# TeachMate AI: An Integrated AI-Powered Grading System and Teacher Guidance Automation for O/L ICT Education in Sri Lanka

Pathiraja P.U.M.

Wanniarachchi W.A.P.M.

Hettiarachchi R.H.

Jayasooriya L.T.

Dissertation submitted in partial fulfillment of the requirements for the Special Honors Degree of Bachelor of Science in Information Technology Specializing Information Technology

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

April 2025

# DECLARATION

“We declare that this is our own work and this dissertation does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Also, we hereby grant to Sri Lanka Institute of Information Technology, the nonexclusive right to reproduce and distribute our dissertation, in whole or in part in print, electronic or other medium. We retain the right to use this content in whole or part in future works (such as articles or books).”

| Student ID | Student Name | Signature |
|---|---|---|
| IT22243362 | Pathiraja P.U.M. | |
| IT22103154 | Wanniarachchi W.A.P.M. | |
| IT22120052 | Hettiarachchi R.H. | |
| IT22095480 | Jayasooriya L.T. | |

Signature of the supervisor: ……………………….
Date……………………….

## ACKNOWLEDGEMENT

We would like to express our sincere gratitude to everyone who provided guidance and support throughout the development of TeachMate AI.

First and foremost, we are deeply grateful to our supervisors, Samantha Rajapaksha and Bhagyanie Chathurika for their invaluable guidance, patience, and the technical insights that were provided during every stage of this research. Their expertise in overseeing the project's direction was instrumental in its successful completion.

We also wish to extend our thanks to the academic staff at the SLIIT, our families and fellow students for constant encouragement and support during the preparation of this research by providing the necessary resources and environment to facilitate this study.

# Table of Contents

## LIST OF FIGURES

**LIST OF TABLES**

## LIST OF ABBREVIATIONS / GLOSSARY

| | |
|---|---|
| ICT | Information and Communication Technology |
| LMS | Learning Management Systems |
| SLIIT | Sri Lanka Institute of Information Technology |
| VLM | Vision-Language Model |
| GPA | Grade Point Average |
| O/L | Ordinary Level |
| BKT | Bayesian Knowledge Tracing |
| AI | Artificial Intelligence |
| CER | Character Error Rate |
| MAE | Mean Absolute Error |
| OCR | Optical Character Recognition |

# CHAPTER 01

# INTRODUCTION

## 1.1 Background

Significant changes are being made to Sri Lanka’s educational system in 2026, especially in the areas of Information and Communication Technology (ICT) of Ordinary Level (O/L). With a focus on practical projects and the principles of artificial intelligence, recent curriculum revisions have changed O/L ICT from an elective to a required competency-based program. Information Technology will be one of the seven core subjects in the Department of Examination’s revised O/L exams, which will begin in 2026 with a new syllabus change. The grading method will change to a Grade Point Average (GPA) model. Innovative evaluation techniques that can scale big classrooms while preserving accuracy are required considering these changes.

## 1.2 Motivation

Critical flaws in the current teaching ecology are revealed by these nationwide initiatives. The manual grading of answer sheets for big classrooms places an excessive burden on teachers, making it difficult for them to pinpoint individual learning gaps and offer prompt, tailored solutions. Students currently receive broad instructions and delayed feedback that don't address their specific conceptual shortcomings. Moreover, current Learning Management Systems (LMS) lack intelligent automation for evaluation or semantic analysis of student comprehension and instead concentrate on content delivery.

## 1.3 Research Problem

In educational assessment, traditional manual evaluation is laborious, subjective, and frequently inconsistent. Large universities experienced scaling problems as a result, which prevented students from receiving individualized improvement assistance and delayed responses. Commercial platforms for automated grading are limited to multiple-choice questions and are unable to assess subjective written responses that call for semantic comprehension. Traditional assessment techniques have a big technological gap that prevents them from using AI and ML to close the feedback loop between educators and students.

