IMPERIAL

AI-enabled Personalised Learning Platform

Yin Xian Liang (Sofia)

💡 Personalised learning, empowered by technology and community, that transforms education.

Supervisors: Dr. Konstantinos Gkoutzis, Dr. Ovidiu Serban

Objectives

Provide a **user-based** support platform for the sharing of learning resources.

Build a **unique learner model** based on learner's profile and learning data.

Engineer **appropriate prompts** to promise quality of **auto-generated content**.

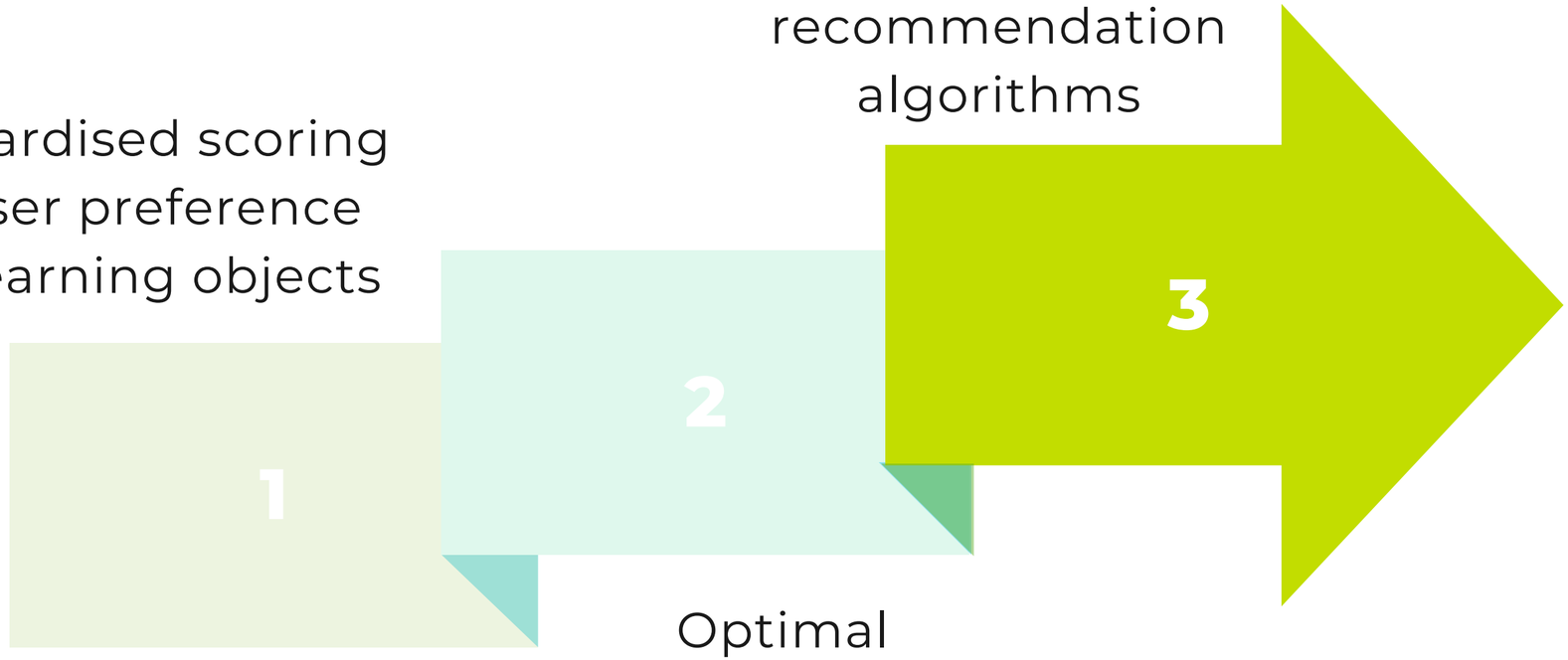
Customise selection of learning content to boost user satisfaction and motivation.

Contributions

Standardised scoring for user preference and learning objects

Optimal metaprompting strategy

Comparison study of recommendation algorithms



Learner Profile

Main aspect of customisation that offers adapted presentation of learning materials

Learner Profile

Learner Model

FSLSM Learning Styles

Index of Learning Styles Questionnaire

Learning Object Model

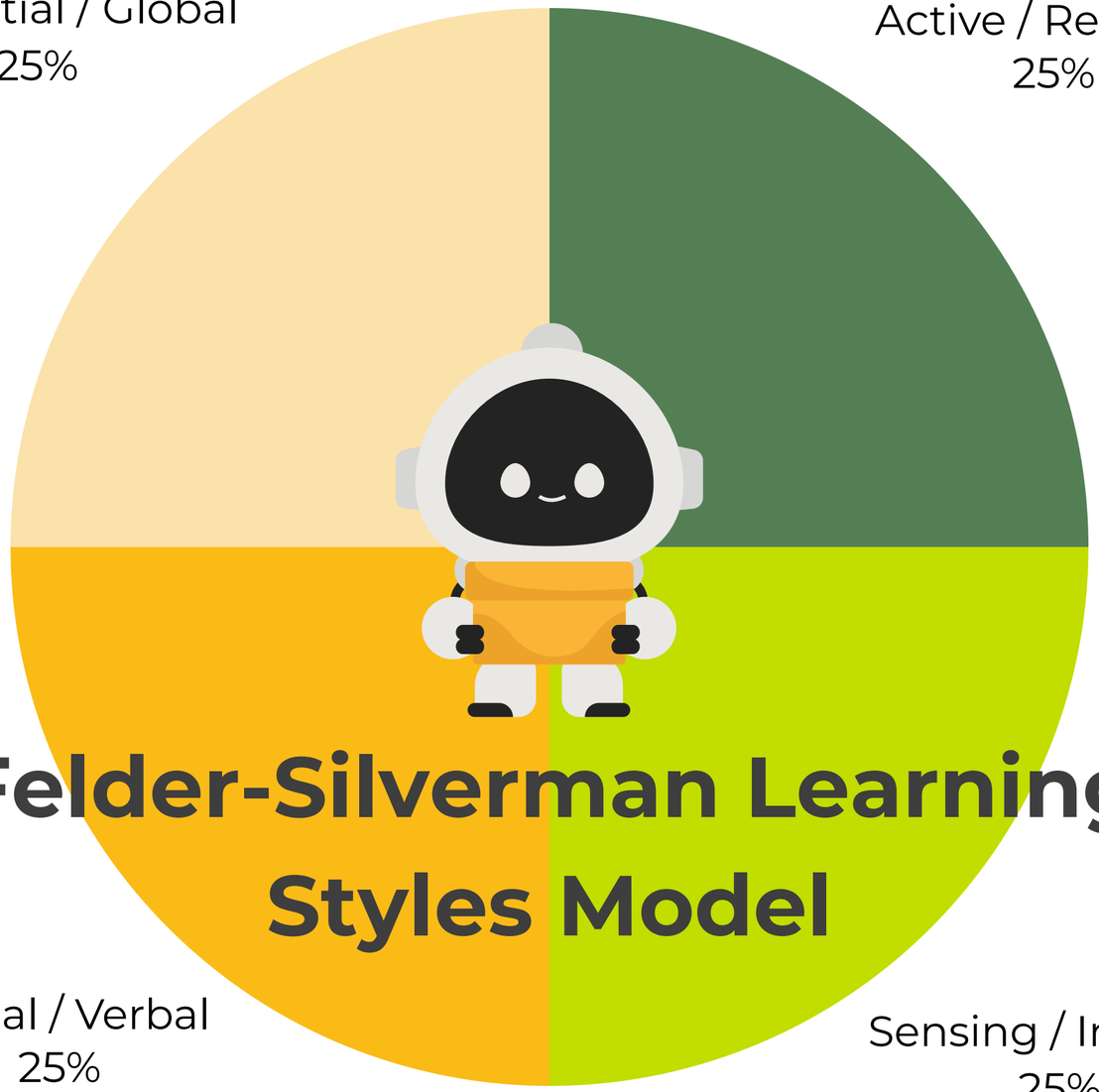
IEEE Learning Object Metadata

Scoring Algorithm

‘A Felder and Silverman Learning Styles Model Based Personalization Approach to Recommend Learning Objects’

Sequential / Global
25%

Active / Reflexive
25%



Felder-Silverman Learning Styles Model

Visual / Verbal
25%

Sensing / Intuitive
25%

Index of Learning Styles Questionnaire

- 44 Questions
- 4 dimensions, -11 to +11
- normalised from 0 to 1
- vector of 8 real numbers

```
learningPreferences: {
  active: 0.63636363
 reflexive: 0.36363636
  sensing: 0.36363636
 intuitive: 0.63636363
  visual: 0.90909090
  verbal: 0.09090909
sequential: 0.09090909
 global: 0.90909090 }
```

Learning Object


```
_id: ObjectId('6669a7b54b1d16676220b010')
▸ general : Object
▸ lifecycle : Object
▸ technical : Object
▾ educational : Object
    interactivityType : "expositive"
    learningResourceType : "narrative text"
    interactivityLevel : "low"
    context : "higher education"
    difficulty : ""
▸ content : Object
▾ score : Object
    act : 0
    ref : 1
    vis : 0
    ver : 1
    sen : 0
    int : 1
    seq : 1
    glo : 0
  __v : 0
  createdAt : 2024-06-12T13:50:45.531+00:00
  updatedAt : 2024-06-12T13:53:43.079+00:00
```


Figure 3.2: Example of a learning object stored in the project database


```
▾ content : Object
    text : "We will now attempt to make the guessing game more user friendly.
            …"
    link : ""
  ▸ audio : Object
  ▾ questionnaire : Object
    ▸ openAI : Array (empty)
    ▸ vertexAI : Array (empty)
  ▸ exercise : Object
  ▸ glossary : Object
  ▸ challenge : Object
  ▸ transcript : Object
  ▸ description : Object
    embed : false
    aiGenerated : false
    image : ""
    video : ""
```


Figure 3.3: The further categorisation of learning object content

IEEE Learning Object Metadata Standard

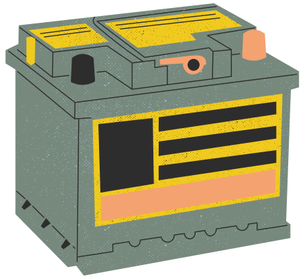
Nr.	IEEE LOM Standard	Value Space
4.1	**format**	MIME types based on IANA (e.g., video/mpeg, text/html)
5.1	**interactivity type**	active, expositive, mixed
5.2	**learning resource type**	exercise, simulation, questionnaire, diagram, figure, graph, index, slide, table, narrative text, exam, experiment, problem statement, self assessment
5.7	**difficulty**	very easy, easy, medium, difficult, very difficult

‘A Felder Silverman Learning Styles Model Based Personalization Approach to Recommend Learning Objects’

0 = Active, 1 = Reflexive

Learning Resource Type value	Active/Reflective dimension	Sensing/Intuitive dimension	Visual/Verbal dimension	Sequential/Global dimension
Exercise	0	0	0	0
Simulation	0	0	0	0
Questionnaire	0	1	0	0
Diagram	1	0	0	0
Figure	1	0	0	0
Graph	1	0	0	0
Index	1	0	0	1
Slide	1	1	0	0
Table	1	0	0	0
Narrative text	1	1	1	0
Exam	0	1	0	0
Experiment	0	0	0	0
Problem statement	0	1	0	0
Self assessment	0	1	0	0
Lecture	1	1	0	1

Modified lecture to be visual and global based on user ratings



Automated Content Generation

A curated bank of lesson content that offers a variety of materials, providing an optimal learning experience to diverse learners.

