

Faculty of Science, Technology and Communication

## A Comprehensive Framework for Al-Driven Multiple-Choice Question Generation with Cognitive Taxonomy Alignment and minimal Item-Writing Flaws

Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master in Information and Computer Sciences

*Author:* Gilles Xiao Zhou Chen Supervisor: Prof. Dr. Steffen Rothkugel *Reviewer:* Prof. Dr. Volker Müller

> *Advisor:* Aryobarzan Atashpendar

August 2024

## Abstract

Creating multiple-choice questions (MCQs) is a complicated and time-consuming task, often plagued by common pitfalls such as implausible distractors and ambiguous language, which can hinder the effective assessment of a student's knowledge. This thesis presents a comprehensive framework that automates the creation of high-quality MCQs using GPT-40, where high quality is defined by the presence of minimal to no writing flaws. The development of the framework was supported by a thorough analysis of effective MCQ characteristics and state-of-the-art MCQ generation with GPT models. This framework can generate MCQs that target the first three cognitive levels of Bloom's Taxonomy: Remembering, Understanding, and Applying.

Manual inspection of 100 generated MCQs revealed that 96% were solvable, demonstrating a strong alignment between the generated questions and the content provided to GPT-40. Additionally, GPT-40 itself assessed the solvability of these MCQs, achieving 98% accuracy based on the same set of 100 questions. An automatic question evaluation tool is used to detect 19 MCQ Item-Writing Flaws (IWFs). Before deploying this tool to evaluate the MCQs, insights gained from researching language models were applied to enhance the tool's performance and accuracy. The results show that the MCQs generated with the new framework are consistently of high quality, though the practical and pedagogical effects were not tested, as this research did not involve direct testing with students or educators. Among the 360 generated MCQs, only 4.6% of 6,840 possible IWFs were present. When accounting for the tool's accuracy of the tool of 95.47%, this indicates that 91.07% of potential IWFs were successfully avoided in the generated MCQs, demonstrating the framework's effectiveness in producing high-quality questions.

This framework has the potential to significantly simplify the creation of high-quality MCQs, reducing the need for expert review as noted in numerous state-of-the-art studies, and saving educators time and effort. Additionally, a user-friendly interface was developed, making it remarkably straightforward to generate MCQs, thereby demonstrating the framework's practical application.

Keywords: Automated Multiple-Choice Question Generation, Artificial Intelligence in Education, Feedback and Distractors, Bloom's Taxonomy, Natural Language Processing (NLP), Generative AI, Prompt Engineering, Language Models, Hallucination Mitigation, Prompt Engineering Techniques, Generative pre-trained Transformer (GPT), Item-Writing Flaws (IWFs)

## **Declaration of Honor**

I hereby declare on my honor that I am the sole author of the present thesis. I have conducted all work connected with the thesis on my own.

I only used those resources that are referenced in the work. All formulations and concepts adopted literally or in their essential content from printed, unprinted or Internet sources have been cited according to the rules for academic work and identified by means of footnotes or other precise indications of source.

This thesis has not been presented to any other examination authority. The work is submitted in printed and electronic form.

Luxembourg, August 2024

Gilles Xiao Zhou Chen

## **Acknowledgments**

I would like to express my gratitude to my advisor, Aryobarzan Atashpendar, for his guidance during this research.

I am also grateful to my supervisor, Prof. Dr. Steffen Rothkugel, for his availability and willingness to provide support whenever needed.

Additionally, I would like to thank Prof. Dr. Volker Müller for his role as a reviewer and his forthcoming participation in my thesis defense.

Prof. Dr. Steffen Rothkugel, along with Prof. Dr. Volker Müller, were my teachers during my Bachelor's studies and were present at my Bachelor's defense. Their dedication and passion for teaching have always been a source of inspiration for me. I am glad that they will also be present for my Master's defense. It is a privilege to have their support and guidance throughout my educational journey.

I extend my sincere thanks to Steven James Moore from Carnegie Mellon University for allowing the use of SAQUET and providing a set of 100 MCQs along with the classification results from expert reviewers.

Finally, I extend my appreciation to my family and friends for their continuous support and encouragement during this journey.

## Contents

Ał	ostra	st	i
Co	onten	ts	ix
Lis	st of	Figures	xi
Lis	st of	Tables	xii
1	Intr	oduction	1
	1.1	Context	. 1
	1.2	Motivation	. 1
	1.3	Background	. 2
	1.4	Objectives	. 3
	1.5	Structure	. 3
2	The	oretical background	5
	2.1	Artificial Intelligence (AI)	. 5
	2.2	Machine Learning	. 5
	2.3	Natural Language Processing (NLP)	. 6
		2.3.1 NLP Techniques	. 7
	2.4	Generative Al	. 8
		2.4.1 Key Methods and Architectures in Generative AI	. 8
		2.4.2 Large Language Models (LLM)	. 10
		2.4.3 Generative Pre-trained Transformer (GPT)	. 11
	2.5	Conclusion	. 12
3	Stat	te of the art	15
	3.1	Advancements and Impact of the GPT Models	. 15
		3.1.1 GPT-3, GPT-4, GPT-4o the newer the better?	. 16
		3.1.2 Educational Potential of GPT-40	. 17
	3.2	Prompt Engineering Techniques and Patterns	. 17
		3.2.1 Prompt Patterns	. 18
		3.2.2 Relevant Prompt Patterns for the creation of MCQs	. 18
		3.2.3 Evaluating Prompt Patterns	. 20
		3.2.4 Prompting Techniques	. 21
		3.2.5 Conclusion	. 26
	3.3	What is a good multiple-choice-question?	. 27
		3.3.1 Characteristics of a Good MCQ	. 27
		3.3.2 Relevance and General Guidelines	. 31
		3.3.3 Determining the Validity of MCQs	. 31
	3.4	Al-Generated Multiple-Choice Questions	. 32

		3.4.1	Effectiveness and Efficiency	32	
		3.4.2	Current Applications of AI in Education and Potential	36	
		3.4.3	Conclusion	36	
4	Design and implementation 39				
	4.1	Decisio	on	39	
		4.1.1	Rationale	39	
	4.2	GPT N	Nodels and Parameters	39	
		4.2.1	GPT Parameters	39	
		4.2.2	GPT Message Types	41	
		4.2.3	GPT Models	41	
		4.2.4	Global GPT settings for this thesis	42	
	4.3	Phase	1: Model Choice, Formatting, and MCQ Creation	42	
		4.3.1	Naive Prompt Testing	42	
		4.3.2	Applying Prompt Engineering Techniques for Consistent Feedback		
			Format	45	
		4.3.3	Model Evaluation and Selection	50	
		4.3.4	Few-shot prompting, refinements, and MCQ Guidelines	51	
		4.3.5	Restricting Information with external source	56	
	4.4	Phase	2: Bloom's Taxonomy	57	
		4.4.1	Modifying the 3-shot examples	58	
		4.4.2	Implementing in Prompt	58	
		4.4.3	Results	59	
	4.5	Detern	nining MCQ Solvability	60	
	4.6	User Ir	nterface and Prompt Assembly Framework	61	
		4.6.1	User Interface	61	
		4.6.2	Framework to assemble the final MCQ generation Prompt $\ldots$ .	63	
5	Eval	uation		65	
	5.1	Using	GPT to determine solvability of MCQs	65	
		5.1.1	Research Question 1	65	
		5.1.2	Research Question 2	67	
	5.2	Assess	ing MCQ Quality with SAQUET	69	
		5.2.1	Research Question 3	71	
		5.2.2	Research Question 4	74	
6	Con	clusion		81	
	6.1	Discus	sion	81	
		6.1.1	Design and Implementation	81	
		6.1.2	Results	82	
	6.2	Limita	tions and Future Work	85	
		6.2.1	Limitations	85	
		6.2.2	Future Work	86	

#### Abbreviations

#### CONTENTS

Append	Appendix			
.1	Prompting Inputs & Outputs	91		
.2	Speed Tables for GPT models	99		
.3	Output Testing	101		
Bibliography 111				

## **List of Figures**

2.1	Neural Network using binary classification to recognize a car	6
2.2	Image Reconstruction using a Variational Autoencoder (VAE)	8
2.3	How GAN works	9
2.4	Simplified high-level workflow of LLM Models	11
3.1	GPT4o vs GPT4 in multilingual tasks. <b>Taken from</b> [3]	16
3.2	Three paradigms of Retrieval-Augmented Generation. Taken from Gao et	
	al. (2020), NeurIPS, $\textcircled{C}$ 2020 by the authors [26].	26
3.3	Meaningful stem	28
3.4	Not Meaningful stem	28
3.5	Examples of Meaningful and Not Meaningful Stems	28
3.6	The Stem contains irrelevant and potentially distracting/confusing infor-	
	mation	28
3.7	Multiple things are wrong in this MCQ. Distractors C and D are not plau- sible at all, meaning they are easily eliminated by test takers. Option A is the more widely recognized contributor, but Option B is technically cor- rect as well. There should only be one clear single correct answer to avoid ambiguity. One could even argue that the extinction of the Dinosaurs contributed to long-term climate changes.	29
3.8	Homogeneity is lacking, as not all options are authors or writers, they mix unrelated professions and historical figures, which will make it easy to identify J.K. Rowling as the correct answer without needing specific	
	knowledge of literature	30
3.9	In this example, the test-taker must rely on their knowledge of the authors rather than a process of elimination based on irrelevant options.	30
3.10	Non-homogeneity vs Homogeneity	30
3.11	The distractors overlap. Both Ibuprofen and Aspirin are used as a pain re- liever and NSAIDs contain both Ibuprofen and Aspirin, making this choice overlap with A) and D)	30
3.12	Comparing the generated MCQs with GPT with human-crafted ones. <b>Taken</b>	
	from [23]. ⓒ 2024 by Jacob Doughty et al., licensed under CC BY 4.0.	34
4.1	Illustration on how the value of the temperature skews the probability distribution of the next word. This is just an example and not based on real distribution numbers.	40
4.2	Form Page	61
4.3	- Result Page	61
4.4	Result Page: Solvability Checked	62
4.5	Final Prompt Assembly	63