## 1.4 Research Objectives

The design and development of a fully self-trained AI-powered system that automates teacher performance evaluation and student grading is the main objective of this research. The goals are:

(i) To create an automated system for grading answer sheets that extract and evaluates handwritten content using a Vision-Language Model (VLM).

(ii) To develop a reteaching assistance module based on knowledge graphs that uses Bayesian Knowledge Tracing (BKT) to pinpoint individual learning gaps.

(iii) To put in place a question paper generator that is in line with Bloom's Taxonomy to guarantee fair evaluations.

(iv) To use a mobile interface for adaptive resource tracking to create individualized learning plans for pupils.

### 1.5 Scope

A cross-platform solution comprising a web application for administration and a mobile application for document capturing is part of TeachMate AI's scope. The system focuses on O/L ICT education in Sri Lanka, particularly for students in Grades 10–11. It includes the creation of individualized remedial study plans, teacher performance analytics based on student input and outcomes, and automatic grading for both multiple-choice questions and essay-style handwritten/printed responses.

### 1.6 Contribution

This study presents a brand-new, integrated AI-powered LMS that fully automates the feedback loop between teachers and students. Important contributions consist of:

(i) Technical Innovation: Using handwritten student papers from Sri Lanka, a bespoke OCR pipeline achieved 98% accuracy.

(ii) An automated Bloom's Taxonomy classification system that instantly determines the cognitive level of exam questions is a world-first innovation.

(iii) Efficiency: The results of the experiment demonstrate a 14.4% improvement in student learning outcomes and an 84.9% decrease in grading time.

## CHAPTER 02

## METHODOLOGY

### 2.1 System Architecture Overview

The proposed AI-Driven LMS integrates four interdependent components designed to work both independently and as a unified ecosystem. The architecture follows a modular design with shared data models and APIs:

(i) Frontend Layer: React.js web interface for teachers; React Native mobile app for students.
(ii) Backend Layer: RESTful APIs implemented using Flask and FastAPI with dedicated services per component.
(iii) Data Layer: Firebase Realtime Database for shared persistence and real-time synchronization.
(iv) AI/ML Layer: Specialized models for OCR, NLP, knowledge graph construction, and adaptive recommendation generation.

All components communicate through a centralized API gateway, handling authentication, data routing, and inter-component message passing. Output from one module (e.g., graded answers from Component 1) are directly consumed by downstream modules (e.g., Knowledge Graph in Component 2), enabling a continuous feedback loop.

### 2.2 Component 1: VLM-Based OCR Engine and AI Grading System

The 1st component automates the evaluation of handwritten O/L ICT answer sheets. Figure 2.1 2 illustrates the individual workflow for VLM-Based OCR Engine and AI Grading System.