Comparison of Large Language Models (LLMs)

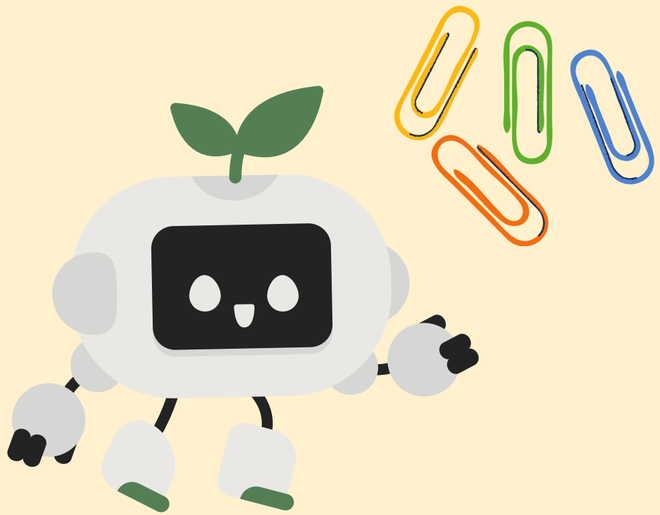
Model	Context window	Modality		Price (<=128K context window)
Gemini-1.5-Pro	Top quality up to 2M tokens	Input	Image	$0.001315 / image
			Video	$0.001315 / second
			Text	$0.00125 / 1k characters
			Audio	$0.000125 / second
		Output	Text	$0.00375 / 1k characters
Gemini-1.0-Pro	32K tokens	Input	Image	$0.0025 / image
			Video	$0.002 / second
			Text	$0.000125 / 1k characters
		Output	Text	$0.000375 / 1k characters
Text-to-Speech		Input	Text	
		Output	Audio	$0.000016 / byte
Speech-to-Text		Input	Audio	
		Output	Text	$0.024 / min

Table 4.1: VertexAI models API requests pricing table as of 14th June 2024 [53] [54] [55]

Model	Context window	Modality		Price (<=128K context window)
GPT-4o	128K tokens	Input	Image	$0.003825 / 1024 x 1024 px
			Text	$0.005 / 1K tokens
		Output	Text	$0.015 / 1K tokens
GPT-3.5	16,385 tokens	Input	Text	$0.0005 / 1K tokens
		Output	Text	$0.0015 / 1K tokens
TTS-1		Input	Text	
		Output	Audio	$0.015 / 1K tokens
Whisper-1		Input	Audio	
		Output	Text	$0.006 / min
Dall-E-3		Input	Text	
		Output	Image	$0.040 / 1024px x 1024px

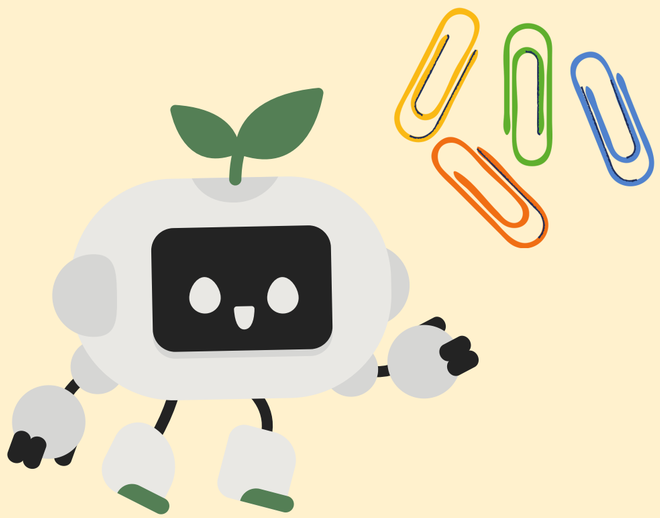
Table 4.2: OpenAI models API requests pricing table as of 14th June 2024 [56]

Component-wise Content Generation



Lesson-wise Content Generation

Component-wise Content Generation



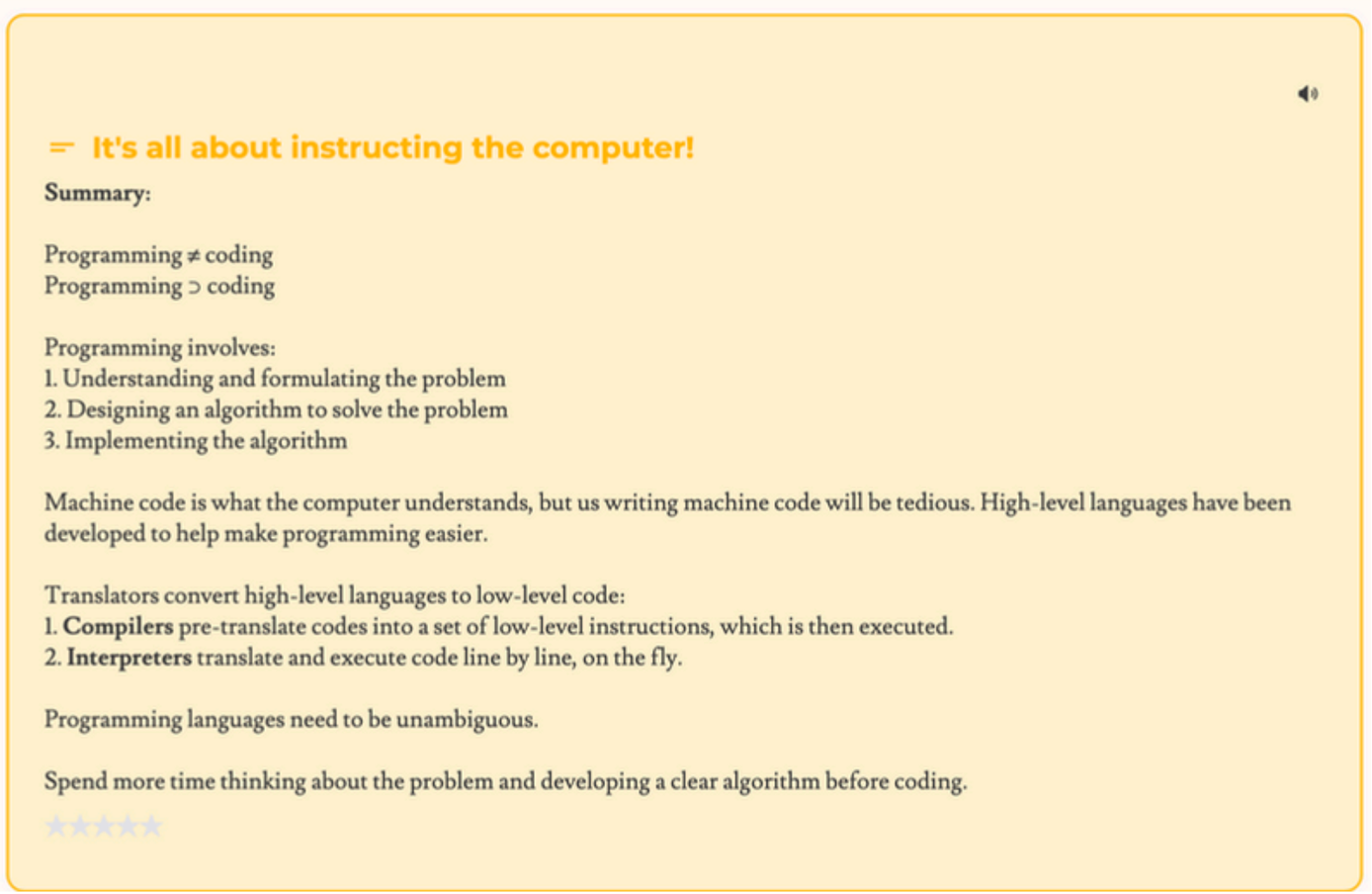
Transitioning same content from one medium to another

pleap

Text-to-Podcast

Increases digestibility, enabling visual learners to engage with a body of text

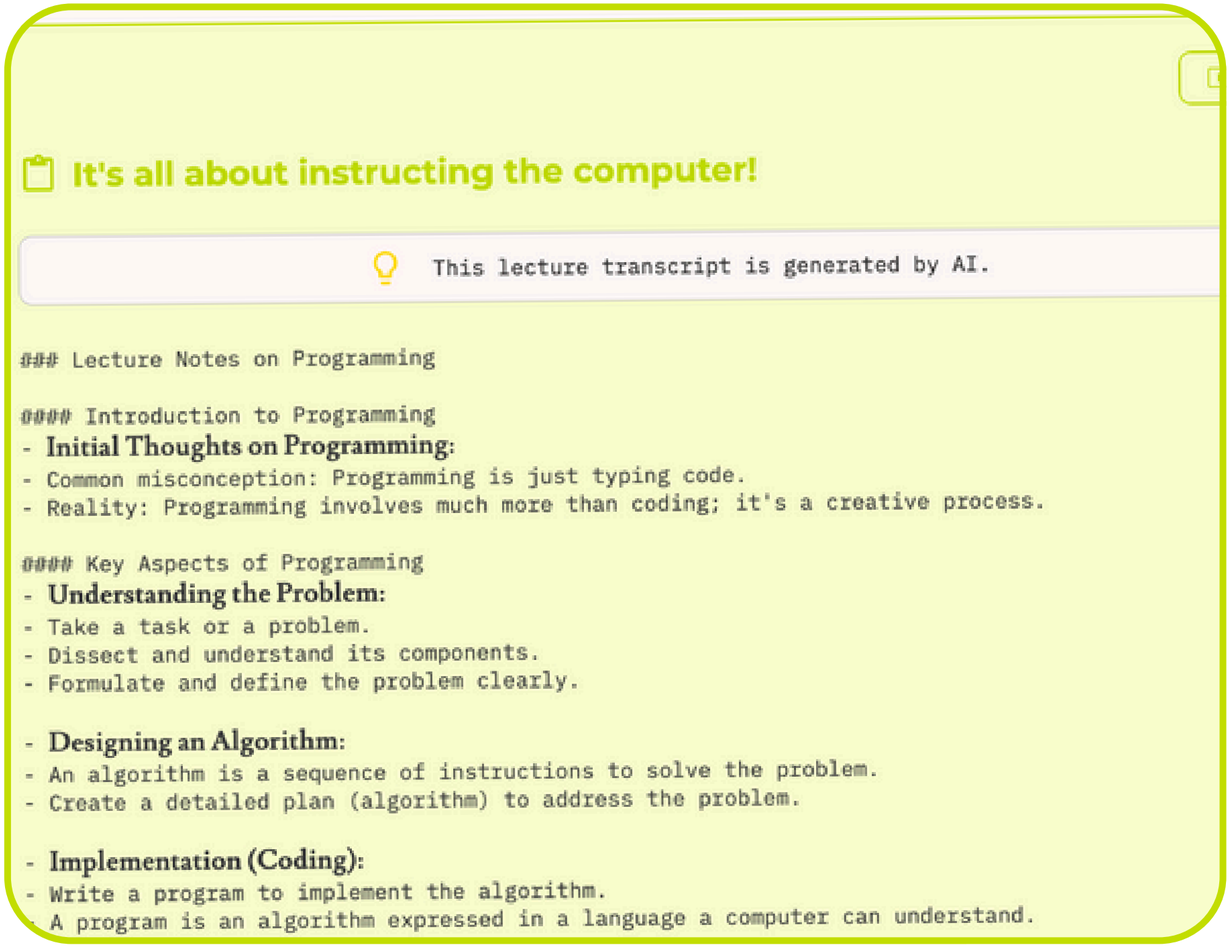
- ✓ **Google Cloud Text-to-Speech**
- ✓ **OpenAI's TTS**