#### LIST OF FIGURES

5.1	The 19 Item-Writing Flaw rubric criteria used by SAQUET. Taken from		
	Moore et al. (2024), AIED 2024 [45]		
5.2	Comparison of Matching Rates		
5.3	Reducing cognitive load by focusing on a single task		
6.1	Comparison of Matching Rates 84		
2	30 Human Crafted MCQs based on Geography, European Union $\ldots$ 101		
3	30 Human Crafted MCQs based on Data Structures $\hdots\hdot$		
4	50 generated MCQs based on Geography, European Union $\ \ldots \ \ldots \ \ldots \ 103$		
5	50 generated MCQs based on Data Structures $\hdots\hd$		

## **List of Tables**

3.1	Parameter counts for GPT-1 to GPT-4	16			
3.2	Categories of Prompt Patterns in Prompt Engineering. Taken from [72]	18			
3.3	Pros and Cons of Prompt Patterns 2				
4.1	Used Models (1 Token is roughly 4 characters, the average word length is				
	4.7 characters in English, so 1 Token is roughly $3/4$ of a word in English.)	42			
4.2	Model Average Speed Comparison Across Prompts. Each value is the				
	average of 5 measurements.	44			
4.3	Model Average Speed Comparison Across Prompts. Each value is the				
	average of 5 measurements.	50			
4.4	Model Average Speed Comparison Across Prompts. Each value is the				
	average of 5 measurements.	50			
4.5	Model Average Speed Comparison Across Prompts. Each value is the				
	average of 5 measurements	50			
5.1	GPT identifying solvability for <b>Geography</b> MCQs	66			
5.2	GPT identifying solvability for <b>Data Structures</b> MCQs	66			
5.3	GPT classifying the generated MCQs' solvability	67			
5.4	GPT classifying the generated MCQs' solvability	68			
5.5	Actual Results vs GPT Results	69			
5.6	Results of SAQUET for all 6 sets (View Sets here 5.2.2)	76			
5.7	Results of SAQUET grouped by $Subject/Topic$ and $Content$ or $No$ $Content$ .				
	DS = Data Structures, $LA = Linear Algebra$ , $GM = General Medicine$				
	$WC = With Content, NC = No Content \dots \dots \dots \dots \dots$	76			
5.8	Comparing the amount of faults for each Taxonomy Level and With Con-				
	tent/No Content $\ldots$	77			
5.9	How much $\%$ of the total flaws each category represents	77			
1	Model Speed Comparison for Prompt 1 - Basic MCQ Generation	99			
2	Model Speed Comparison for Prompt 2 - Streamlined Output	99			
3	Model Speed Comparison for Prompt 3 - Indicating Correct Answers	99			
4	Model Speed Comparison for Prompt 4 - Insightful Feedback				
		99			
5	Model Speed Comparison for Prompt 5 - Consistent Feedback Format				
		100			
6	Model Speed Comparison for Prompt 6 - Using Template Pattern				
		100			
7	Model Speed Comparison for Prompt 7 - Achieve JSON Format				
		100			

## **1** Introduction

Creating multiple-choice questions (MCQs) that provide a reliable and valid measure of students' abilities, free from common pitfalls such as ambiguous language, implausible distractors, and unintended clues, is a complicated and time-consuming task that requires expertise and careful consideration of various factors. In this thesis, "high-quality" MCQs refer to MCQs that have minimal to none of these common pitfalls, but it does not directly insinuate that these MCQs are optimal for educational applications.

This chapter introduces the research on automating the creation of high-quality MCQs using prompt engineering techniques and the text generation model GPT-40. This chapter sets the context for the study, explains the motivation behind the research, and provides an overview of the background and objectives of the study.

### 1.1 Context

Multiple-choice questions are a widely used assessment method in educational settings. They offer a convenient and efficient way to evaluate students' knowledge, understanding, and application of concepts. However, creating high-quality MCQs that accurately assess students' cognitive abilities is a labor-intensive and time-consuming process. They require educators or experts with a deep understanding of the subject matter and also the ability to craft questions that effectively target specific cognitive levels. One of the major challenges in creating high-quality MCQs is avoiding common item-writing flaws, such as ambiguous information, implausible distractors, and clues that can help students identify the correct option. These flaws negatively impact the validity and reliability of assessments, which ultimately affects students' learning outcomes.

## 1.2 Motivation

Despite the importance of MCQs in educational assessments, the process of creating MCQs is often manual, relying heavily on the expertise and experience of educators. This manual process can lead to inconsistencies in question quality, and variability in assessment outcomes, and is a significant burden on educators' time and resources. Additionally, the increasing demand for online learning has created a need for large-scale, high-quality assessments that can be delivered efficiently and effectively.

Therefore, there is a need to develop solutions that can automate the creation of highquality MCQs, reduce the workload of educators, and improve the overall quality and consistency of assessments. By leveraging the advances in natural language processing

#### 1.3. BACKGROUND

(NLP), machine learning (ML), and Generative AI, it is possible to develop automated question-generation systems that can create high-quality MCQs quickly and efficiently.

## 1.3 Background

Recent advances in natural language processing, machine learning, and Generative AI have enabled the development of automated question-generation systems that can create MCQs from text-based content. However, despite these advances, there are still several challenges and limitations associated with these systems.

One big concern with AI models like GPT-40 is the common language model phenomenon called "hallucination". When AI hallucinates, it generates nonsensical or contextually irrelevant content to the provided instructions. This leads to MCQs that are not relevant to the assessment context or even MCQs that are based on false information. Many advancements to tackle hallucination are being made and it is still an ongoing research topic, but this challenge can be mitigated significantly with advanced prompt engineering techniques, reducing the likelihood of hallucination.

Another concern is the quality of the resulting MCQs. These MCQs often do not meet quality standards, as they may contain multiple answers, ambiguous statements, and irrelevant/bad distractors. In this research, few-shot prompting will be used to provide clear examples of high-quality MCQs, along with a set of guidelines on what makes a good MCQ. The purpose of providing these guidelines and examples is to improve the overall quality of the generated MCQs.

Finally, the human intervention that is currently needed to review the generated MCQs is still very apparent, as highlighted as a limitation and recommendation by the vast majority of studies. The quality of the output must be verified, but it is possible to significantly reduce this work by using an automatic question usability evaluation tool. These state-of-the-art evaluation tools show impressive capabilities in the detection of Item-Writing Flaws (IWFs) but can struggle with the detection of certain Item-Writing Flaws.

In summary, while recent advances in NLP, ML, and Generative AI have enabled the development of automated question-generation systems, significant challenges and limitations remain. Hallucination, poor question quality, and the need for human intervention are major concerns that hinder the widespread adoption of these systems. However, by leveraging advanced prompt engineering techniques, few-shot prompting, and automated evaluation tools, it is possible to mitigate these challenges and improve the overall quality and relevance of generated MCQs. This research aims to address these limitations by refining an existing evaluation tool and utilizing it in conjunction with carefully designed prompts and guidelines to produce high-quality MCQs that meet specific quality standards, ultimately reducing the need for human intervention and paving the way for more efficient and effective assessment practices.

## 1.4 Objectives

The primary objective of this research is to develop a comprehensive framework for automating the creation of high-quality MCQs using prompt engineering techniques and GPT-40. Specifically, this study aims to:

- Develop a framework that generates MCQs that target specific cognitive levels and have a minimum amount of item-writing flaws.
- Refine an existing evaluation tool to improve its effectiveness in assessing MCQ quality.
- Evaluate the usability of the generated MCQs using the refined evaluation tool.

## 1.5 Structure

This thesis is organized into several chapters, each addressing a key aspect of the research:

- Chapter 1: Introduction Provides an overview of the research context, motivation, background, and objectives.
- Chapter 2: Theoretical Background Discusses the foundational concepts of artificial intelligence, machine learning, natural language processing, and generative AI, including key methods and architectures.
- Chapter 3: State of the Art Reviews advancements in GPT models, prompt engineering techniques, characteristics of effective MCQs, and current research on the generation of MCQs with GPT.
- Chapter 4: Design and Implementation Details the decision-making process, model parameters, iterations and refinements of prompts, and the final framework.
- Chapter 5: Evaluation Describes the methods used to assess the quality and solvability of the generated MCQs, including the use and refining of the evaluation tool.
- Chapter 6: Conclusion Summarizes the findings, discusses the implications, and outlines limitations and future work.

#### 1.5. STRUCTURE

## 2 | Theoretical background

The purpose of this chapter is to provide the theoretical background for this thesis. It focuses on a broad overview of Artificial Intelligence (AI) and the following subsets within the field of AI: Machine Learning (ML), Natural Language Processing (NLP), Large Language Models (LLM), and Generative AI. These interconnected topics are the foundation of modern AI-driven language processing and text generation technologies like the Generative Pre-trained Transformer (GPT).