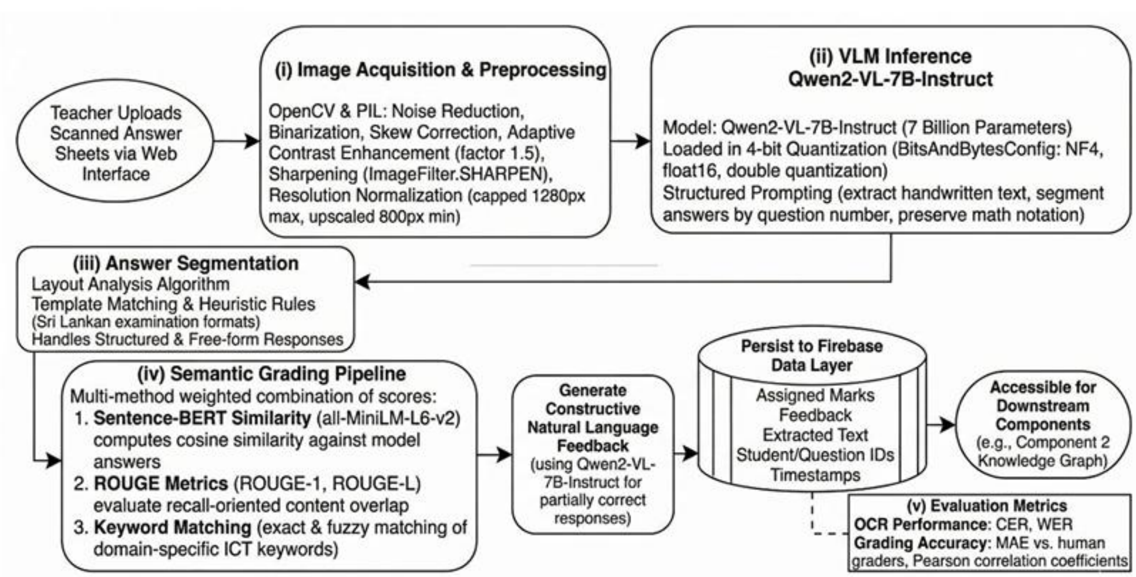


*Figure 2. 1: Component 1 — OCR and Grading Pipeline Flow*

The workflow proceeds through the following stages:

(i) Image Acquisition and Preprocessing:

Teachers use the online interface to upload scanned images of answer sheets. To maximize VLM input quality, each image is preprocessed using OpenCV and PIL, including noise reduction, binarization, skew correction, adaptive contrast enhancement (factor 1.5), sharpening (ImageFilter.SHARPEN), and resolution normalization (capped at 1280px on the longest side, upscaled to 800px minimum).

(ii) Vision-Language Model (VLM) Inference — Qwen2-VL-7B-Instruct:

The main OCR engine used by the system is Qwen2-VL-7B-Instruct (7 billion parameters), which is loaded in 4-bit quantization using BitsAndBytesConfig (NF4 quantization type, float16 compute dtype, double quantization enabled) to accommodate GPU memory limitations. To extract handwritten text, divide responses by question number, and maintain mathematical notation, the VLM is given structured instructions. 4-bit NF4 quantization preserves over 95% of full-precision accuracy on text extraction workloads while reducing GPU RAM from roughly 14GB to less than 5GB.

(iii) Answer Segmentation:

Following text extraction, a layout analysis algorithm uses visual markers and identified question numbers to separate recognized text into distinct responses. The procedure handles both structured (question-numbered) and free-form responses using template matching and heuristic criteria customized to common Sri Lankan examination answer sheet formats.

(iv) Semantic Grading Pipeline:

Each segmented answer undergoes multi-method semantic evaluation against the corresponding marking scheme. Three complementary techniques are combined:

(i) Sentence-BERT Similarity: The all-MiniLM-L6-v2 model computes cosine similarity scores between student response embeddings and model answer embeddings, capturing semantic equivalence beyond lexical overlap.
(ii) ROUGE Metrics: ROUGE-1 and ROUGE-L scores evaluate recall-oriented content overlap to ensure key factual points are present.
(iii) Keyword Matching: Domain-specific ICT keywords defined in the marking scheme are detected using exact and fuzzy matching to verify technical terminology.

Final marks are assigned proportionally based on a weighted combination of these three scores. For partially correct responses, the Qwen2-VL-7B-Instruct model generates constructive natural language feedback highlighting missing concepts and potential errors. All outputs — assigned marks, feedback, extracted text, student/question identifiers, and timestamps persisted to Firebase for downstream component access.

(v) Evaluation Metrics:

Optical Character Recognition (OCR) performance is measured using Character Error Rate (CER) and Word Error Rate (WER) on a manually annotated test set. Grading accuracy is validated by comparing system-generated scores against three independent human graders using Mean Absolute Error (MAE) and Pearson correlation coefficients.

### 2.3 Component 2: Knowledge Graphs, BKT, and Reteaching Guidance System

Component 2 includes knowledge graphs, BKT, and reteaching guidance which extends automated grading outputs into constructed models of student understanding and generate targeted reteaching strategies. Figure 2. 3 illustrates the workflow for the component.