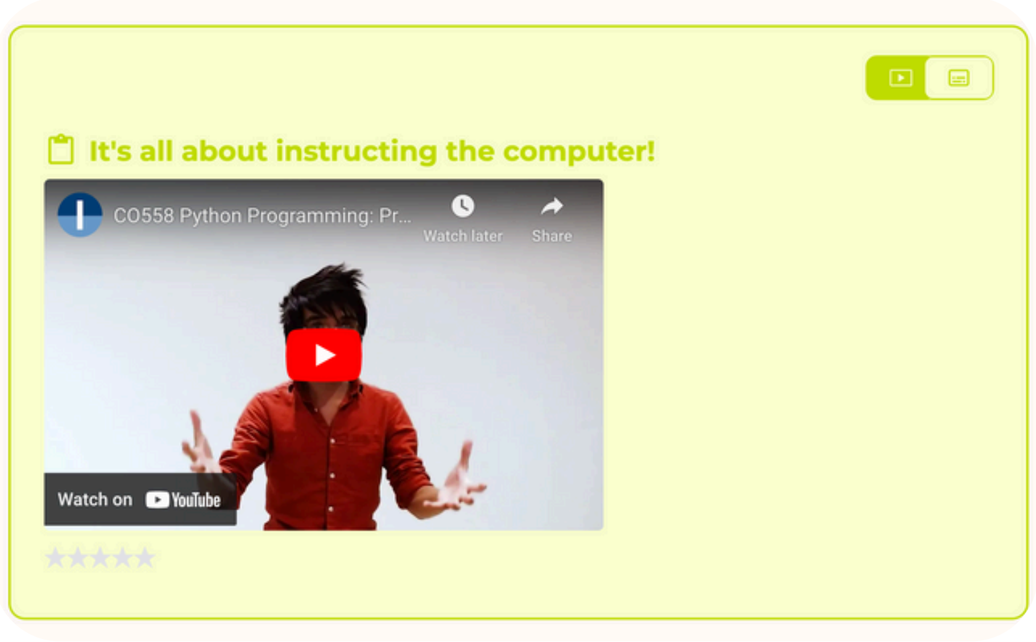


pleap

Video Transcript

Rewrite transcript in format of lecture notes





OpenAI Whisper-1

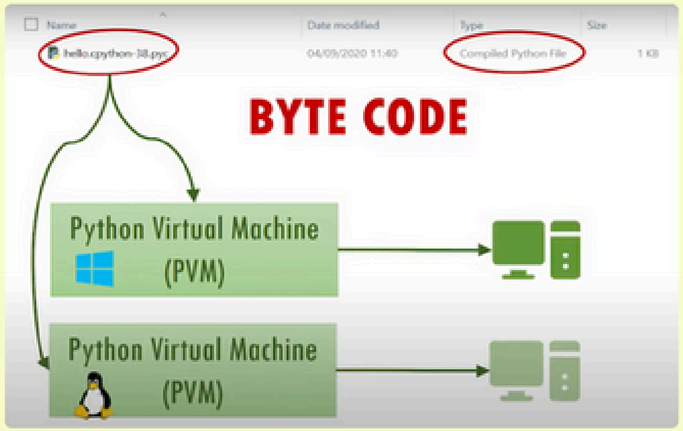
Google Cloud Speech-to-Text

pleap

Slide-to-Lecture Notes

Write alternative lecture notes based on the diagram

It's all about instructing the computer!



It's all about instructing the computer!

This slide description is generated by AI.

Python Execution Model

Python code is not directly executed by the operating system. Instead, it undergoes a two-step process:

1. **Compilation to Byte Code:**
- When a Python program (`.py` file) is executed, it is first compiled into byte code.
- Byte code is a low-level, platform-independent representation of the code.
- The compiled byte code is saved in a `.pyc` file (as shown in the image with `hello.cpython-38.pyc`).

2. **Execution by Python Virtual Machine (PVM):**
- The byte code is then executed by the Python Virtual Machine (PVM).
- The PVM is an interpreter that reads and executes the byte code instructions.
- The PVM is platform-specific, meaning there are different PVMs for Windows, Linux, macOS, etc.

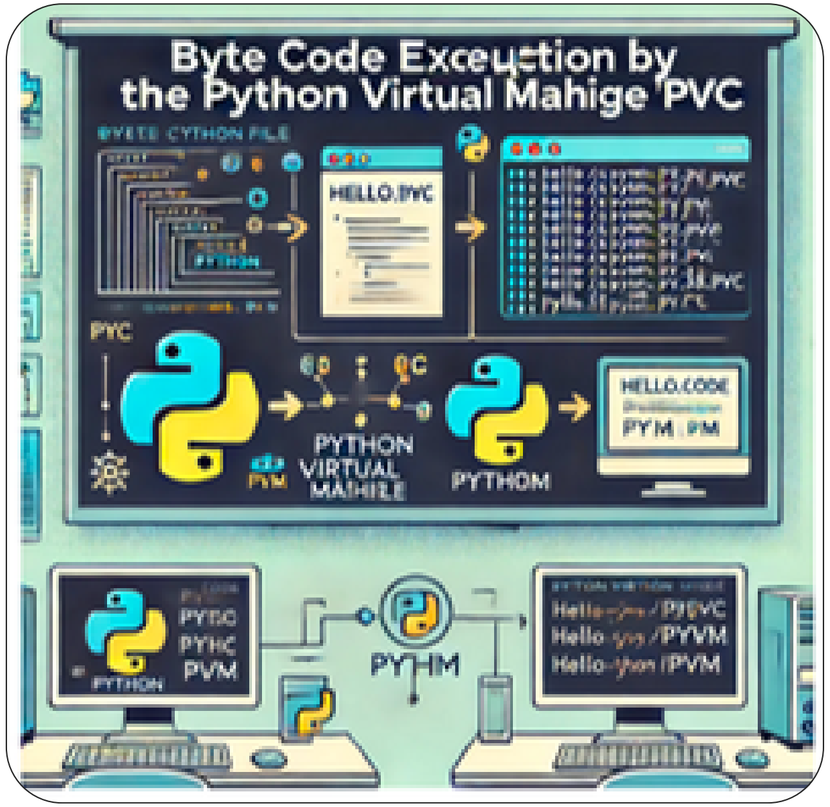
Gemini-1.5-Pro

GPT-4o

pleap

Lecture Notes-to-Slides

Struggled to present coherent text or effectively represent key message of text through relevant images



