated students, 400-item bank).

C. Discussion and threats to validity

Both experiments are simulations. In the classification experiment the features are informative by construction, so the reported accuracy only shows that the pipeline runs and that the metrics behave sensibly; real data will contain missing values, class imbalance, changing curricula and weaker signals. In the adaptive experiment the item parameters are known exactly, whereas in practice they must be estimated from response data, and real students exhibit guessing, fatigue and other effects that the 2PL model ignores. Exposure control and content balancing, which constrain adaptive selection in practice, were not simulated and would reduce the gain. Finally, all simulations used a single random seed. Phase-II will repeat both studies on real, anonymised assessment data with repeated runs and confidence intervals.

VII. EVALUATION PLAN

Each module will be evaluated with measures suited to its purpose (Table 5). Generated questions will be rated by teachers, and agreement between raters will be reported. Testing will use sample or controlled data before any deployment with real student records.

Table 5. Planned evaluation measures for each module.

Module	Measures	Method
Question generation	Relevance to source, validity, answerability, duplication rate, teacher acceptance rate	Teacher rating (1 to 5 scale) with inter-rater kappa; embedding similarity
Difficulty classification	Accuracy, macro-F1, agreement with teacher label and facility index	Held-out labelled questions
Paper generation	Blueprint conformance (marks, topic and difficulty deviation), duplicates, generation time	Automated checks on generated papers
Assessment scoring	Agreement with manual scoring, mean absolute error in marks	Comparison with teacher-scored scripts
Performance analysis (ML)	Accuracy, macro-F1, quadratic kappa, RMSE, feature importance	Stratified split and cross-validation

Adaptive assessment	Ability-estimation RMSE, test length, item exposure rate	Simulation and pilot testing
System quality	Response time, usability (SUS score), access-control and security tests	Load, user and security testing

VIII. LIMITATIONS AND ETHICAL CONSIDERATIONS

- **Data quality.** Generated questions and analytics depend on the completeness and correctness of syllabus content, the question bank and student responses. Ambiguous questions or missing responses can degrade results.
- **Fallible models.** ML classifications and GenAI outputs can be wrong, and learning patterns vary between students. Outputs are decision-support information, not guaranteed results.
- **Human oversight.** Generated questions must come from approved content and be reviewed by authorised teachers before use. Important actions such as question approval, assessment review and changes to student records remain under teacher supervision, and the system makes no final academic decision automatically.
- **Privacy and security.** Student academic data must be protected from unauthorised access. Role-based access, secure storage, content validation and audit logging are part of the design, in line with the principles of accountability and transparency emphasised in [6].
- **Content provenance.** External educational content must be verified for relevance, correctness and authorisation before it is used for generation.
- **Controlled testing.** The system will first be tested on sample or controlled data, and papers and reports will be validated before use in formal examinations or academic decisions.
- **Fairness.** Model performance will be checked across student groups and question types to detect systematic disadvantage before any use that affects students.

IX. CONCLUSION AND FUTURE WORK

This paper proposed an integrated framework for intelligent educational assessment that combines controlled Generative AI question generation, a structured question bank, blueprint-based paper generation, automated assessment, and ML and IRT-based analytics under teacher supervision. Phase-I established the problem through a school and college survey, reviewed the literature, identified the research gap and defined the architecture and evaluation plan. Synthetic-data experiments showed that the learning-level classification pipeline works end to end and indicated that adaptive item selection can reach a given measurement precision with substantially fewer questions.

Future work in Phase-II includes implementing the portals and services, building and validating the question bank, integrating the GenAI generation pipeline with retrieval and review, calibrating IRT parameters on real response data, and evaluating all modules on real assessment data with teachers and students as described in Section 7.

REFERENCES

- [1] G. Kurdi, J. Leo, B. Parsia, U. Sattler, and S. Al-Emari, "A systematic review of automatic question generation for educational purposes," *Int. J. Artif. Intell. Educ.*, vol. 30, pp. 121–204, 2020.
- [2] M. Last and G. Danon, "Automatic question generation," *WIREs Data Mining and Knowledge Discovery*, 2020.
- [3] O. Zawacki-Richter, V. I. Marín, M. Bond, and F. Gouverneur, "Systematic review of research on artificial intelligence applications in higher education – where are the educators?," *Int. J. Educ. Technol. High. Educ.*, vol. 16, art. 39, 2019.
- [4] E. Kasneci *et al.*, "ChatGPT for good? On opportunities and challenges of large language models for education," *Learn. Individ. Differ.*, vol. 103, art. 102274, 2023.
- [5] A. Tlili *et al.*, "What if the devil is my guardian angel: ChatGPT as a case study of using chatbots in education," *Smart Learn. Environ.*, vol. 10, art. 15, 2023.

- [6] W. Holmes *et al.*, "Ethics of AI in education: Towards a community-wide framework," *Int. J. Artif. Intell. Educ.*, vol. 32, pp. 504–526, 2022.
- [7] A. Vaswani *et al.*, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 5998–6008.
- [8] T. B. Brown *et al.*, "Language models are few-shot learners," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 1877–1901.
- [9] P. Lewis *et al.*, "Retrieval-augmented generation for knowledge-intensive NLP tasks," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 33, 2020, pp. 9459–9474.
- [10] L. Breiman, "Random forests," *Mach. Learn.*, vol. 45, no. 1, pp. 5–32, 2001.
- [11] R. K. Hambleton, H. Swaminathan, and H. J. Rogers, *Fundamentals of Item Response Theory*. Newbury Park, CA, USA: Sage, 1991.
- [12] B. S. Bloom, Ed., *Taxonomy of Educational Objectives, Handbook I: Cognitive Domain*. New York, NY, USA: David McKay, 1956.
- [13] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
- [14] W. J. van der Linden and C. A. W. Glas, Eds., *Computerized Adaptive Testing: Theory and Practice*. Dordrecht, The Netherlands: Kluwer, 2000.