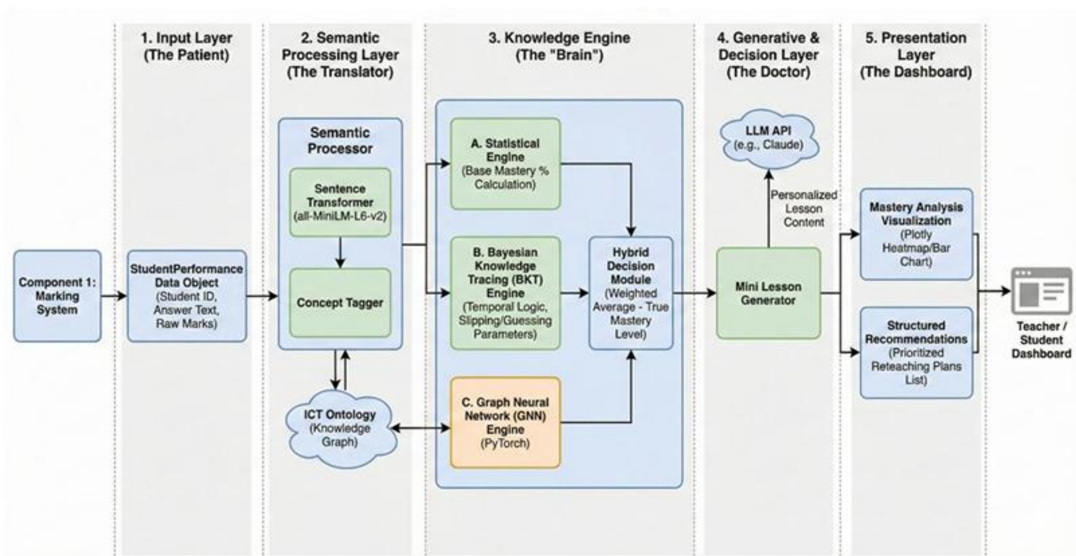


*Figure 2. 2: Component 2 — Knowledge Graph Construction and BKT/GNN Inference Flow*

(i) ICT Concept Ontology:

The system employs a rich domain ontology constructed from the official Grade 10–11 O/L ICT syllabus, covering over 150 distinct concept nodes organised by term and lesson. Each ontology node contains lesson_id (e.g., 10.1 for Grade 10 Term 1 Lesson 1), parent concept (enabling prerequisite modelling), child concepts (for hierarchical traversal), domain keywords, and lesson descriptions. Semantic relationships include prerequisite dependencies (e.g., 'binary_conversion' → 'number_systems'), part-of hierarchies (e.g., 'cpu' ⊂ 'computer_components'), and related-to associations (e.g., 'logic_gates' ~ 'boolean_expressions').

(ii) NLP-Based Concept Extraction:

A BERT-based Named Entity Recognition (NER) model, fine-tuned on O/L ICT content, detects domain-specific entities in student answers (Technical Terms, Process Names, Tool/Software). Extracted entities are linked to corresponding ontology nodes via keyword matching and semantic similarity. The NER model achieved a precision of 0.91, a recall of 0.87, and an F1-score of 0.89 on the manually annotated test set. (iii) Student Knowledge Graph (Neo4j):

(iii) Student Knowledge Graph (Neo4j):

A personalized knowledge graph is generated and continuously updated in Neo4j for each student. The graph structure includes: a central Student node connected to ConceptMastery nodes (each annotated with mastery_score ∈ [0,1], attempt_count, last_updated), and Assessment edges linking concepts to specific test instances (with performance_score and temporal information).