DALL-E

Generate more interactive elements for active learning

Lesson-wise Content Generation

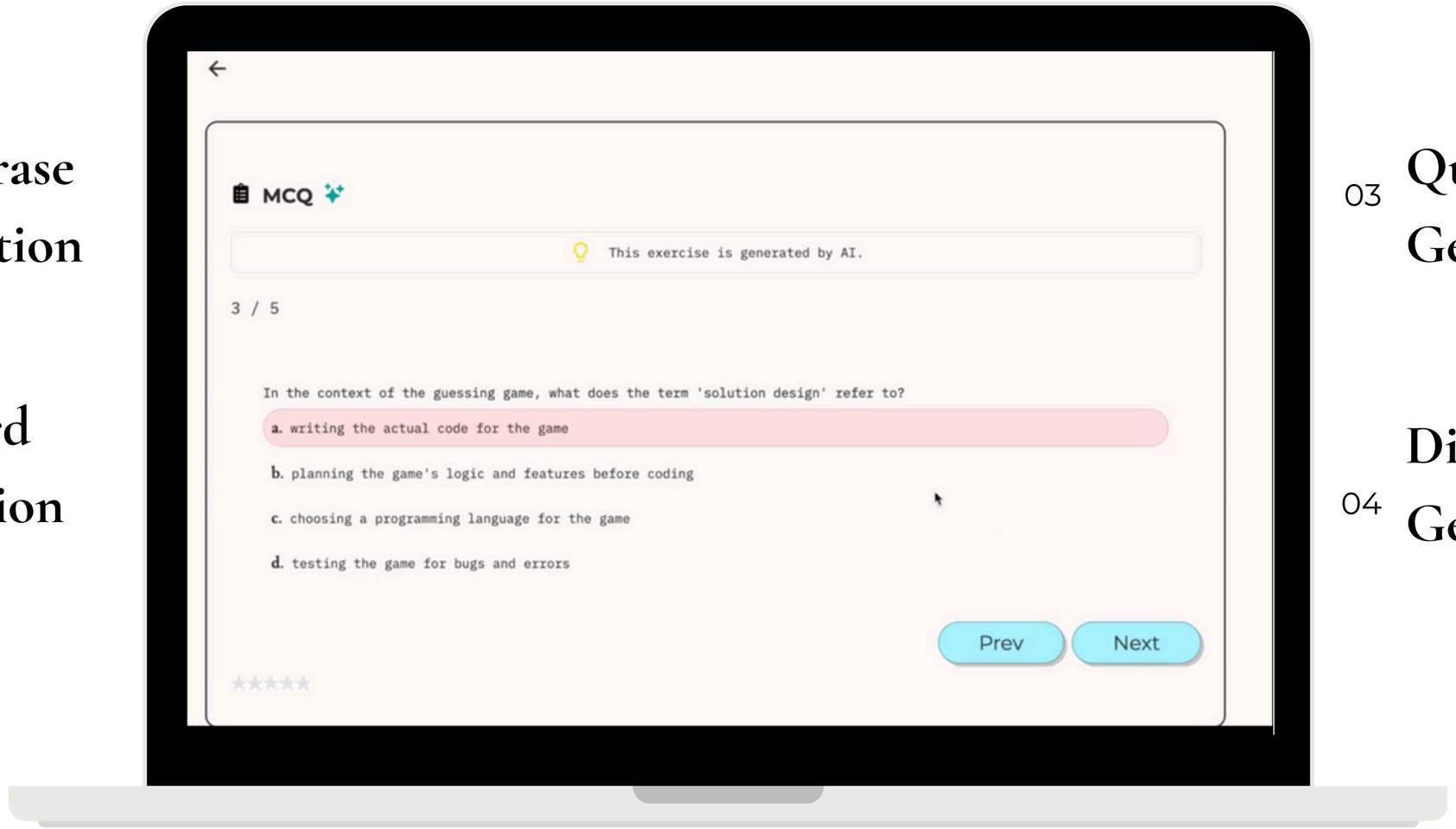
pleap

Multiple Choice Questions

A multi-stage approach

01 Paraphrase Generation

02 Keyword Extraction



03 Question Generation

04 Distractor Generation

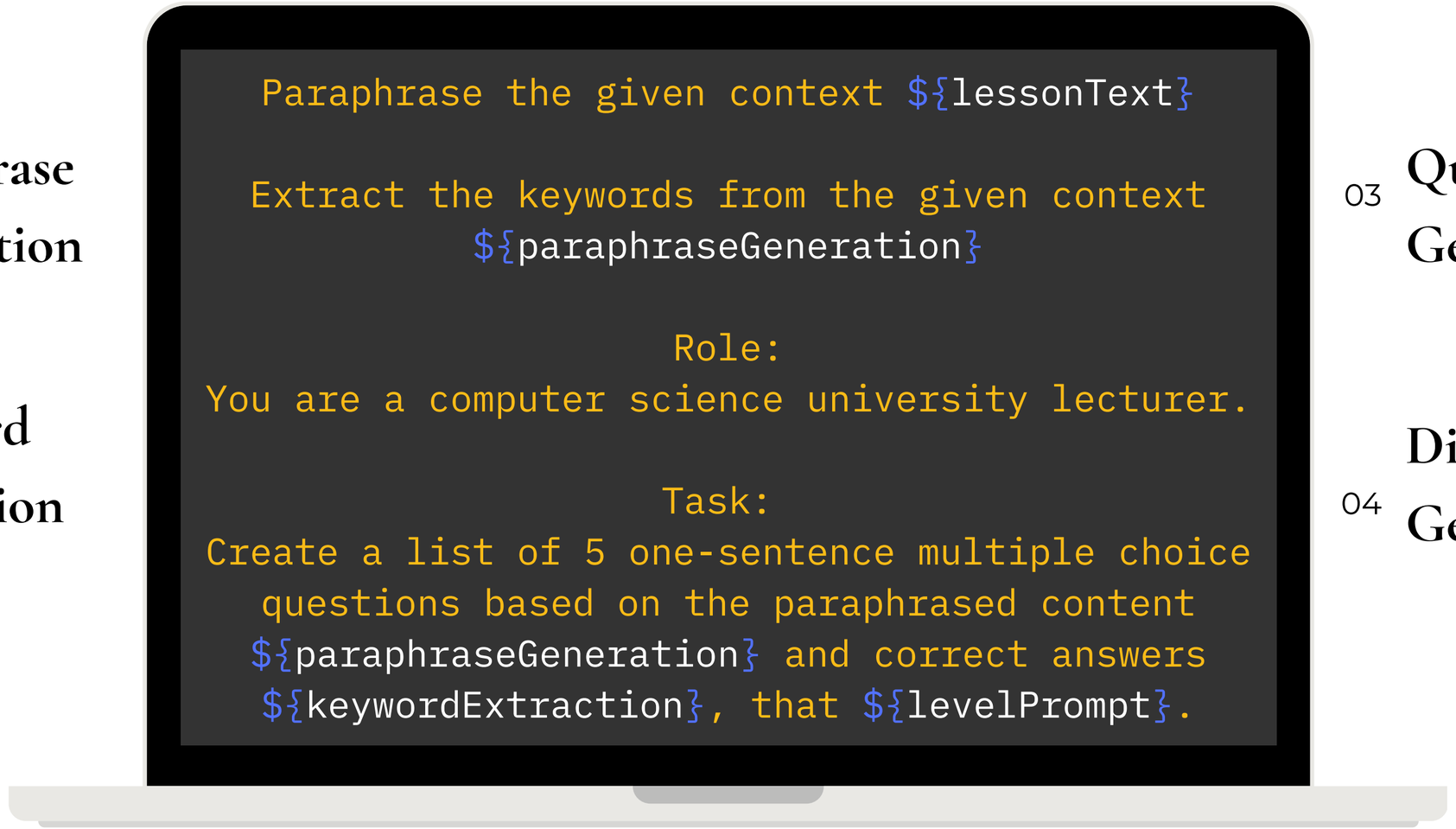
pleap

Multi-stage Prompting

MCQ Generation

01 Paraphrase Generation

02 Keyword Extraction



03 Question Generation

04 Distractor Generation

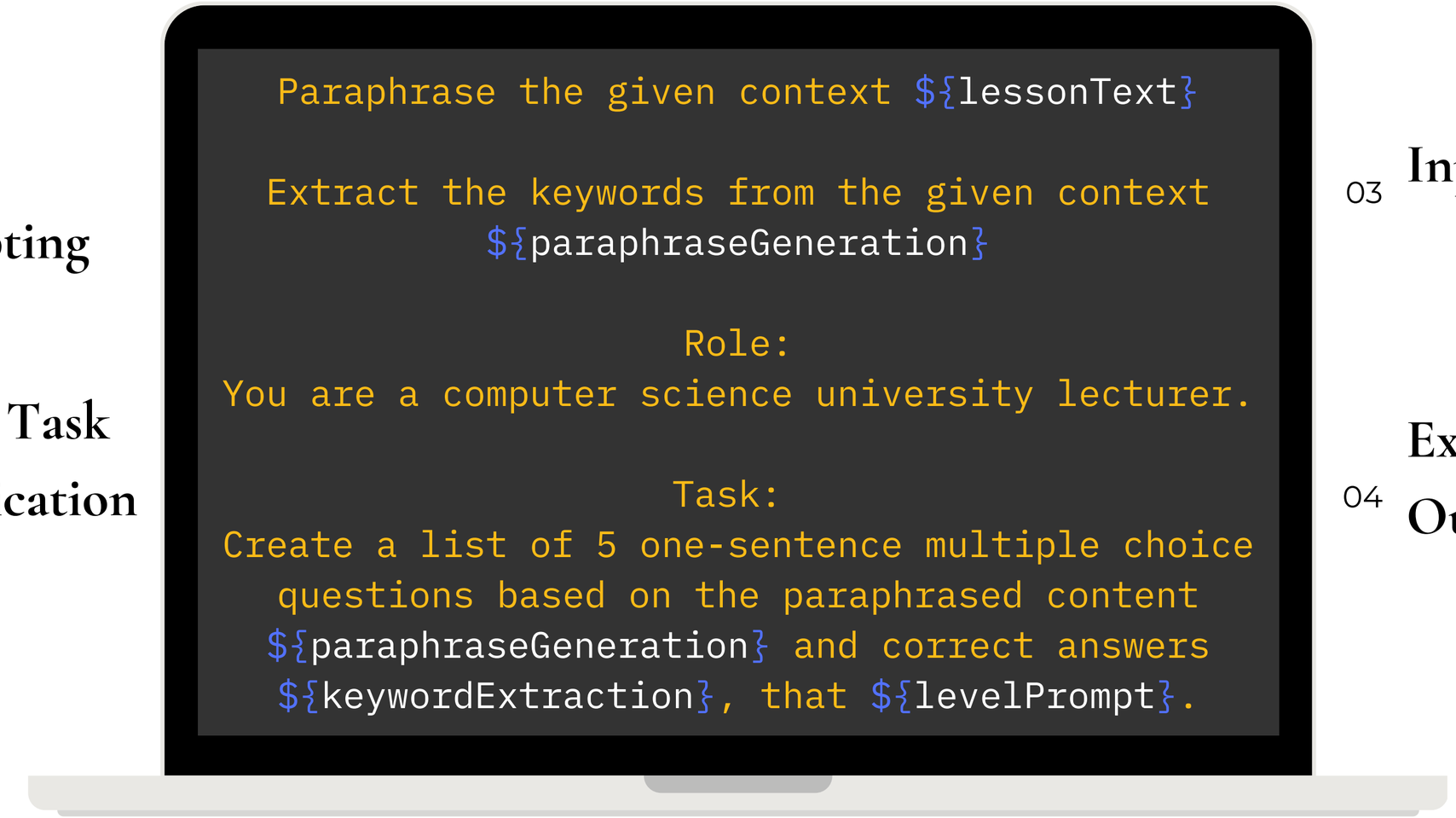
pleap

Meta Prompting

Structuring an effective prompt

01 Role Prompting

02 Direct Task Specification



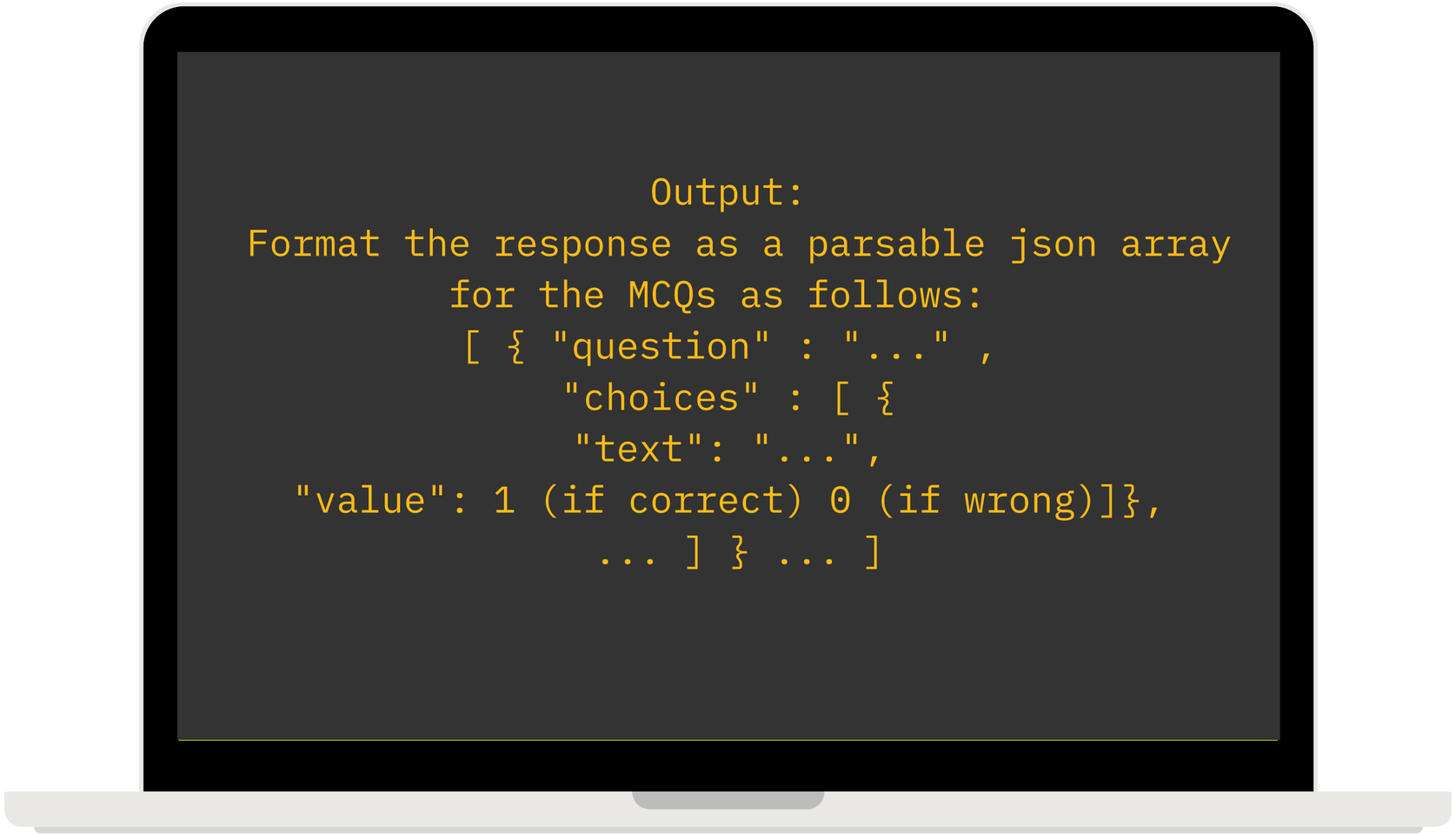
03 Input Data

04 Expected Outcome

pleap

JSON Formatting

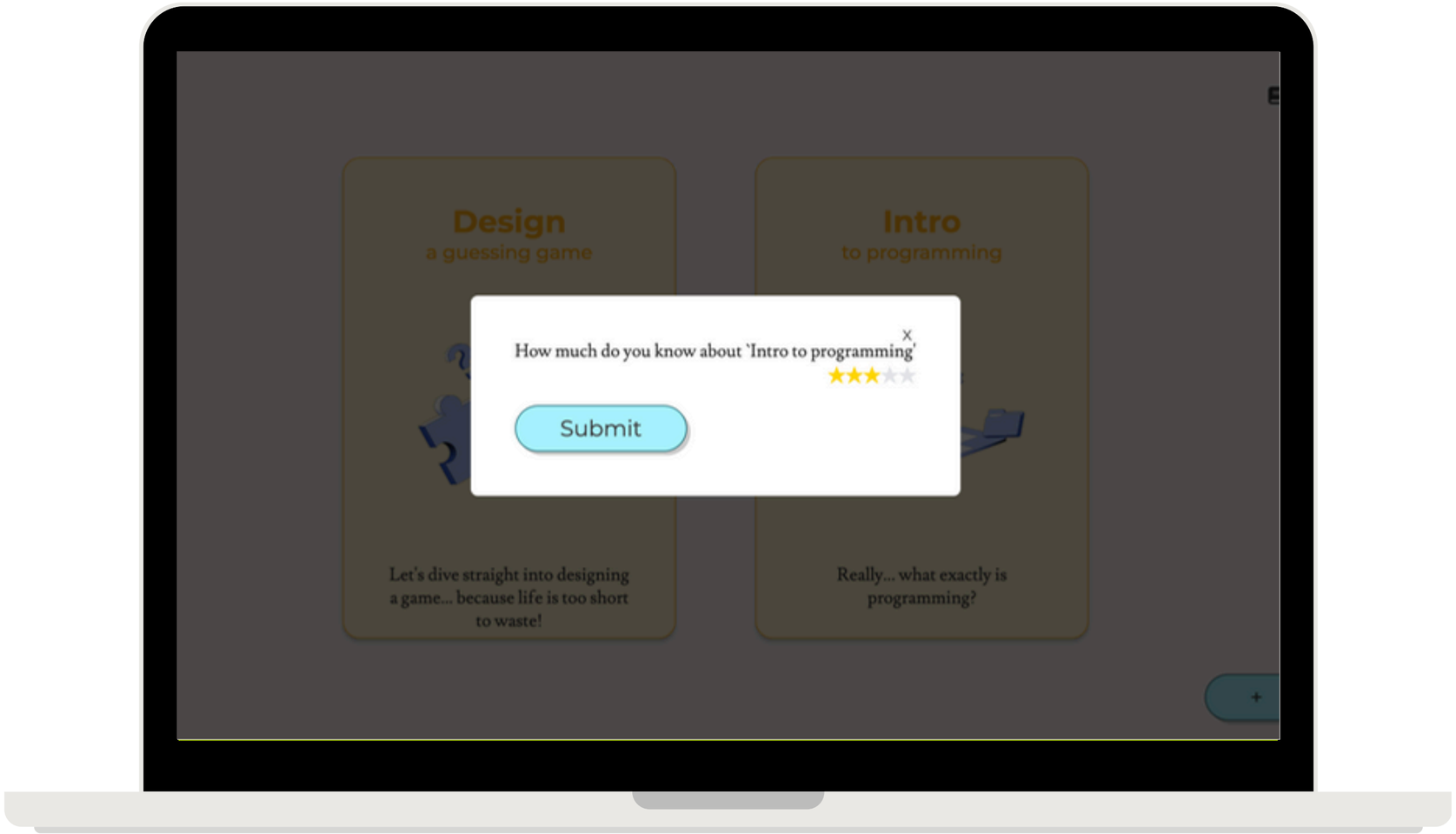
Prompt Output Formatting



pleap

Bloom’s Taxonomy