## 2.1 Artificial Intelligence (AI)

Artificial Intelligence (AI) aims to simulate and reproduce human intelligence in machines. AI has many subfields, such as machine learning and natural language processing, which were important for the development of Large Language Models. The main purpose of AI is to create systems that are capable of performing actions (understanding natural language, recognizing patterns, etc.) that typically require human intelligence.

Al is divided into two categories: narrow and general Al. General Al is the ultimate goal of Al research. It aims to give Al the ability to perform any intellectual task that a human can do, meaning that general Al would be able to understand, learn, and apply knowledge in similar cognitive processes as humans do. This field is still largely theoretical and extremely far from reach, but it would enable machines to perform well across a vast range of domains and activities [25].

Narrow Al's purpose is to perform specific tasks. These specific tasks could range from speech recognition, and language translation to image classification. Narrow Al operates under strict constraints and is specialized to perform within predefined functions. Examples include virtual assistants like Siri and Alexa, and chatbots [25].

The following sections are dedicated to exploring the various subsets of AI and how they contributed to innovative and truly groundbreaking creations such as the Generative Pre-trained Transformer.

## 2.2 Machine Learning

Machine Learning (ML) is a subset of AI that trains algorithms to be able to recognize patterns and make predictions based on data. For example, ML is used in financial statement fraud detection systems where they can recognize transaction patterns to identify suspicious activity [56]. There are different techniques used within Machine Learning, namely supervised learning, unsupervised learning, reinforcement learning, and deep learn-

#### 2.3. NATURAL LANGUAGE PROCESSING (NLP)

ing which use neural networks with multiple layers.

**Supervised learning** involves training a model using labeled data, where each input is paired with a known output [63]. For example, a dataset containing labeled images of elephants, camels, and cows would enable the model to correctly identify new examples based on the labeled data it has learned from.

**Unsupervised learning**, trains a model on unlabeled data to find hidden patterns within the data [63]. An example of unsupervised learning is customer segmentation in marketing, where customers are grouped based on purchasing behavior without pre-labeled categories.

**Deep learning** uses neural networks with multiple layers, also known as deep neural networks, (see Figure 2.2) to model complex patterns in data. Neural networks are a set of algorithms that are designed to recognize patterns [63]. Data is interpreted through machine perception, labeling, or clustering of raw input. Deep learning models are particularly effective in handling large volumes of data and have been crucial in achieving state-of-the-art performance in tasks such as natural language processing, image recognition, and also autonomous systems [69].



Figure 2.1: Neural Network using binary classification to recognize a car

**Reinforcement learning** is used to train an agent to make a sequence of decisions. The agent is rewarded for desirable actions and punished for undesirable actions, thus leading to the agent learning to maximize cumulative rewards over time [63]. An example of an agent trained through reinforcement learning is AlphaGo. AlphaGo learned to play the board game "Go" through deep neural networks, supervised learning from human expert games, and reinforcement learning to achieve mastery [67].

## 2.3 Natural Language Processing (NLP)

Natural Language Processing (NLP) focuses on the interaction between computers and natural (human) languages. Its' objective is to enable machines to comprehensively

#### CHAPTER 2. THEORETICAL BACKGROUND

process and generate human language. NLP applications may include translation, speech recognition, sentiment analysis, and summarizing text. All of these tasks require extremely sophisticated algorithms that can process and subsequently analyze large amounts of natural language data [30].

### 2.3.1 NLP Techniques

Different techniques were used within NLP to handle different tasks:

**Rule-based Approaches** were heavily used in Early NLP systems. This involved the creation of clear step-by-step linguistic rules to process language data, which was effective for specific applications, but this approach was naturally limited to handle the complexity and variability of natural language [36].

**Statistical Methods** gained popularity in NLP as computing became more powerful and large datasets became available. These methods use statistical models to learn patterns from large amounts of text data. Unlike rule-based approaches, statistical methods learn from actual textual examples [36].

#### Machine Learning in NLP

Machine Learning [see section 2.2] techniques are heavily used by modern natural language processing to process and understand language. Text classification, sentiment analysis, and named entity recognition enhance NLP systems' ability to interpret large amounts of textual data. For example, text classification assigns categories to text, useful in spam detection for emails, while sentiment analysis determines the sentiment behind a string of text, and named entity recognition identifies and classifies entities like names and companies [36].

**Deep learning** techniques such as Convolutional Neural Networks (CNNs) and recurrent Neural Networks (RNNs), have enabled more accurate and context-aware language understanding and generation. CNNs, initially popular in image processing, have been adapted for text classification, and RNNs are useful for processing sequential data in tasks like language modeling and machine translation [36].

**Transformers** provide a new approach to understanding and generating natural language. The transformer architecture, introduced by Google in 2017 [70], utilizes selfattention mechanisms, allowing them to weigh the importance of different words in a sentence relative to each other, regardless of their position within the text [70]. This mechanism enables transformers to identify long-range dependencies and relationships in a given text, overcoming the limitations of earlier models like RNNs that struggled with longterm context [36]. This capability has led to more accurate and context-aware language processing, paving the way for advanced models such as BERT (Bidirectional Encoder Representations from Transformers) [36] and GPT (Generative Pre-trained Transformer) [52]. These models excel in various NLP tasks, including translation and summarizing texts [70].

### 2.4 Generative AI

Generative AI's objective is to create new content such as text, images, music, and more. Sophisticated models and architectures are deployed to generate data that is similar to the training data it was exposed to. Generative AI models include image models like Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs), and text generation models like BERT and GPT [10].

### 2.4.1 Key Methods and Architectures in Generative AI

A key technique in generative AI involves using neural networks to recreate and understand complex patterns that can be found in data. One popular method is the VAE method which teaches models to compress input data into a simpler, hidden space that is called the latent space and then to expand it back to its original form. In this process, the model can generate new and different data samples by taking random points from this latent space and decoding them back into the original data format. A combination of neural networks and probabilistic modeling is used to create realistic data. This method is especially valuable for generating images, where the aim is to produce variations of high-quality images that are supposed to look similar to the ones used for training [10].



Figure 2.2: Image Reconstruction using a Variational Autoencoder (VAE)

#### CHAPTER 2. THEORETICAL BACKGROUND

Generative Adversarial Networks (GANs) utilize a unique training mechanism involving two neural networks: the generator and the discriminator (see Figure 2.3). The generator uses random input to create synthetic data samples. The discriminator then evaluates the authenticity of these synthetic data samples while also evaluating real data samples. The generator's objective is to produce data that can deceive the discrimination. The discriminator wants to be able to distinguish between real and synthetic data. This dynamic allows the neural networks to improve each other iteratively [27].

The discriminator provides loss feedback to both the generator and the Discriminator. The generator receives feedback on how well its synthetic data fools the discriminator, such that it can adjust and produce more realistic data. The discriminator receives feedback based on how accurately it can distinguish real data from synthetic data, such that it can make adjustments to improve its classification performance. This competitive setup leads to the generation of highly realistic images [27].



Figure 2.3: How GAN works

### 2.4.2 Large Language Models (LLM)

The concept of large language models (LLMs) has been around for decades but was relatively simple and limited in scope. The combination of Deep Learning, NLP, and the Transformer Model made this concept a reality. LLMs are a significant advancement in Aldriven language understanding and generation. BERT (Bidirectional Encoder Representations from Transformers) by Google and the GPT (Generative Pre-trained Transformer) Series by OpenAI are notable examples of modern LLMs [10, 31].

LLMs are a category of their own within generative AI, as their focus is specifically on generating and understanding human language. Unlike VAEs and GANs, which are often used for generating images or other types of media, the role of LLMs is to use the Transformer architecture to process and generate highly coherent and contextually relevant human-like text [73].

The self-attention mechanism of the Transformer [2.3.1] is a key difference from other architectures such as the VAE and GAN. It allows models to assign different levels of importance to words in an input sequence relative to each other, improving contextual understanding and coherence of generated text [31]. The Transformer's encoder-decoder structure processes all words in the input sequence simultaneously, improving both context comprehension and modeling efficiency [73].

LLMs undergo an unsupervised pre-training phase (2.2) and a supervised fine-tuning phase. During the former, the model is exposed to a large amount of text data, helping it to learn language patterns, grammar, and context without any labeled data. This broad knowledge base is important for the model's performance during the fine-tuning phase. During the supervised fine-tuning phase (2.2), the LLM's abilities are refined by training it on smaller, labeled datasets that are tailored to specific tasks such as translation and summarizing text [31].



Figure 2.4: Simplified high-level workflow of LLM Models

### 2.4.3 Generative Pre-trained Transformer (GPT)

The Generative Pre-trained Transformer (GPT), developed by OpenAI, is one of the biggest innovations in the field of LLMs to date. The core innovation of GPT is the utilization of the Transformer architecture, specifically the decoder-only structure, which is great at generating human-like text [57].

Just like LLMs, the GPT models go through two key training phases: self-supervised pre-training, which is a type of unsupervised learning where the data itself provides supervision, and supervised fine-tuning (similar to Figure 2.4). In the pre-training phase, the model is exposed to a large amount of text data, including internet data, books, and articles. With the help of the decoder-only structure of the Transformer, this self-supervised learning phase teaches the model to iteratively predict the next word in a sentence based on the preceding context. This enables GPT models to achieve a broad and deep understanding of language, including grammar, facts, and reasoning, resulting in generated text that is often indistinguishable from the writing of a human. During this pre-training phase, a method called maximum likelihood estimation is used to help the model adjust its settings, helping it to find complex patterns and connections in the data.