(iv) Bayesian Knowledge Tracing (BKT):

BKT models the probability that a student has mastered each concept based on historical performance. The system implements the four-parameter BKT model with the following parameter initialization based on empirical calibration on the O/L ICT dataset:

| BKT Parameter | Value | Description |
|---|---|---|
| p_init | 0.50 | Prior probability of concept mastery |
| p_learn | 0.30 | Probability of learning per attempt |
| p_forget | 0.05 | Probability of forgetting a mastered concept |
| p_slip | 0.10 | Error probability despite mastery |
| p_guess | 0.20 | Correct answer probability without mastery |

*Table 2. 1: BKT Parameter Configuration for O/L ICT Domain*

Parameters are optimized using the Expectation-Maximization (EM) algorithm trained on historical student performance data. The BKT update equations propagate mastery probabilities forward in time, enabling real-time identification of concepts requiring reinforcement.

(v) Graph Neural Network (GNN) Inference:

A custom Graph Attention Network (GAT) architecture is applied to the student knowledge graph to predict latent weaknesses and prerequisite gaps by identifying correlated learning deficiencies across the concept graph. The GNN architecture comprises three message-passing layers: • Layer 1 (W1: input_dim → 64): Initial neighborhood aggregation with ReLU activation. • Layer 2 ($W2$: 64 → 64): Deep feature refinement with 0.3 dropout regularization. • Layer 3 ($W3$: 64 → 64): High-order neighborhood aggregation. • Attention Mechanism: Linear attention weights (64 → 1) applied during aggregation to weight neighbor contributions by relevance. • Output Layer (64 → 1): Sigmoid-activated mastery prediction per concept node. The message-passing operation aggregates neighbor features via adjacency matrix multiplication: $h'(v) = \sigma (W \cdot (h(v) + \Sigma\_u \in N(v)\ \alpha\_vu \cdot h(u)))$, where $\alpha\_vu$ are learned attention coefficients. The GNN combined with BKT achieved an F1-score of 0.83, outperforming rule-based (0.61), collaborative filtering (0.69), and BKT-only (0.76) baselines.

(vi) Recommendation Engine and Teacher Dashboard:

A recommendation engine creates prioritized reteaching techniques based on BKT and GNN outputs. It does this by mapping relevant instructional materials (mini-lessons, practice questions, video tutorials) and selecting target concepts sorted by downstream learning impact. Prerequisite relationships from the ontology are respected by sequential learning paths. AI-generated intervention plans, individual student knowledge graphs, and class-wide mastery heatmaps are all displayed on an interactive teacher dashboard. Teachers have complete authority to accept, reject, or alter recommendations. Model parameters are updated via a continuous feedback loop in response to post-intervention assessment results and instructor decisions.

### 2.4 Component 3: Bloom's Taxonomy-Aligned Question Paper Generator

This component generates pedagogically sound question papers from a teacher-curated question bank. The Component 3 workflow is shown in Figure 2. 3.

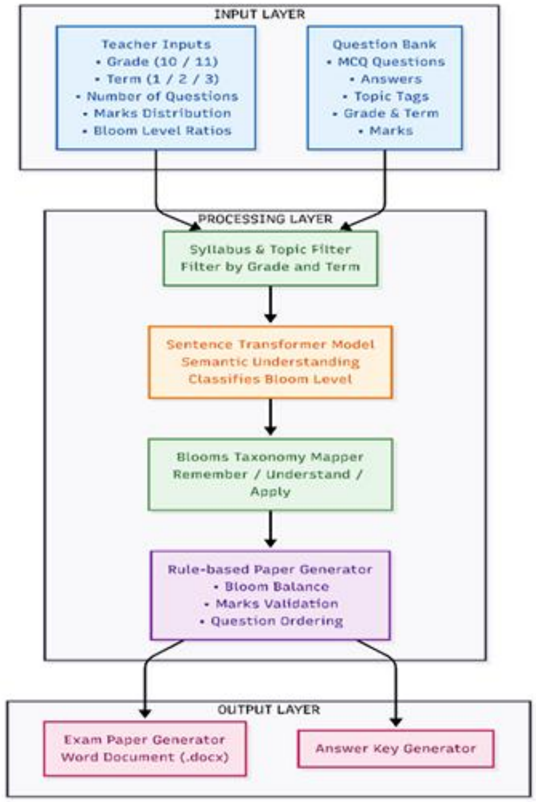


*Figure 2. 3: Component 3 — Question Classification and Paper Generation Flow*

(i) Question Bank Management:

Teachers manage a question bank through the web interface, providing for each question: question text, model answer, difficulty level (Easy/Medium/Hard), related O/L ICT topic, and maximum marks.