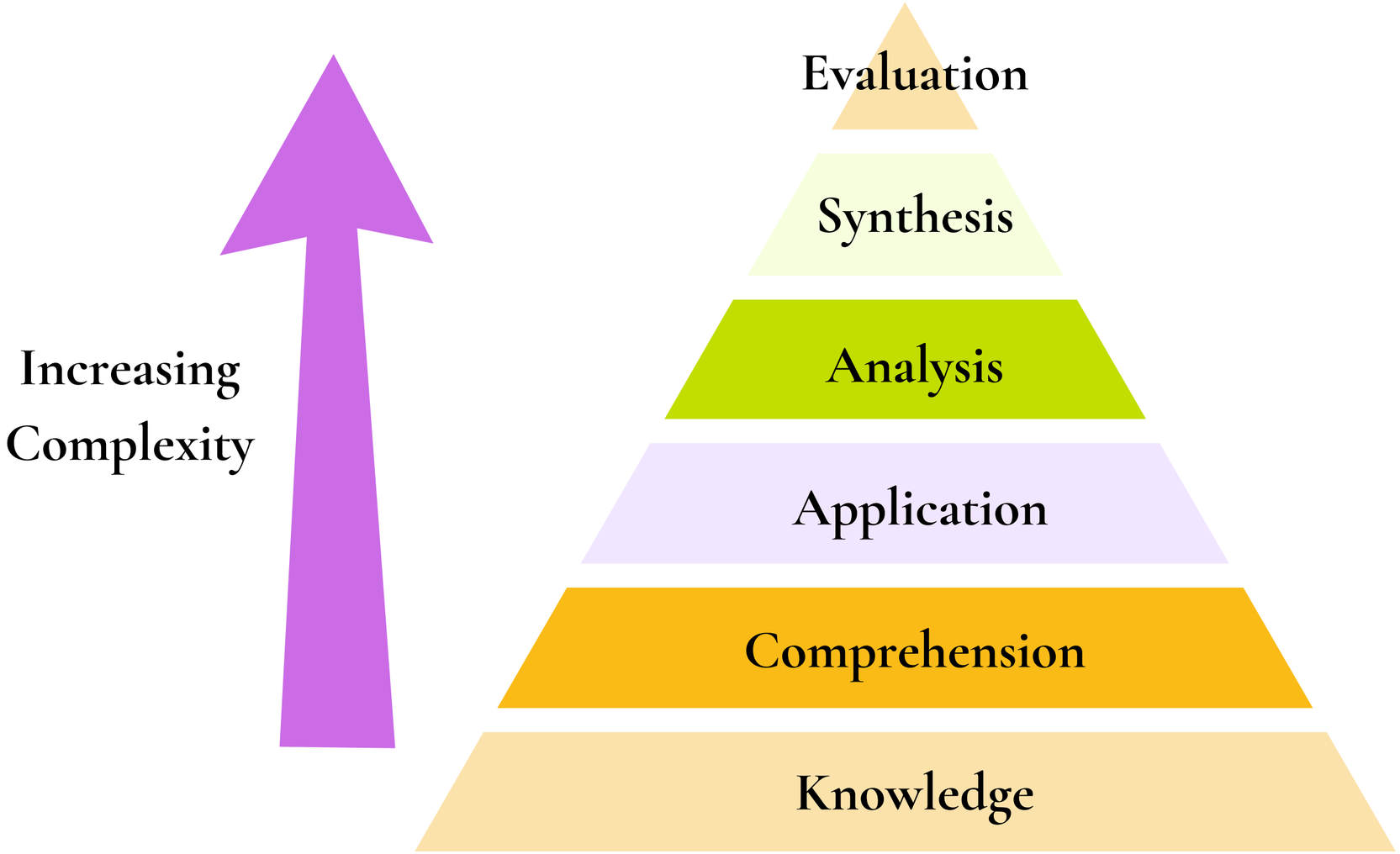
Test Different Levels of Understanding



pleap

Bloom's Taxonomy

Test Different Levels of Understanding



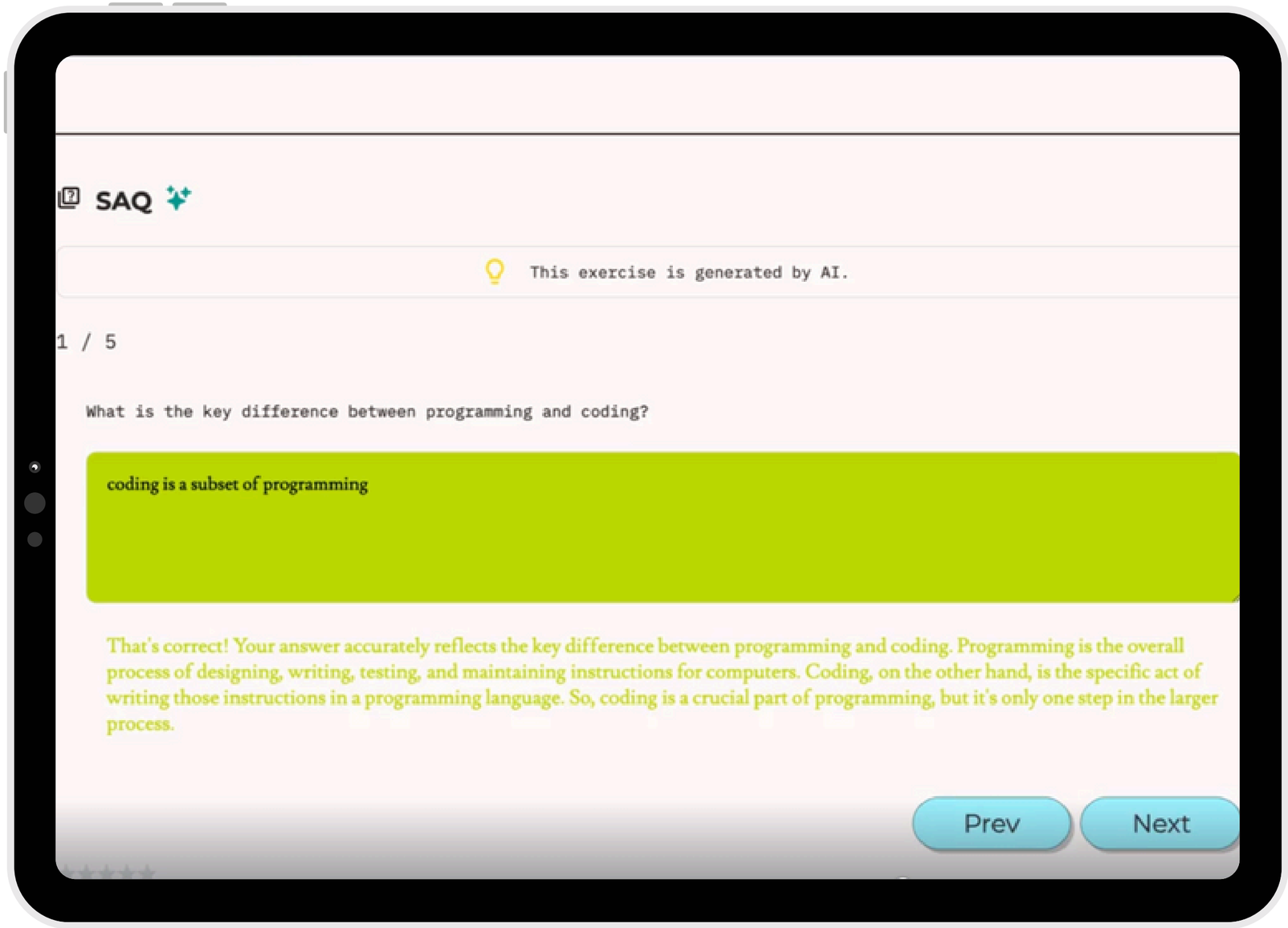
pleap

Bloom’s Taxonomy

Test Different Levels of Understanding

IEEE LOM Difficulty Level	Level Prompt
very easy	assess **recall** of core concepts through memory of terminology, facts, setting, methods
easy	assess **comprehension** by translation of attained knowledge to similar instances
medium	assess **application** of knowledge to new situations, problem-solving, and analysis
hard	assess **evaluation** of knowledge, **judgement**, and dissection, through argumentative exercises
very hard	assess **creation** of new knowledge, **synthesis** of information, and **design** of new patterns, by producing purposeful outcomes by rearranging patterns