#### 2.5. CONCLUSION

Then, in the supervised fine-tuning phase, the GPT models are trained on more specific and labeled datasets, tailoring them to more particular tasks such as translation and summarization. Fine-tuning involves the adjustment of model parameters to improve performance on these specific tasks, enhancing the contextual relevance and accuracy of the generated output. This phase also employs reinforcement learning to further refine the model's output by having human reviewers evaluate it. The feedback is then used to train a reward model. The main model is then fine-tuned using this reward model, directly optimizing and influencing its performance based on human preferences. This helps to ensure that the GPT model output is closely aligned with human expectations [57].

From each iteration of the GPT models, there has been a substantial and exponential increase in the number of parameters, leading to significant improvements from model to model. A parameter refers to a numerical value within the model that is adjusted during training to optimize performance. These include weights and biases, which are essential for neural network operations. The self-attention mechanism of the Transformer architecture also has parameters that determine the focus on different parts of the input sequence. Each layer in the model, including feed-forward neural networks and normalization layers, has its own parameters as well. Embedding matrices, which convert input tokens into high-dimensional vectors capturing semantic information, also consist of parameters. Finally, the output layer has parameters responsible for predicting the probabilities of the next word. During training, algorithms like gradient descent adjust all these parameters to minimize prediction errors [15].

### 2.5 Conclusion

This chapter has outlined the theoretical foundation necessary for understanding the advancements in Artificial Intelligence and its subsets, including Machine Learning, Natural Language Processing, Large Language Models, and Generative AI. These components form the core of modern AI-driven language processing and text generation technologies, as demonstrated by the Generative Pre-trained Transformer.

We learned that AI is divided into narrow AI, focused on specific tasks like virtual assistants, and that general AI is a theoretical goal aiming to mimic human intellectual abilities in various tasks. Within AI, Machine Learning introduces techniques such as supervised, unsupervised, and reinforcement learning, alongside deep learning, which enables systems to detect patterns and make predictions.

Natural Language Processing enables the interaction between computers and human language, going from rule-based systems to sophisticated architectures like transformers. The introduction of transformers, with their self-attention mechanisms, has significantly enhanced the ability to process and generate human-like text, leading to the development of models such as BERT and GPT.

Generative AI extends the capabilities of AI to the creation of new content. Techniques like Variational Autoencoders (VAEs) and Generative Adversarial Networks (GANs) con-

#### CHAPTER 2. THEORETICAL BACKGROUND

tribute to generating realistic images, while Large Language models focus on producing coherent and contextually relevant human language.

In summary, this chapter provides a comprehensive theoretical background, setting the stage for the literature review. The insights gained here will be useful for understanding the application and implications of these technologies in the creation of MCQs using the Generative Pre-trained Transformer (GPT).

#### 2.5. CONCLUSION

## **3** | **State of the art**

# 3.1 Advancements and Impact of the GPT Models

The journey of GPT began with GPT-1, which started the practical application of the transformer architecture for language modeling. Through unsupervised learning on large text data, GPT-1 was able to effectively capture linguistic features and structures, proving the model's capabilities necessary for language processing [57].

With GPT-2, the model size was increased to 1.5 billion parameters, up from 117 million in GPT-1, which demonstrated the advantages of scaling up parameters. This increase in parameters improved the model's ability to generate coherent and contextually appropriate text. The increase highlighted the relationship between the number of parameters and the model's proficiency in handling diverse prompts and generating high-quality text [58].

GPT-3 leaped to 175 billion parameters and set new benchmarks in natural language processing, achieving a state-of-the-art performance in tasks such as text generation, translation, summarization, and question answering. This increase improved the model's ability to generalize across tasks and also enhanced its proficiency to understand and generate nuanced language [15].

While the number of parameters is well known from GPT-1 to the GPT-3 models, the number of parameters for GPT-4 is based on rumors, as the commercialization of OpenAI and fierce competition with companies such as Google, Meta, and Aleph Alpha, have led to OpenAI becoming less transparent about specific details. GPT-4 is rumored to have 1.76 trillion parameters [65]. With this massive jump, recent reports have shown that GPT-4 can tackle complex and nuanced language tasks with greater accuracy and understanding of context [9, 59]. The architecture of GPT-4 is based on the Mixture of Experts model, which combines multiple smaller models to work collaboratively to improve performance and specialization across various tasks [59]. Additionally, GPT-4 includes a vision encoder, allowing it to process images, thereby broadening its potential applications within AI [50].

GPT Model	Number of Parameters	
GPT-1	117 million [57]	
GPT-2	1.5 billion [58]	
GPT-3	175 billion [15]	
GPT-4	1.76 Trillion* [65]	

Table 3.1: Parameter counts for GPT-1 to GPT-4

\* Based on leaks and rumors by George Hotz, but not officially confirmed by OpenAI

### 3.1.1 GPT-3, GPT-4, GPT-40 the newer the better?

There are not any studies that directly compare the MCQ generation capabilities of the different GPT models, but it is clear that the newer GPT-4 model performs better than its predecessors. Studies on Machine Translation [35], Grammatical Error Correction [21] and various other language processing tasks [60], demonstrated that there was a significant performance boost with GPT-4. The most recently released model, GPT-4o (May 2024), has been shown to perform better than GPT-4 in most task-specific benchmarks, especially in multilingual tasks and tasks that require complex reasoning [3, 2]. Further research on the speed and results in terms of multiple-choice questions will be done in the Implementation Chapter 4 later.



Figure 3.1: GPT4o vs GPT4 in multilingual tasks. **Taken from** [3]

### 3.1.2 Educational Potential of GPT-40

OpenAI showcased GPT-4o's potential as an educational tool through a series of videos, demonstrating its multimodal capabilities. The demonstrated model effectively processes and generates diverse media inputs and outputs, excelling in tasks such as translating text from images, providing context for audio-visual content, and engaging in real-time conversations with rapid response times (as fast as 232 milliseconds) [51]. In educational settings, GPT-4o can be deployed to interactively engage with students by offering detailed explanations and facilitating interactive learning experiences. It is capable of real-time problem-solving and providing personalized feedback, adapting instructional content to suit each learner's comprehension level. These features enable GPT-4o to support a variety of learning needs, including enabling self-directed learning and assisting educators in delivering customized educational experiences [52, 22].

Despite its potential, many features demonstrated during GPT-4o's introduction are not yet publicly available. Nonetheless, the model's ability to tailor interactions based on individual student needs and its proficiency in handling complex educational tasks position it as an important asset in modern education.

## 3.2 Prompt Engineering Techniques and Patterns

Prompt engineering could be seen as a form of programming that customizes a language model's output. It is a method for instructing and guiding language models to produce desired results. There is a direct relationship between the quality of a language model output, such as that of GPT, and the quality of the instructing prompt. Various prompt engineering techniques can be applied for specific uses depending on the application domain, ranging from question-answering and commonsense reasoning to code generation and beyond [62].

Key challenges of LLMs that prompt engineering aims to address include mitigating hallucination, where the LLM generates information not based on the provided input or is factually incorrect, enhancing user interaction, ensuring consistency and coherence, accurately understanding user intent, and improving reasoning and logic.

Prompt patterns, introduced by White et al. (2023), resemble software design patterns and provide practical, reusable solutions to common problems encountered when interacting with LLMs. These patterns enhance the effectiveness and efficiency of LLMs across diverse tasks, particularly in the context of software development, by offering structured approaches to manage input semantics, output customization, error identification, and more. While the research on Prompt Patterns by White et al. (2023) is focused on software development, many of these patterns are versatile and can be adapted to other domains, such as improving the instructions for creating MCQs [72].

#### 3.2.1 Prompt Patterns

Prompt patterns are reusable solutions to common problems encountered when interacting with LLMs. These patterns are documented in a structured format similar to software design patterns, which enhances their reuse and transferability across different contexts and domains [72].

The paper by White et al. (2023) [72]. introduces several categories of prompt patterns, each addressing different aspects of prompt engineering (see Table 3.2). Each prompt pattern is documented to provide a comprehensive guide for implementation. Each fits within a category and has its own intent and context - indicating the problem it solves and the goals that it achieves. There is also a clear motivation behind each pattern, highlighting the rationale behind the problem. Additionally, each prompt pattern is analyzed for its pros and cons when applied in practice.

Category	Description and Patterns	
Input Semantics	Influences the understanding and handling of input of an LLM.	
Output Customization	Modifies the format and structure of the output.	
Error Identification	Aims to identify and resolve errors in the LLM's output.	
Prompt Improvement	Enhances the quality of the input and output.	
Interaction	Focuses on the interaction between the user and the LLM.	
Context Control	Allows for the specification or removal of context for a conversation with an LLM.	

Table 3.2: Categories of Prompt Patterns in Prompt Engineering. Taken from [72]

#### 3.2.2 Relevant Prompt Patterns for the creation of MCQs

I will lay out the prompt patterns by White et al (2023), that are highly relevant and potentially useful for the generation of MCQs.