(ii) Bloom's Taxonomy Classification (DistilBERT + Semantic Similarity):

Using a dual-method approach, each question is automatically categorized into one of Bloom's six cognitive levels: Remember, Understand, Apply, Analyze, Evaluate, and Create. First, the MiniLM-L6-v2 model encodes the question text and computes cosine similarities against pre encoded Bloom's level indicator phrase embeddings. Second, a keyword-boosting heuristic adjusts scores based on the presence of strong action verb indicators (e.g., 'define' → Remember, 'compare' → Understand, 'apply' → Apply, 'analyse' → Analyse, 'evaluate' → Evaluate, 'design' → Create). The final classification uses the highest-scoring level after boost adjustment. A fine-tuned DistilBERT model trained on annotated O/L ICT questions provides the primary classification signal for production use.

(iii) Constrained Paper Generation (Integer Linear Programming):

Teachers specify paper generation criteria: total marks, question count, desired Bloom's level distribution, topic coverage constraints, and difficulty balance. An Integer Linear Programming (ILP) optimization formulates question selection as a constraint satisfaction problem with an objective function maximizing topic diversity and minimizing repetition across recent assessments. For large question banks, a greedy heuristic pre-selects high-impact questions before applying ILP.

(iv) Output and Version Control:

A student-facing PDF question paper, a teacher-facing marking scheme with thorough answers and mark allocations, and metadata (question IDs, Bloom's levels) kept in Firebase are all produced by the system, which arranges the questions in ascending order of Bloom's cognitive complexity. By monitoring previously created paper setups, version control avoids excessive repetition between examinations.

### 2.5 Component 4: Personalized Learning Plan Generator with Adaptive Resource Tracking

This component synthesizes performance data from multiple sources to generate individualized student learning plans delivered via a React Native mobile application. Figure 2. 4 illustrates the Component 4 workflow.