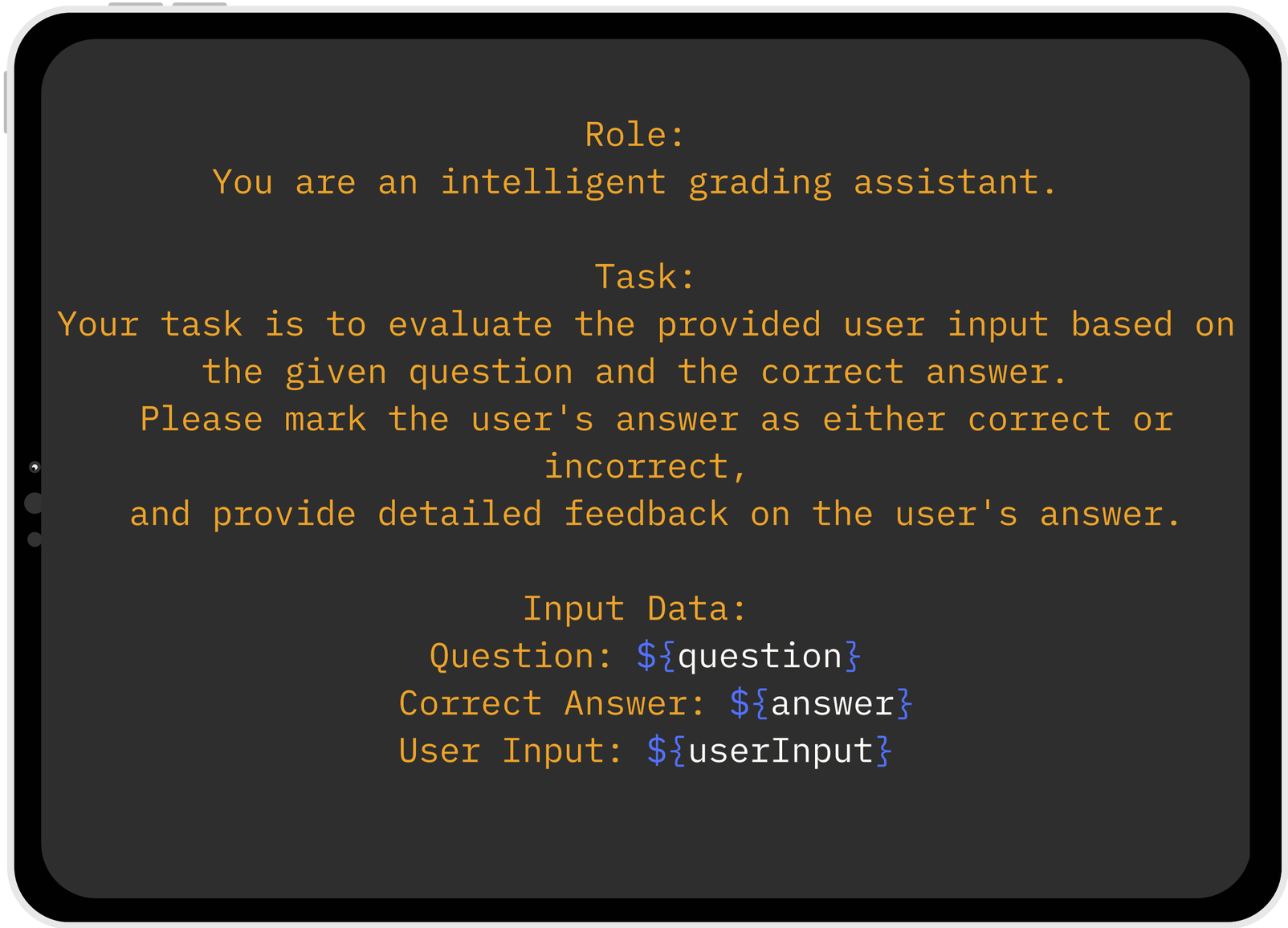
Question & Answer



01 Open-ended questions

02 AI-generated feedback (Based on answers aligned with lesson)

Intelligent Grading Assistant



01 Role Prompting

02 Direct Task Specification

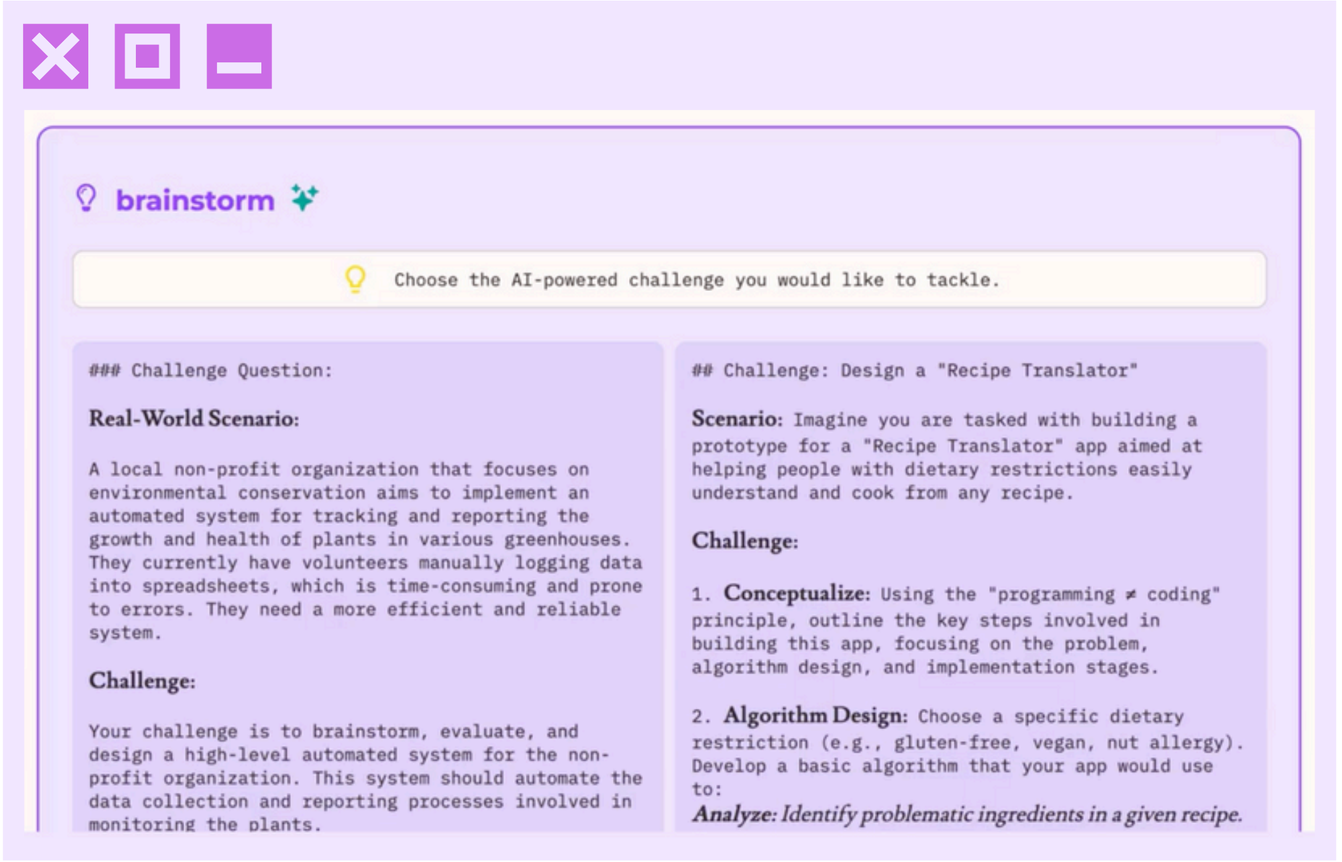
03 Input Data

04 Expected Outcome

pleap

Challenge

Brainstorm, Evaluate, Create



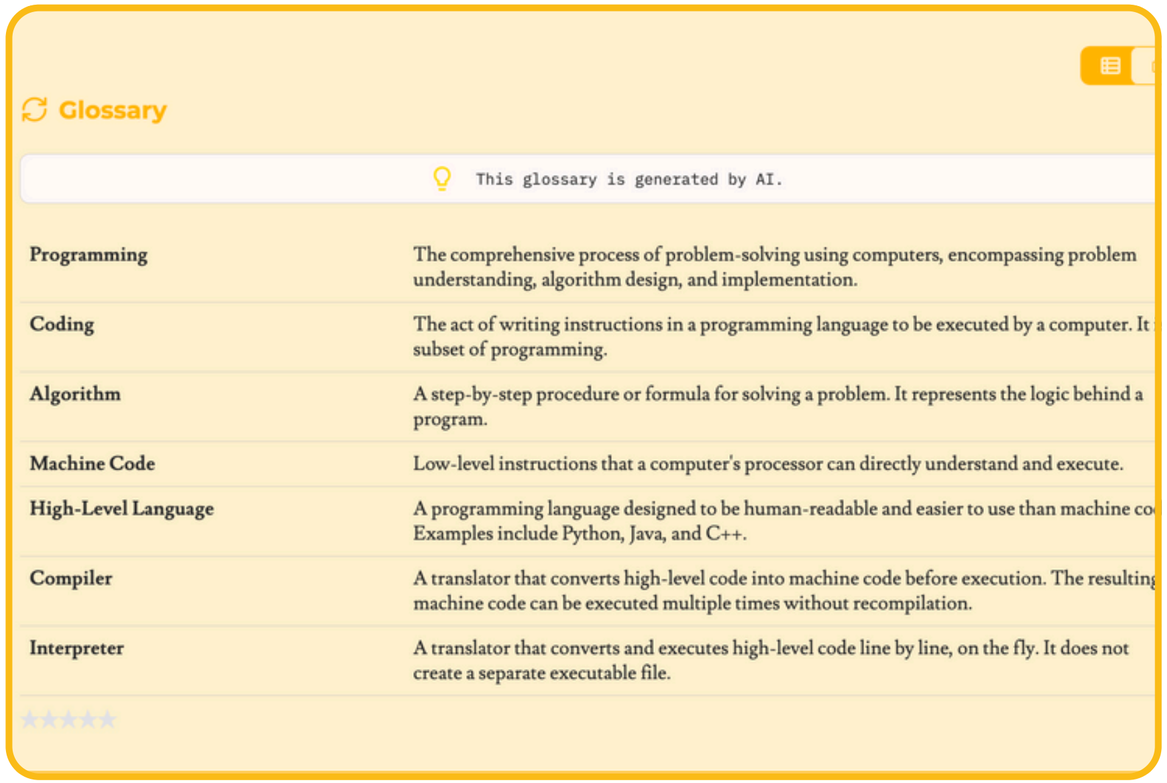
Gemini-1.5-Pro

GPT-4o

pleap

Glossary

Table Form / Index Cards



Learning Object Recommendation

Personalise display based on initialised learning preferences and user ratings.

Content-based Filtering

Predict ratings of new learning object by clustering previously rated learning objects by same user

K-means Clustering — Hamming Distance

Get Nearest Cluster — Cosine Similarity

Get Top-N in Nearest Cluster — Pearson Correlation

Predict New Learning Object Rating — Pearson Correlation

Collaborative Filtering

Predict ratings of new learning object based on ratings by other users on platform that share similar learning preferences.

K-means Clustering — Euclidean Distance

Get Nearest Cluster — Cosine Similarity

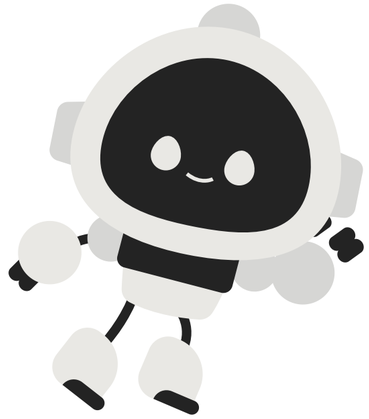
Get Top-N in Nearest Cluster — Pearson Correlation

Predict New Learning Object Rating — Pearson Correlation

Hybrid Filtering

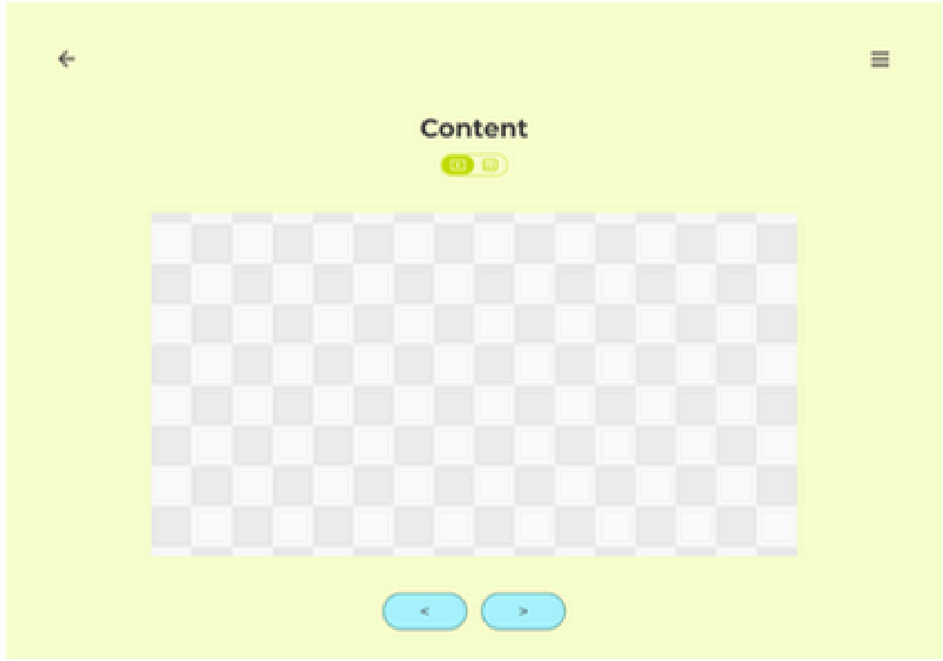
Predict ratings of new learning object based on both previous user ratings and learning preferences of other learners.