#### Prompt Pattern Category: Output Customization

The **Persona Pattern** is a method where a specific role is assigned to an LLM. This helps the LLM to generate appropriate types of output and prioritize details and qualities that are relevant within the context of the role. The motivation behind this pattern is that often users lack the knowledge of what specific outputs or details an LLM should focus on for a given task. It is easier for the user to identify the job, expertise, or role required for the task. This is where the Persona Pattern allows users to express their

#### CHAPTER 3. STATE OF THE ART

needs in terms of a role without needing to specify exact details. To generate MCQs, the Persona Pattern can be implemented like this:

#### User message:

Act as an assessment expert in the field of study of Biology. Generate MCQs that are challenging and test a deep understanding of the subject.

Another Output Customization pattern is the **Template Pattern**. It ensures that an LLM's output follows a given structure, which is useful when the generated output needs to fit a certain data structure (e.g. JSON, CSV, ...). The motivation behind this pattern is applications that require the input they are given (in this case the output from a LLM) to be in a specific format. Without specifically instructing the LLM, it will not know which format is required. When generating multiple MCQs throughout multiple prompts, the format needs to stay the same. This is where the Template Pattern comes in handy:

#### User message:

Generate an MCQ about Sorting Algorithms in the context of Programming that follows this JSON format:

```
{
    "question": "Sample Question Text",
    "options": {
        "A": "Option A",
        "B": "Option B",
        "C": "Option C",
        "D": "Option D"
    },
    "correct_answer": "A"
}
```

#### Prompt Pattern Category: Prompt Improvement

The Fact Check List Pattern will be an important tool to use when creating MCQs that are based solely on the knowledge of the LLM. LLMs, including GPT, usually produce convincing but not always factually correct text. The text appears persuasive - therefore a user might not verify its accuracy. With this pattern, the LLM generates a list of facts in the output, allowing for subsequent verification by the user. A lot of the inaccuracies stem from the fact that LLMs such as GPT draw a lot of their training data from the internet. When asking GPT to create MCQs based on a Subject and Field of Study without the content that the MCQs should specifically target, this would be extremely useful for educators to be able to go through a list of facts that the LLM went through, to easily remove the ones that were based on inaccuracies.

#### 3.2. PROMPT ENGINEERING TECHNIQUES AND PATTERNS

#### User message:

When generating an MCQ, include a list of the facts that you used to create that MCQ.

Create a multiple-choice question about Geography in the European Union.

## GPT Response

#### **Response:**

Question: Which River is the longest in the European Union?

A: Rhine

B: Volga

C: Danube

D: Dnieper

Correct Answer: C

Used Facts:

- 1. The Danube River is 2860km long
- 2. The Danube River is the longest in the European Union.
- 3. The second longest river is the Ural at 2482km

#### Prompt Pattern Category: Context Control

The **Context Manager Pattern** is a method that specifies or omits context when conversing with an LLM. This pattern is designed to narrow down on specific topics or exclude irrelevant information, to provide greater control over what the LLM will use as context when generating output. Commonly, LLMs do not interpret context correctly, leading to unwanted outputs. In the context of creating MCQs with GPT, this pattern could help to restrict the creation of MCQs based on a given context.

#### User message:

Answer the following MCQ, but you may only use the following text as your source of knowledge: <text>

### 3.2.3 Evaluating Prompt Patterns

In evaluating these prompt patterns, we observe that each offers distinct advantages and challenges. The Persona pattern enhances relevancy by tailoring outputs to specific roles but risks oversimplifying complex roles. The Template pattern ensures consistent output formats ideal for integration with other systems, yet may struggle with complex information. The Fact Check List pattern boosts reliability by listing verifiable facts, though it demands additional verification work from users. Lastly, the context manager pattern allows precise control over the context that the LLM considers, but this requires the instructions to be clear and specific, or it may not be effective.

#### CHAPTER 3. STATE OF THE ART

Prompt Pattern	Pros	Cons
Persona	<ul> <li>Enhanced relevancy and context in the LLM's output</li> <li>Tailors responses based on specific roles or expertise</li> <li>Easier for users to get targeted and useful results</li> </ul>	<ul> <li>Potential to oversimplify complex roles</li> <li>May misunderstand nuances of the assigned persona</li> <li>Could lead to less accurate or overly generalized outputs</li> </ul>
Template	<ul> <li>Consistent and reliable output format</li> <li>Crucial for applications requiring standardized input structures</li> <li>Simplifies integration with other systems</li> <li>Ensures data generated by LLM is immediately usable</li> </ul>	<ul> <li>Potential rigidity in fitting complex or nuanced information</li> <li>Users might need to invest time in designing and refining templates</li> </ul>
Fact Check List	<ul> <li>Provides a clear list of verifiable facts</li> <li>Helps educators quickly identify and correct inaccuracies</li> </ul>	<ul> <li>Additional workload on the user to manually verify each fact</li> <li>Effectiveness depends on the user's ability to verify facts</li> <li>May require domain-specific knowledge or access to reliable sources</li> </ul>
Context Manager	<ul> <li>Enhances focus and relevance in the conversation</li> <li>Allows precise control over the context the LLM considers</li> </ul>	<ul> <li>Risk of unintentionally removing useful context</li> <li>Requires clear and specific instructions to be effective</li> </ul>

Table 3.3: Pros and Cons of Prompt Patterns

## 3.2.4 Prompting Techniques

In Sahoo et al. (2024) [62], a systematic survey of Prompt Engineering Techniques and their applications is conducted. I will investigate the ones that are highly relevant and potentially useful for the creation of MCQs. These prompt engineering techniques are versatile and can be applied for specific uses depending on the application domain.

Zero-shot prompting is a technique [58] where a model is given a task without

#### 3.2. PROMPT ENGINEERING TECHNIQUES AND PATTERNS

any prior examples or additional training data related to that task. This means that this method uses the pre-trained knowledge of the model to perform new tasks directly based on the provided prompt [62]. Zero-shot prompting removes the need for extensive retraining or fine-tuning, allowing models to adapt to new tasks swiftly. Even simple prompts with zero-shot prompting can yield impressive results, which is actually the most common way users interact with conversational language models. A prompt with zeroshot prompting may be just a simple question, without any specific examples or additional context to guide the model's response.

User message (Zero-Shot prompting): What is photosynthesis?

#### GPT Response

**Response:** Photosynthesis is the process by which green plants and some other organisms use sunlight to synthesize foods with the help of chlorophyll from carbon dioxide and water. It produces oxygen as a byproduct.

**Few-shot prompting** is a technique [15] where a model is given a few examples of the desired output. The idea behind this is to give the LLM a sense of understanding of a given task. This is contrary to zero-shot prompting where no examples are given. By just giving the LLM a few high-quality examples, the model's performance can improve drastically. In benchmarks and model comparisons, you might encounter terms like "5-shot prompting," which just indicates the number of output examples given to the model, as the number of output examples influences a language model's performance [44].

A complicating aspect of few-shot prompting is that it is sensitive to the order in which examples are presented. Different permutations of the input-output examples can lead to state-of-the-art performance, while others result in near-random performance. Lu et al. (2022) have also discovered that this sensitivity is persistent across different model sizes and tasks, which may indicate that this is a fundamental aspect of few-shot learning [44].
#### CHAPTER 3. STATE OF THE ART

# User message (3-shot prompting):

Translate the following English sentences into French:

English: "Hello, how are you?" French: "Bonjour, comment ça va?"

English: "I would like a coffee, please." French: "Je voudrais un café, s'il vous plaît."

English: 'What time is the meeting?" French: "À quelle heure est la réunion?"

English: "Can you help me with this?" French: "[Your response]"

#### GPT Response

#### Response:

English: "Can you help me with this?" French: "Pouvez-vous m'aider avec ça?"

**Chain-of-Thought (CoT)** prompting is a technique [71] that enhances the performance of language models by guiding them through a sequence of reasoning steps, leading to more structured and thoughtful responses. This method mimics human problem-solving processes by encouraging step-by-step thinking, which is extremely helpful for complex tasks requiring multi-step reasoning [62]. A series of experiments demonstrated the effectiveness of CoT prompting in guiding language models through logical reasoning chains, achieving impressive results in math and commonsense reasoning tasks [71]. CoT prompting enabled PaLM 540b, a model developed by Google [18], to achieve an accuracy of 90.2% [62] in these areas. CoT prompting significantly improves a language model's ability to generate detailed and logical answers [5].

**Zero-Shot CoT**: Prompt the LLM to help in the generation of the reasoning steps iteratively

**Manual CoT**: In this variation, the user provides reasoning examples as a sort of template to guide the model toward an output. The disadvantage of this approach is the reliance on good examples, as the crafting of such examples can be labor-intensive.

In the creation of multiple-choice questions (MCQs), CoT prompting can be particularly useful as it ensures that each question is developed through a thorough reasoning process, resulting in well-constructed questions.

#### 3.2. PROMPT ENGINEERING TECHNIQUES AND PATTERNS

#### User message (No reasoning steps given):

Question 1: Jason is 5 times older than his brother. Four years later, Jason now only is twice as old as his brother. How old are they now?

Answer: Jason is now 10 years and 8 months old, and his brother is 5 years and 4 months old.

Question 2: Felix is double the age of his sister. Five years ago, Felix was triple the age of his sister. How old are they now?

John is twice as old as Mary. Five years ago, John was three times as old as Mary. How old are they now?

Please solve Question 2.

#### GPT Response

#### Response (Answer is incorrect):

John is 30 years old and Mary is 15 years old.