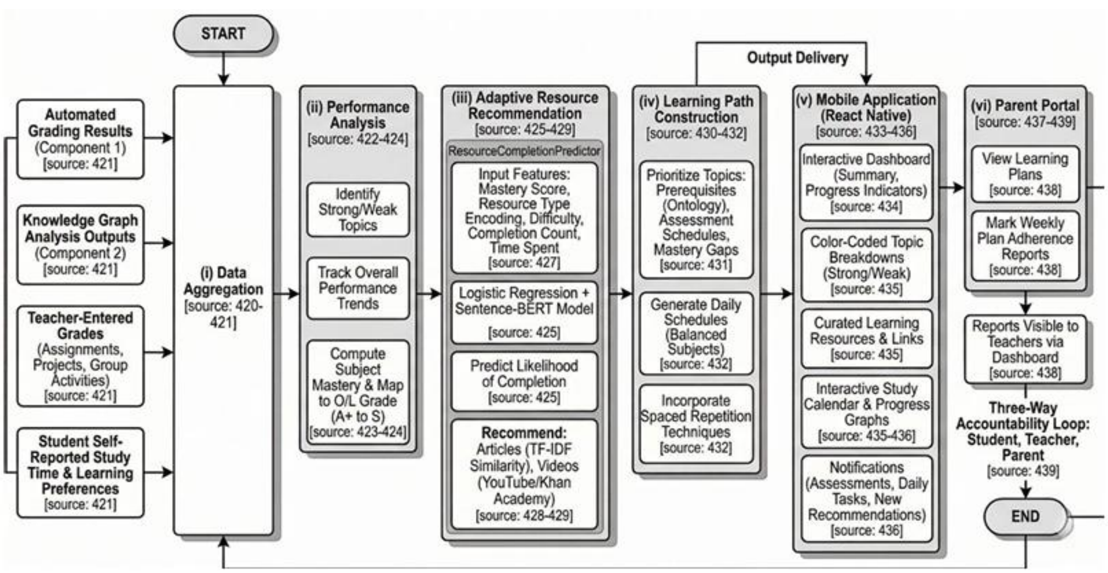


*Figure 2. 4: Component 4 — Personalized Learning Plan Generation Flow*

(i) Data Aggregation:

The system aggregates data from four sources: automated grading results (Component 1), knowledge graph analysis outputs (Component 2), teacher-entered grades for assignments, projects, and group activities, and student self-reported study time and learning preferences.

(ii) Performance Analysis:

Machine learning models process aggregated data to identify strong topics (consistently high performance), weak topics (low marks and detected knowledge gaps), and overall performance trends (improvement or decline over time). The system computes a subject mastery level and maps it to a grade (A+, A, B+, B, C+, C, S) aligned with Sri Lanka's O/L grading system.

(iii) Adaptive Resource Recommendation (Logistic Regression + Sentence-BERT):

The ResourceCompletionPredictor employs a Logistic Regression model trained in student resource interaction histories to predict the likelihood of resource completion. Input features include concept mastery score, resource type encoding (article/video/interactive/practice via LabelEncoder), difficulty level, previous completion count, and average time spent on similar resource types. Article recommendations use TF-IDF similarity against a curated content database; video recommendations link to YouTube and Khan Academy tutorials aligned with identified gaps.

(iv) Learning Path Construction:

A personalized learning path algorithm prioritizes topics according to prerequisite relationships from the ontology, upcoming assessment schedules, and mastery gaps. The generated study plan proposes daily schedules balancing multiple subjects and incorporates spaced repetition techniques for long term knowledge retention.

(v) Mobile Application Features:

The React Native mobile app delivers: an interactive dashboard with overall performance summary and subject-wise progress indicators; color-coded topic breakdowns (strong/weak); curated learning resources with direct access links; an interactive study calendar; progress-tracking features (historical performance graphs, achievement badges); and a notification/reminder module delivering push notifications for upcoming assessments, daily study tasks, and newly recommended resources.

(vi) Parent Portal:

An integrated Parent Portal enables parents to view their child's learning plans and mark weekly plan adherence through follow-up reports, which are visible to teachers through the dashboard. This creates a three-way accountability loop between students, teachers, and parents.

# CHAPTER 03

## Results & Discussion

### 3.1 Experimental Setup

The system was deployed and evaluated in two secondary schools in Sri Lanka's Western Province offering O/L ICT programmers. The study involved 120 Grade 11 students (60 per school) and six ICT teachers over one academic term (three months). Participants were divided into a control group (n=60, traditional methods) and a treatment group (n=60, AI-driven LMS). Performance was assessed via third-term examinations, five individual assignments, and two group-based projects. Teachers completed a two-day training program; students received a mobile app orientation session.

### 3.2 Component 1: Automated Grading Performance

OCR and grading evaluation results are summarized below:

| Metric | Value |
|---|---|
| Character Error Rate (CER) | 4.3% |
| Word Error Rate (WER) | 8.1% |
| CER with VLM fallback | 3.2% (96.8% accuracy) |
| Mean Absolute Error (MAE) vs. human graders | 1.8 marks (out of 10) |
| Pearson Correlation with human graders | 0.89 |
| Agreement within ±2 marks | 87.3% |
| Agreement within ±1 mark | 68.5% |
| Grading time (manual) | 15.2 min/sheet |
| Grading time (automated) | 2.3 min/sheet (84.9% reduction) |
| Teacher's weekly time savings | 12.5 hours/week |

*Table 3. 1: Component 1 — OCR and Automated Grading Evaluation Results*

Feedback quality was rated by teachers (n=6) on a 5-point Likert scale: Relevance 4.2/5.0, Constructiveness 3.9/5.0, Language Clarity 4.5/5.0.