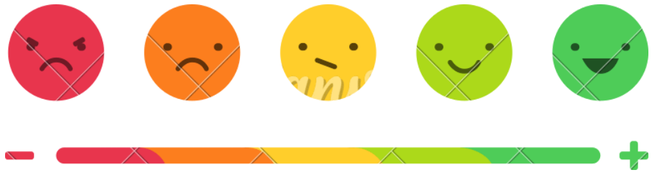
```
if (ratingsForLO.length === 0 && allRatingsByUser.length === 0) {
    predictedRating = predictInitialRating(userLS, loScore);
} else if (ratingsForLO.length === 0) {
    predictedRating = await calculateContentPrediction(allRatingsByUser, loScore, accessToken);
} else if (allRatingsByUser.length === 0) {
    predictedRating = calculateCollaborativePrediction(nearestClusterLS, userLS, ratingsForLO);
} else {
    const r1 = calculateCollaborativePrediction(nearestClusterLS, userLS, ratingsForLO);
    const r3 = await calculateContentPrediction(allRatingsByUser, loScore, accessToken);
    predictedRating = weight * r1 + (1 - weight) * r3;
}
```



Recommendation & Sequencing

- Exercise/Example-first
- Filter LOs above median rating





Evaluation

Verify **effectiveness** and **accuracy** of **filtering** algorithms, and assess **benefit** of **AI-generated** content.

Evaluation Setup

10 test subjects

Imperial College London

diverse courses and learning styles

20 learning objects per lesson

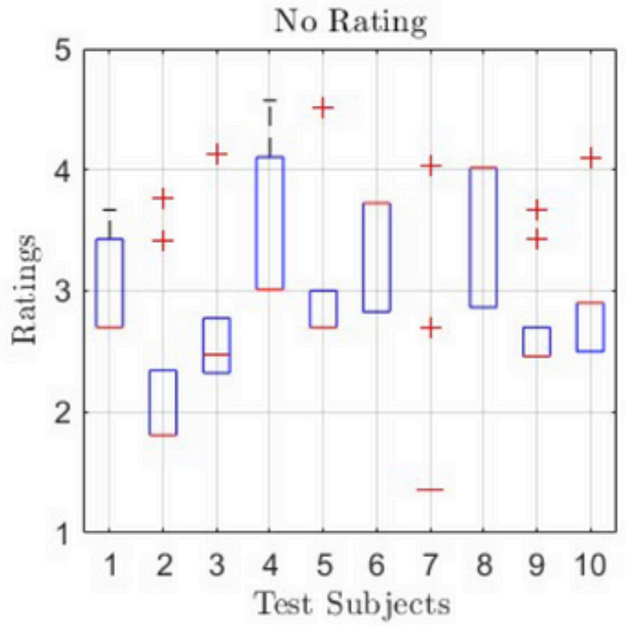
Test Subjects	1	2	3	4	5	6	7	8	9	10
Active	0.182	0.318	0.636	0.818	0.545	0.182	0.636	0.182	0.364	0.000
Sensing	0.636	0.636	0.909	0.909	0.636	1.000	0.364	0.909	0.545	0.364
Visual	0.636	0.955	0.909	0.636	0.727	0.818	0.909	0.636	0.818	0.455
Sequential	0.455	0.455	0.409	0.727	0.636	0.727	0.091	0.636	0.636	0.182

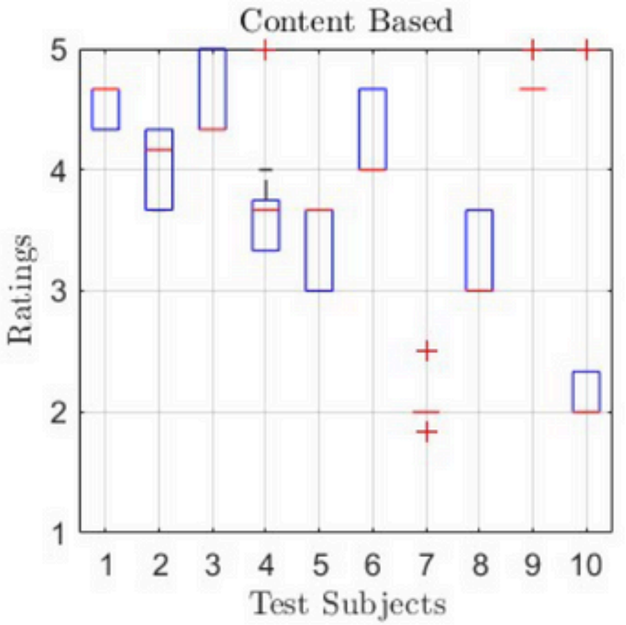
Table 6.1: Learning Preferences of the Test Subjects

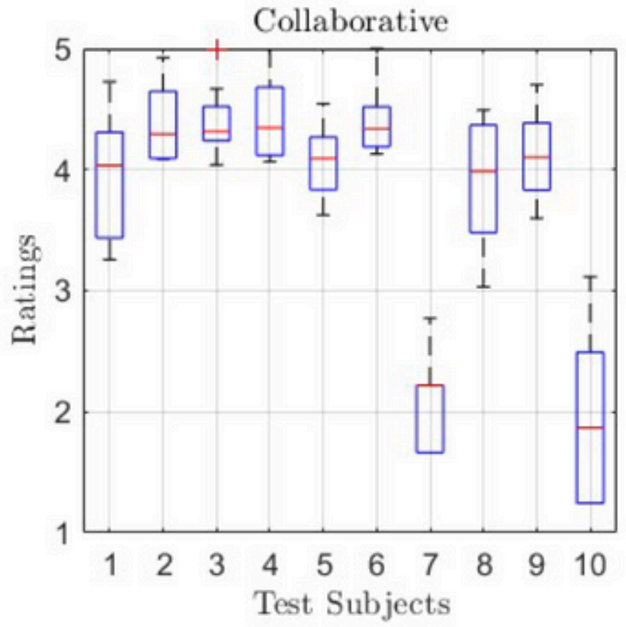
Note: The scores range from 0 to 1 for each FSLSM dimension. E.g: 0 indicates least active and most reflexive, 1 indicates least reflexive and most active (opposite polarity).

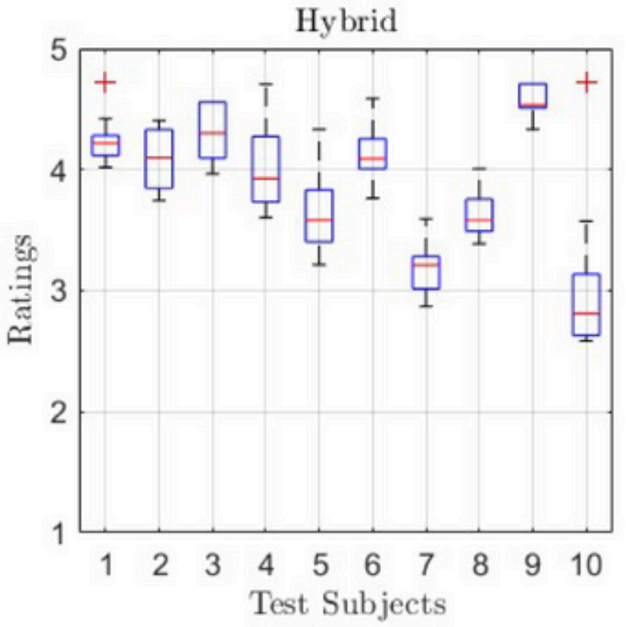
Comparison of Overall Ratings

Learning objects above median normalised by min-max normalisation



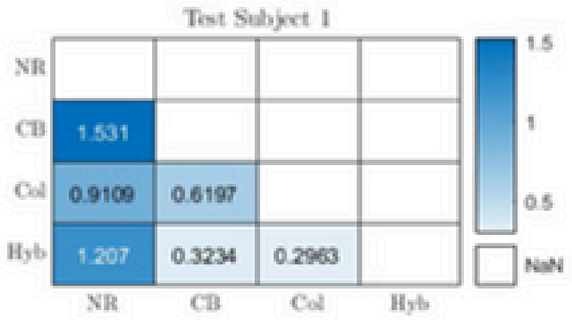


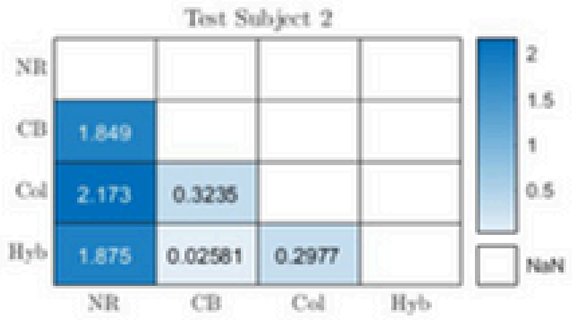


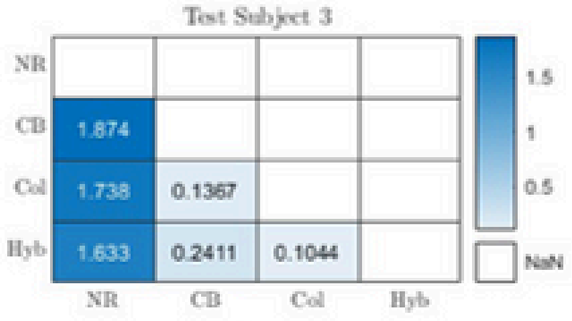


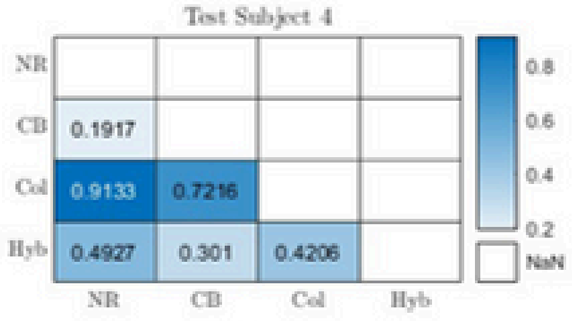
Effectiveness of Recommendation Algorithms

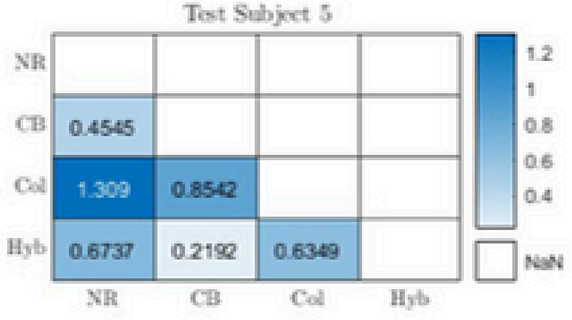
Tukey's Honest Significant Difference (HSD) Test for all 10 test subjects