When the model is given the problem without any examples, it struggles to understand the steps needed to solve the problem accurately, leading to an incorrect response. Note that this is just an example for demonstration, modern LLMs like GPT-4 can easily solve simple problems like these without the need of CoT prompting.

#### CHAPTER 3. STATE OF THE ART

User message (Chain-of-Thought Prompting): Solve the following algebraic word problems.

Question 1: Jason is five times as old as his brother. In four years, Jason will be twice as old as his brother. How old are they now? Answer: Let J be Jason's age and B be his brother's age. From the problem, we have two equations: 1) J = 5B2) J + 4 = 2(B + 4)Substitute the first equation into the second: 5B + 4 = 2(B + 4)5B + 4 = 2B + 83B = 4B = 4/3J = 5B = 20/3

Jason is 20/3 years old (approximately 6 years and 8 months), and his brother is 4/3 years old (approximately 1 year and 4 months).

Question 2: Felix is double the age of his sister. Five years ago, Felix was triple the age of his sister. How old are they now?

Please solve Question 2.

#### **GPT** Response

#### Response

Solution: Let F be Felix's age and S be his sister's age. From the problem, we have two equations: 1) F = 2S2) F - 5 = 3(S - 5)2S - 5 = 3(S - 5) 2S - 5 = 3S - 15 10 = S J = 2S = 20 Felix is 20 years old and his sister is 10 years old.

Here, the reasoning steps and a detailed solution are given to the model for a similar problem, directing the model to use the same logical process, which leads to the correct solution.

**Retrieval-augmented generation (RAG)** is an advanced technique used in natural language processing that enhances the capabilities of LLMs by integrating external knowledge sources into the generation process. The core idea behind RAG is to further enrich the inherent knowledge of LLMs with relevant information retrieved from vast external databases or knowledge repositories [26]. This is accomplished through a multi-step pro-

#### 3.2. PROMPT ENGINEERING TECHNIQUES AND PATTERNS

cess where a query is first transformed into a vector representation and used to search a pre-constructed index of document vectors for the most semantically similar content. The retrieved information is then combined with the original query to form a comprehensive prompt that the language model uses to generate a response. This approach helps address common LLM challenges such as hallucination and outdated knowledge by basing the responses on up-to-date and contextually relevant information. RAG has proven particularly effective in knowledge-intensive tasks like question answering and domain-specific content generation, where access to current and precise data significantly enhances the quality and accuracy of generated outputs. However, implementing RAG can be complex, as it involves integrating various machine learning components and requires expertise in both retrieval and generation technologies. By combining the language model's generative capabilities with retrieval-based precision, RAG systems offer a robust framework for producing highly relevant and informative content [26].



Figure 3.2: Three paradigms of Retrieval-Augmented Generation. Taken from Gao et al. (2020), NeurIPS,  $\bigcirc$  2020 by the authors [26].

# 3.2.5 Conclusion

This exploration into various prompt engineering patterns and techniques shows us that there is a niche with nuanced strategies that are significant for enhancing the effectiveness of language models. The quality of a language model's output is undoubtedly directly linked to the quality of the input prompts.

Prompt engineering addresses challenges such as hallucination and user interaction and also improves the overall accuracy and reliability of LLM outputs. Prompt patterns provide structured and reusable approaches to tackle common problems encountered when conversing with any conversational LLM. These patterns, such as the Persona, Template, Question Refinement, Fact Check List, and Context Manager, each come with distinct

#### CHAPTER 3. STATE OF THE ART

benefits but also their limitations, so it is important to carefully consider which ones to use and combine based on application context. These patterns have the potential to improve the generation of MCQs to ensure consistency, relevance, and factual accuracy. The use of the Persona and Template pattern allows for role-specific content generation (e.g. an assessment expert) and standardized, consistent output formats.

Prompting techniques like zero-shot, few-shot, and chain-of-thought (CoT) demonstrate how language models can be guided by instructing them with example-driven input to achieve desired outcomes. Retrieval-augmented generation (RAG) further extends the potential of language models by including external knowledge sources.

A lot of research has gone into prompt engineering in the last 3 years and has been ongoing, especially as conversational LLMs like ChatGPT, LLama, and Claude gain popularity. With the help of prompt engineering, these general-purpose tools are turned into more domain-specific tools that are capable of addressing complex tasks.

# 3.3 What is a good multiple-choice-question?

Multiple-choice questions (MCQs) are a staple of educational assessments, providing an efficient means to evaluate student knowledge across a wide range of subjects. The quality of MCQs determines the reliability and validity of these assessments, making it important to understand what a well-constructed MCQ even is. This section explores the essential characteristics of effective MCQs, focusing on stem quality and the design of distractors, using different sources and then aggregated by Brame [14].

A well-designed stem presents a clear, self-contained question that aligns with learning objectives, while effective distractors are plausible, non-guessable options that reflect common misconceptions. By adhering to these principles, educators can create MCQs that not only accurately assess student understanding but also enhance the overall learning experience. The following subsections delve into the specifics of these characteristics.

# 3.3.1 Characteristics of a Good MCQ

#### Stem Quality

The stem is the part of the MCQ that presents the problem or question being asked to the student. The quality of the stem is crucial as it sets the context for the question and guides the student in understanding what is being asked.

A well-constructed stem in an assessment should be understandable and meaningful on its own, without requiring additional context. This ensures that the stem clearly presents the problem and accurately assesses the student's knowledge of the learning objective, rather than their ability to interpret vague descriptions (see Fig 3.5).

To maintain the reliability and validity of test scores, the stem should avoid including

<b>Options:</b> A. The Hamburger originated in Germany
B. The Hamburger originated in Hungary
C. The Hamburger originated in France

Figure 3.3: Meaningful stem

Figure 3.4: Not Meaningful stem

Figure 3.5: Examples of Meaningful and Not Meaningful Stems

irrelevant information that can distract or confuse students. Additionally, stems should be phrased negatively only when necessary to assess significant learning outcomes, as negatively phrased items can be difficult for students to understand. If negative phrasing is used, the negative part should be highlighted through italics or capitalization to ensure clarity.

**Stem:** In the late 1800s, many scientists, including Albert Einstein and Marie Curie, were exploring the nature of energy and matter. Einstein is famous for developing the theory of relativity, which revolutionized physics. Around the same time, radioactivity was discovered, leading to significant advancements in nuclear physics and chemistry. In the context of nuclear reactions, which of the following particles is emitted during beta decay?

#### **Options:**

A. Alpha particleB. Beta particleC. Neutron

Figure 3.6: The Stem contains irrelevant and potentially distracting/confusing information.

#### CHAPTER 3. STATE OF THE ART

The format of the stem should ideally be a complete question rather than a partial sentence, as complete questions enable students to concentrate on providing an answer without the cognitive load of remembering an incomplete sentence while considering each alternative.

Finally, the stem should be clear, straightforward, and written in simple, direct language to facilitate understanding and focus on the assessment objectives.

#### **Effective Distractors**

The distractors are the choices provided in an MCQ from which the student must select the correct answer. The effectiveness of these distractors is critical to the functionality of the MCQ.

When designing multiple-choice questions, it's essential to ensure that distractors are both plausible and relevant to the question. They should reflect common misconceptions or errors to effectively differentiate between students who understand the material and those who do not.

The correct answer should require genuine knowledge and understanding of the subject matter and not be easily guessed. Each question should have one clear, unambiguous correct answer to avoid guesswork and ensure fairness in assessment. Multiple plausible answers could lead students to choose different but justifiable responses.

Stem: What contributes to climate change?

#### **Options:**

- A. The release of greenhouse gases
- B. Increased solar radiation
- C. The popularity of ice cream flavors
- D. The extinction of the Dinosaurs

Figure 3.7: Multiple things are wrong in this MCQ. Distractors C and D are not plausible at all, meaning they are easily eliminated by test takers.

Option A is the more widely recognized contributor, but Option B is technically correct as well. There should only be one clear single correct answer to avoid ambiguity. One could even argue that the extinction of the Dinosaurs contributed to long-term climate changes.

Distractors should be clear and concise to avoid testing reading ability rather than the learning objective. The distractors must be homogenous in content, style, or structure as varied content can inadvertently hint at the correct answer (see Fig 3.10).

Stem: Who wrote the novel "The	Stem: Who wrote the novel "The		
Casual Vacancy" in 2012?	Casual Vacancy" in 2012?		
Options:	Options:		
A. Albert Einstein	A. J.R.R. Tolkien		
B. J.K. Rowling	B. J.K. Rowling		
C. George Washington	C. George R.R. Martin		

Figure 3.8: Homogeneity is lacking, as not all options are authors or writers, they mix unrelated professions and historical figures, which will make it easy to identify J.K. Rowling as the correct answer without needing specific knowledge of literature. Figure 3.9: In this example, the test-taker must rely on their knowledge of the authors rather than a process of elimination based on irrelevant options.

Figure 3.10: Non-homogeneity vs Homogeneity

Additionally, distractors should be mutually exclusive, not overlapping with the correct answer or each other, to prevent creating "trick" items that can undermine trust in the testing process (see Fig 3.11).

Stem: Which of these is used as a pain reliever?

#### **Options:**

- A. Ibuprofen
- B. Acetaminophen
- C. NSAIDs
- D. Aspirin

Figure 3.11: The distractors overlap. Both Ibuprofen and Aspirin are used as a pain reliever and NSAIDs contain both Ibuprofen and Aspirin, making this choice overlap with A) and D)

Distractors should be consistent and match the stem in grammar, form, length, and language style, such that they do not provide clues about which one is correct. Popular distractors such as "all of the above" and "none of the above" should be avoided since they can allow students to use partial knowledge to guess correctly.