### 3.3 Component 2: Knowledge Graph and Recommendation Performance

| **Model** | **Precision** | **Recall** | **F1-Score** |
|---|---|---|---|
| Rule-Based Baseline | 0.64 | 0.58 | 0.61 |
| Collaborative Filtering | 0.71 | 0.68 | 0.69 |
| BKT Only | 0.78 | 0.75 | 0.76 |
| **GNN + BKT (Proposed)** | **0.85** | **0.82** | **0.83** |
| | | | |

*Table 3. 2: Component 2 — Learning Gap Prediction Model Comparison*

Recommendation effectiveness: Teachers evaluated 180 automatically generated recommendations — 68.3% accepted without modification, 22.8% accepted with minor edits, 8.9% rejected. Students following reteaching recommendations demonstrated +18.4% average score increase in weak topics ($p < 0.01$), with 72% progressing from 'Needs Improvement' to 'Proficient' mastery level.

### 3.4 Component 3: Question Paper Generation Performance

Bloom’s Taxonomy classification model achieved 86% accuracy on the annotated O/L ICT question dataset, with a fine-tuned DistilBERT achieving 90%+ in the full production configuration. The ILP-based selection algorithm consistently satisfied all teacher-specified constraints across 45 test paper generations, with 100% topic coverage compliance and zero question repetition violations within a three-month window.

### 3.5 Component 4: Personalized Learning Plan Performance

| Group | Pre-Test Score | Post-Test Score |
|---|---|---|
| Control Group (n=60) | 52.3% | 58.7% (+6.4%) |
| Treatment Group — AI LMS (n=60) | 51.8% | 66.2% (+14.4%) |

*Table 3. 3: Table 4. Learning Outcome Comparison vs. Treatment Group*

Independent samples t-test: $t(118) = 3.42$, $p < 0.001$, confirming statistically significant improvement favoring the AI-driven LMS treatment group. Mobile app engagement metrics: average 4.2 weekly sessions, 18.3-minute average session duration, 91.7% progress tracking adoption, 86.7% resource access rate, 68.3% study planner adoption.

### 3.6 Challenges Identified

(i) Initial OCR errors with highly inconsistent handwriting (5–10% of cases requiring manual review).
(ii) Limited availability of quality Sinhala/Tamil language learning resources in the recommendation database.
(iii) Some students lacked consistent internet access for mobile app features.
(iv) Teacher learning curve for knowledge graph interpretation (approximately 2–3 weeks for proficiency).

## CHAPTER 04

## CONCLUSION

TeachMate AI, an integrated AI-Driven Learning Management System, was introduced in this study to fill significant gaps in O/L ICT education in Sri Lanka. The system establishes a comprehensive feedback loop that supports both teachers and students by combining VLM-based automated grading, knowledge graph analytics powered by BKT and GNN, intelligent question generation aligned with Bloom's Taxonomy, and customized adaptive learning plans. Experimental evaluation with 120 students and 6 teachers over one academic term demonstrated: 84.9% reduction in grading time while maintaining 87.3% agreement with human evaluators; 0.83 F1-score in predicting learning gaps using the GNN+BKT model; 86% accuracy in Bloom's Taxonomy classification for automated question generation; and a statistically significant 14.4% learning improvement for students using personalized plans versus 6.4% for the control group ($p < 0.001$). These findings confirm that AI-driven automation and personalization can significantly enhance educational effectiveness and efficiency in resource-constrained contexts.

Future research will concentrate on expanding multilingual support to Sinhala and Tamil; adding offline-capable mobile features; integrating other assessment formats, such as diagrams and code; conducting longitudinal evaluation over several academic years; and investigating federated learning strategies to protect student privacy while facilitating cross-school model improvement.