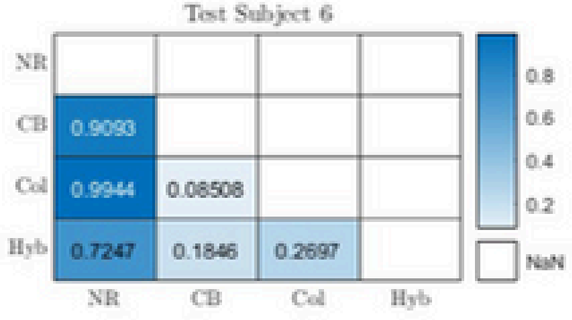


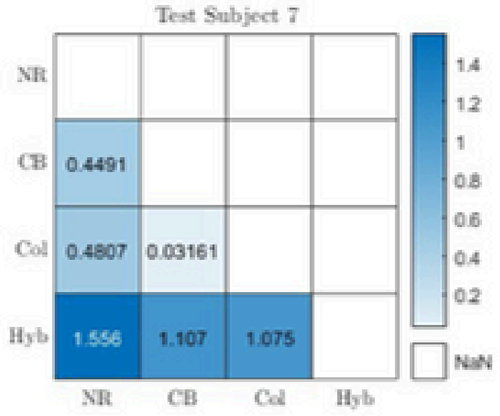


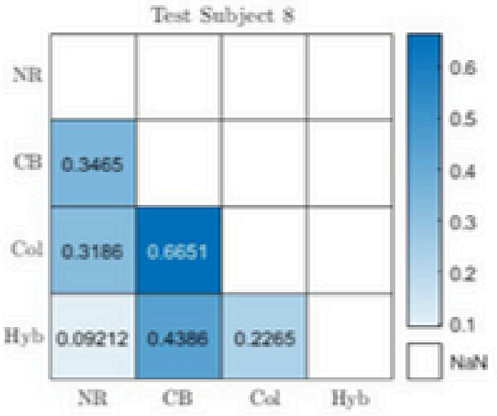


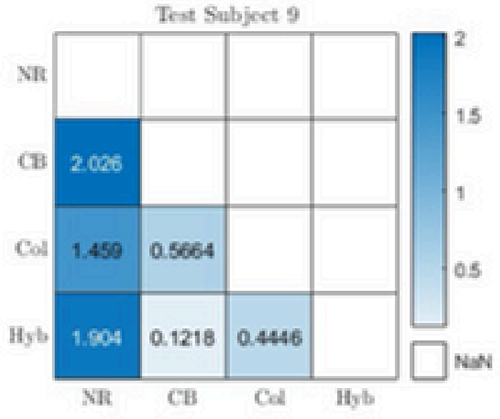


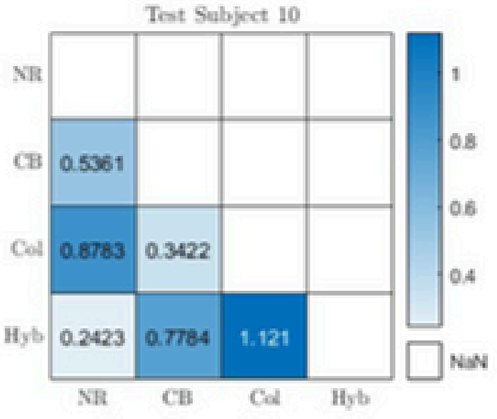












Mean Squared Error (MSE)

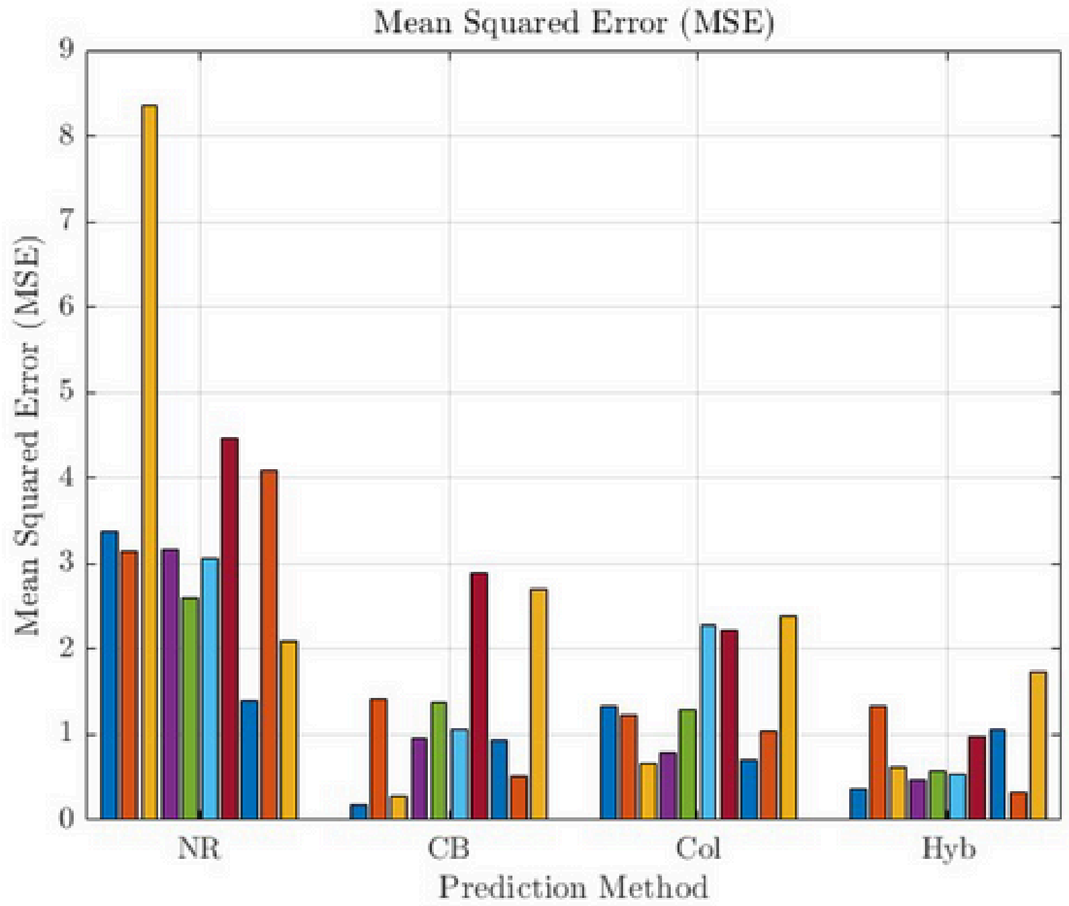


Figure 6.3: Mean squared errors between predicted ratings and actual ratings of recommended learning objects across four recommendation algorithms (No Rating, Content-Based, Collaborative, Hybrid).

AI-generated Content

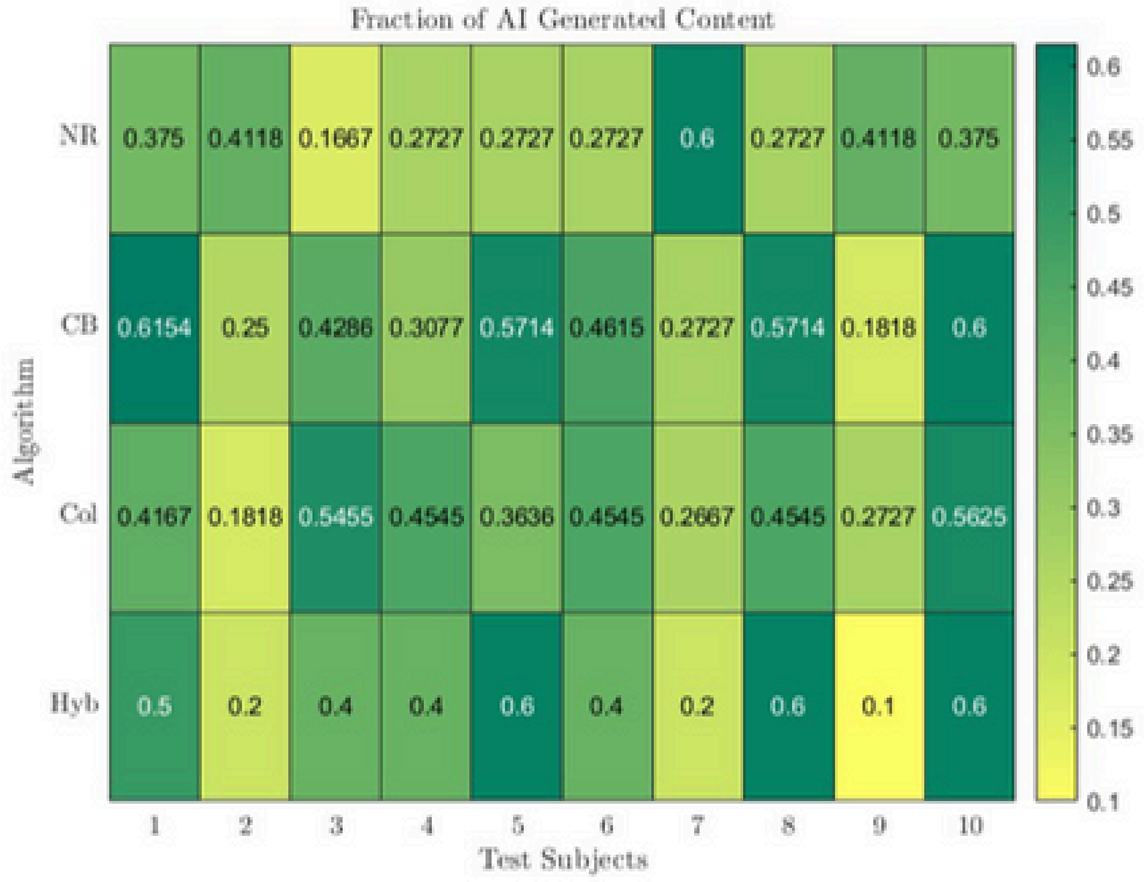


Figure 6.4: Fractions of AI-generated content in learning object recommendations for each test subject across four algorithms.

Conclusion

This project uncovered an effective personalised learning strategy using a **fully algorithmic recommendation technique** and enhanced the learning experience with AI through **automated content generation**.

Findings

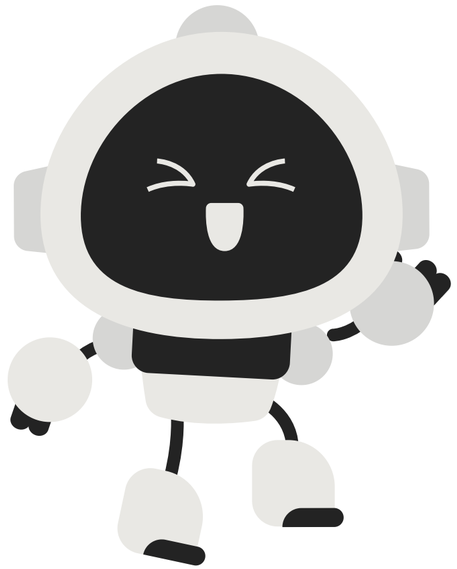
LLMs by OpenAI and VertexAI each have their **own strengths** in generating a diverse set of learning objects that cater to **various learning styles**

Hybrid filtering learning object recommendation (α = 0.5) provides best person-alisation, with **highest** and **most accurate** overall ratings

AI-generated content is **effective** in catering to diverse learning styles

Future Works

1. Probabilistic student model
2. Other classification algorithms
3. Text-to-image research
4. Learning path construction
5. Collaborative learning



thank you!