The order in which Distractors are present should be in a logical order, alphabetical (a,b,c,d) or numerical (1,2,3,4), to avoid bias toward specific positions.

Finally, the number of distractors can vary as long as they are relevant and plausible. Plausible distractors serve as effective distractors for students who haven't mastered the subject, while those who have mastered it will easily disregard them.

## 3.3.2 Relevance and General Guidelines

To ensure the educational effectiveness of MCQs, several general guidelines should be followed. First, questions should be relevant to the learning objectives and assess knowledge or skills suitable to these objectives, focusing on key concepts rather than trivial details. The grammar and format of the questions should be consistent, ensuring that all options are grammatically correct and use the same tense to maintain clarity and coherence.

Evaluation and feedback are also critical components. MCQs should allow for meaningful feedback by explaining why the correct answer is correct and why the other options are incorrect. This feedback helps reinforce learning and corrects misconceptions. Complex multiple-choice items, which include options consisting of different combinations, should be avoided as they can be confusing for test-takers.

Furthermore, when multiple MCQs are used, each question should be independent in content to prevent test-takers from using information from one question to answer another, thereby preserving the validity of the test. By adhering to these guidelines, MCQs can effectively assess and reinforce learning objectives.

# 3.3.3 Determining the Validity of MCQs

In the context of generating MCQs through GPT, there is also a need to validate the quality of the MCQs and whether they adhere to the established principles. In a study by Moore et al. (2023), 200 student-generated MCQs in four different subject areas were assessed by two experts who had content-area expertise across all four domains, more than adequate experience in creating MCQs and also were trained in writing high-quality assessments [46]. They evaluated each MCQ based on 19 Item-Writing Flaws (IWFs), including ambiguous information, implausible distractors, "none/all of the above" distractors, repeating words in stems and correct response, unfocused stem, grammatical cues, and more.

The study then compared the performance of a rule-based approach and GPT-4 in evaluating MCQs against the experts' assessment. The rule-based method, which applied specific logic to check each IWF, matched 90.87% of the classifications made by the experts, and the GPT-4 method, which used a prompt for each IWF, matched 78.89%. The rule-based method had an exact match ratio of 15% and GPT-4 12% (meaning that the results were identical to the experts)[46].

Both methods tended to identify more flaws than human experts, with GPT-4 identifying the most. The rule-based method was found to be more effective. Despite the lower match rate with human experts, GPT-4 showed potential in explaining why criteria were met or violated, indicating that it may be extremely useful in settings where feedback is important [46].

The study highlights the potential for these automated methods in improving MCQ

#### 3.4. AI-GENERATED MULTIPLE-CHOICE QUESTIONS

quality by first filtering out questions with multiple flaws before they are given to humans for review, thereby saving time and ensuring better educational outcomes [46].

In an updated study by Moore et al. (2024), the Scalable Automatic Question Usability Evaluation Toolkit (SAQUET) was introduced as an innovative solution to evaluate MCQs across diverse domains, including Chemistry, Statistics, Computer Science, Humanities, and Healthcare. Once again, it uses the 19 IWFs to evaluate the structural and pedagogical aspects of MCQs [45].

Compared to its predecessor (the rule-based approach), SAQUET demonstrated an accuracy rate of more than 94% in detecting Item-Writing Flaws that the experts identified. SAQUET was tested on a dataset of 271 MCQs and achieved a 38% exact match ratio with expert evaluations, compared to the 15% matching exact matching rate from the rule-based approach [45]. SAQUET clearly surpassed the performance of the previous rule-based methods by improving the exact match ratio with experts and reducing misclassification rates.

SAQUET by Moore et al. (2024) provides a robust framework for ensuring high-quality MCQs across various domains and has immense potential in revolutionizing the quality assessment of MCQs [45].

# 3.4 AI-Generated Multiple-Choice Questions

Creating high-quality MCQs is traditionally labor-intensive and requires significant expertise not only in the subject matter but also in aligning with learning objectives and following best practices for constructing effective MCQs. Recent advancements in generative AI, particularly in LLMs like GPT, have demonstrated the potential to automate this process and produce promising results in specialized fields such as programming education and medical examinations [37, 23, 17].

The underlying technology of generative AI models like GPT involves advanced NLP algorithms (2.3) and machine learning techniques (2.2). These models are trained on diverse datasets, which enables them to understand natural language instructions and generate human-like text in return. With the transformer architecture (2.3.1), GPT uses the attention mechanism of the transformer, to understand complex instructions. GPT is widely seen as the leading LLM, therefore the majority of the literature surrounding the creation of MCQs with generative AI focuses on using the GPT models. All of the following sections about AI-generated MCQs are based on literature focusing on the application of GPT models.

## 3.4.1 Effectiveness and Efficiency

GPT has become an extremely valuable tool, especially in medical education, as MCQs are used frequently to assess the knowledge and skills of medical students. Therefore it is natural that the medical education field has published many studies that assess the

#### CHAPTER 3. STATE OF THE ART

quality and validity of MCQs created by GPT. A literature review by Kıyak and Emekli [40] explored the results and findings of 23 relevant studies in the medical field, but it is important to note that some of these studies used the GPT-3.5 model version, as they were conducted before the improved GPT-4 version. Other mentioned studies were conducted in different fields such as Python Programming in Higher Education [23] and more general Programming Classes [64].

#### **Positive Findings**

There are significant advantages when using GPT to generate MCQs in medical education. The biggest benefit is reducing the time spent by educators on crafting high-quality MCQs. In one study [74], the creation time of base-cased MCQs by GPT compared to human creators were reduced by 75% - 83.33%, and in another study, humans required 211 minutes to write 50 MCQs, whereas GPT did so in just over 20 minutes [17].

As GPT is trained on an extremely large amounts of text data during the pre-training phase, including internet data, books, and articles (2.4.3), the studies have shown that it is able to leverage this knowledge to be applied in various fields in medical fields and subjects such as physiology [32, 66], dermatology [7], anatomy [32, 66], internal medicine [17], surgery [17], diabetes [24] and many more.

Some of the studies have achieved similar levels of discrimination and difficulty compared to human-written MCQs, but only when appropriate prompts were used, showing that these GPT-generated MCQs were effective in distinguishing between high-performing and low-performing students [74, 39]. Discrimination is an index that ranges from -1 to 1 in the context of educational assessments and refers to the ability of a test item (in this case MCQs) to distinguish between high-performing and low-performing students based on their overall test scores. The higher the discrimination, the better the reliability and validity of an assessment. Scores above 0.2 are considered acceptable, and scores above 0.3 are considered ideal [40]. In the 23 relevant studies reviewed by Kıyak and Emekli [40], the results were extremely mixed. Studies have reported ideal levels above 0.3 [39], some had acceptable levels of discrimination [74] but there were also discrimination scores below 0.2 [41], but these differences can largely be attributed to the difference in qualities of the prompts that were used.

In a study by Doughty et al. (2024) that focused on Python Programming, GPT-4 was used to produce 651 MCQs for Python programming courses and then evaluated alongside 449 human-crafted MCQs. It was found that GPT-4 was able to generate MCQs that are clear, concise, and well-aligned with specified learning objectives (LOs), and also of comparable quality in terms of language clarity and plausibility of distractors (see Fig 3.12) [23]. It is important to note that the instructors who created the MCQs were not aware of the requirements and that if aware, they could have created MCQs that better satisfy the requirements. Additionally, the study itself paired the human MCQs with the learning objectives, which could be part of the reason for the observed MCQ-LO misalignment for Humans [23].

#### 3.4. AI-GENERATED MULTIPLE-CHOICE QUESTIONS



Figure 3.12: Comparing the generated MCQs with GPT with human-crafted ones. Taken from [23].  $\bigcirc$  2024 by Jacob Doughty et al., licensed under CC BY 4.0.

#### Concerns

While GPT has shown potential in generating MCQs, several concerns need addressing to ensure their reliability and effectiveness in educational settings.

**Hallucination** is a significant concern in LLMs because these models can generate content that appears plausible but lacks a factual basis, posing risks in applications where accuracy is critical. A study suggests that hallucinations in LLMs arise from the models' tendency to produce unverifiable or fabricated information, especially when dealing with ambiguous or open-ended prompts [34]. Similarly, another study highlights that hallucinations can undermine trust in generative AI systems, particularly in fields such as healthcare and the legal sector, where the reliability of information is extremely important [8].

In addition to hallucination issues, **Domain-Specific Performance Variability** which describes the varying performance of GPT across different domains, further complicates its reliability. For instance, it performed worse in Chemistry and Biochemistry compared to Statistics and CollabU, possibly due to the insufficient coverage of specific jargon or nuanced language in its training data [46].

There is also the potential for **content inaccuracies** when using GPT for generating MCQs. Experts reviewed the MCQs and the results were vastly different, ranging from 1% [37] to 60% [7] of inaccuracy. In a study, experts even concluded that all MCQs were acceptable [39]. Inaccuracies lead to incorrect or misleading questions, and unless they are checked retroactively by experts, the result is the spread of incorrect information. A study found that 4.9% of automatically generated MCQs had multiple correct answer choices, compared to only 1.1% for human-crafted MCQs [23]. This issue is significant as it requires human intervention to fix. Therefore AI-generated MCQs should optimally be reviewed by experts and validated to ensure their factual correctness, which could nullify one of the

#### CHAPTER 3. STATE OF THE ART

aforementioned advantages, namely the time that is saved by using GPT to create MCQs. Klang et al. (2023) [37] generated 210 MCQs for medical examination, but around 15% of the generated MCQs required revisions due to inaccuracies or methodological flaws. GPT-generated MCQs were also more likely to have distractors that were obviously wrong or gave away the correct answer, with 4.0% of the generated questions showing this flaw compared to 0.9% of human-crafted questions. This issue was most pronounced in Fill-in-the-Blank and Scenario-Based MCQ types [23].

The quality of each generated MCQ is highly variable, as the output of GPT is highly dependent on the design of the prompt, the given instructions, and the requirements. **Prompt Design Dependency** is important because poorly designed prompts lead to more questions that are unsuitable and/or fail to meet educational objectives. In some cases, the correct answer was revealed in the question stem itself, which can render the question ineffective as an assessment tool [23]. To effectively use GPT, educators must also invest time in developing precise and detailed prompts, by using prompting techniques (3.2.4) and prompt patterns (3.2.1). Without careful and knowledgeable design of the prompts, the resulting MCQs will most likely not fulfill their intended purpose, which would weaken the utility of GPT in creating MCQs.

#### Re-emphasizing the importance of prompt engineering

Prompt Engineering (3.2) plays an extremely important role in effectively using GPT and drastically reducing the aforementioned disadvantages. The Quality and relevance of the AI-generated questions are heavily determined by the prompts that are given to the model. Detailed instructions, specific references, and prompt engineering techniques are needed to guide GPT in producing content that aligns closely with the intended educational outcomes. These include prompts that reference specific exam styles using a form of the few-shot prompting technique (3.2.4) as it offers implicit examples of the question format, style, and level of difficulty expected [11, 12, 7]. The Persona pattern (3.2.2) was also adapted in some of the studies [28], such as "You are developing a question bank for medical exams ..." [38, 39]. Some studies were also submitting text as a reference [7, 17] for GPT to use and generate responses from, which is a form of the Retrieval-Augment Generation Prompting technique (3.2.4) and the Context Manager Pattern (3.2.2).

The correlation between the quality of the prompt and the achieved outcomes is apparent. In the study by Ngo et al. (2024), an extremely simple prompt was used "Write 4 multiple choice questions with 4 answers and explanations for the incorrect and correct answers". These MCQs were then evaluated by 4 expert reviewers and according to them 32% of the MCQs required minor revisions, 43% of the MCQs required major revisions and 25% were not usable at all [49].

On the other hand, more structured prompts lead to better quality and relevance of the output. In Kiyak et al. (2023) a more structured prompt is presented that implements the Persona Pattern, Template Pattern, and Chain-of-Thought prompting [39]. Two of these generated MCQs were actually included in a pharmacotherapy exam and ideal levels

of discrimination were found (0.41 and 0.39), meaning that these questions were effective in differentiating between students that perform at high and low levels [39].

# 3.4.2 Current Applications of AI in Education and Potential

Though a lot of research is still necessary and ongoing, real-life applications of Algenerated MCQs have already been implemented in large-scale online education platforms such as Coursera and Quizlet but also tools for educators to create MCQs such as WIRISquizzes and HyperQuiz [68, 29].

Coursera, for instance, has incorporated generative AI tools to auto-generate course materials, including readings, assignments, and MCQs, which significantly reduces the time and cost of producing high-quality content [19]. Additionally, Coursera has developed features like the Quick Grader to provide efficient, scalable feedback to students [20]. HyperQuiz is an online tool that helps teachers to generate MCQs automatically. It requires a text input or a YouTube video link (using the YouTube captions as the text input), to create MCQs [29].

Al-generated MCQs can also be adapted to students' needs, offering a personalized learning experience. In this context, adaptive assessments can automatically adjust their difficulty and content based on the student's performance, providing a customized educational experience to help identify areas of strength and weakness. For example, as noted in the study by Owan et al. (2023) [55], adaptive AI systems can dynamically modify the question pool based on real-time analysis of a student's responses, ensuring that the assessments remain challenging yet achievable, and thereby supporting effective learning progression [19].

Systems like ALEKS and Knewton use AI to offer adaptive assessments aligned with students' learning trajectories. Adiguzel et al. (2023) report that these systems provide immediate feedback and personalized learning paths, significantly improving student engagement and performance. ALEKS, for example, uses adaptive questioning to identify knowledge gaps and provide targeted instruction, enhancing learning outcomes in mathematics [1].

# 3.4.3 Conclusion

Using generative AI models such as GPT-40 to create MCQs is a big step forward in educational technology. It has been demonstrated that these LLMs can generate high-quality, contextually relevant questions across multiple domains, including medical and programming education. With this new application, it is possible to massively alleviate the workload for educators while at the same time enhancing the scalability of assessment creation. With a combination of careful prompt engineering and expert validation, AI-generated MCQs can match the levels of quality and discrimination comparable to those

#### CHAPTER 3. STATE OF THE ART

crafted by human experts.

Further research and development are required, as challenges such as hallucinations, domain-specific performance variability, and content inaccuracies are still big factors. It is clear that prompt engineering is extremely important as it directly influences the relevance and reliability of the generated MCQs. But the outlook is positive, as LLMs continue to evolve (as of August 2024, models such as LLama 3.1b, Claude 3.5 Sonnet, and Gemini 1.5 have emerged after GPT-40 and are said to outperform GPT-40 in some categories [33].), we are guaranteed to see more applications of generative AI in educational settings. NLP capabilities will improve and hallucination will most likely become less prevalent, which will inevitably also increase the quality of MCQs.

The next chapter will delve into the Implementation Chapter, exploring the practical application of GPT models in generating MCQs. This chapter will cover the methodologies and techniques employed to fully use the full potential of GPT in creating high-quality MCQs.

#### 3.4. AI-GENERATED MULTIPLE-CHOICE QUESTIONS

# 4 Design and implementation

# 4.1 Decision

In developing a framework for generating MCQs using GPT models, I chose to utilize the standard GPT models without any custom fine-tuning. This decision aligns with my thesis's primary objective of exploring the capabilities of GPT models in generating MCQs.

# 4.1.1 Rationale

I decided to utilize the standard GPT models without any fine-tuning, due to some strategic considerations and constraints in the context of this thesis. The primary aim of this research is to explore the capabilities of GPT models in generating MCQs. Therefore I do not fine-tune the model due to practical limitations and the nature of this thesis.

Fine-tuning a model requires a large, high-quality dataset that is closely aligned with the desired output. In the context of this thesis, this would mean a large amount of expertly crafted MCQs, accompanied by the necessary context and subject matter. The development of such a dataset would be extremely time-consuming and resource-intensive, which was not feasible given the scope of this thesis.

Additionally, substantial computational resources are needed to fine-tune GPT, including high-performance GPUs and storage for data processing. The process of fine-tuning also involves complex experimentation with hyperparameters and model architectures, demanding both time and expertise that are beyond the scope and timeline of this thesis.

# 4.2 GPT Models and Parameters

#### 4.2.1 GPT Parameters

**Temperature** (from 0-2) is an important setting that has a significant impact on the output of GPT. It is called **temperature sampling** and the value controls the amount of randomness that is allowed in the output, and adjustments will lead to different results. The lower the temperature, the more likely GPT will choose words with a higher probability of occurrence (as a reminder, GPT constantly predicts the next word with the highest likelihood to create a response, as described in the Theoretical Background 2.4.3), which is useful for generating consistent, relevant, and more accurate outputs, with less variability. On the other hand, choosing a higher temperature will lead to GPT choosing words that are less likely to occur, which could be extremely useful when trying to generate creative

#### 4.2. GPT MODELS AND PARAMETERS

ideas or to increase the variability in responses.



Figure 4.1: Illustration on how the value of the temperature skews the probability distribution of the next word. This is just an example and not based on real distribution numbers.

**Top\_P** (from 0-1) is a similar setting to the temperature. It is called **nucleus sampling**, in which the model only takes into account the tokens with the highest probability mass - determined by the value given to the top\_p parameter. Probability mass refers to the total probability assigned to a group of tokens, with top\_p controlling how many of the most likely tokens are considered until their combined probability reaches a specified threshold.

OpenAI generally recommends to use of either temperature sampling or nucleus sampling, but not both [54]. For the experiments here, I will use temperature sampling to easily control the randomness and creativity of the model's outputs, allowing for more diverse outputs if necessary.

**Maximum Length** determines the maximum length of the API's output. The maximum output length varies from model to model. The general guideline is that one token is roughly equivalent to 4 characters or 0.75 words for English text. With this, one can estimate:

- 1 token  $\approx$  4 characters in English
- 1 token  $\approx$  0.75 words
- 100 tokens  $\approx$  75 words

There are also the parameters of Frequency & Presence Penalty, which determine how

#### CHAPTER 4. DESIGN AND IMPLEMENTATION

the model penalizes new words in the output. Due to the need for technical or factual content in this research, where precise language and consistent terminology are crucial, these penalties will not be applied to avoid introducing unnecessary variability and to maintain the accuracy and coherence of the generated content.

# 4.2.2 GPT Message Types

There are three types of messages used to facilitate interaction with the GPT API: system, user, and assistant messages. Each type plays a distinct role when the prompt is sent to the API to generate a response. The **system message** sets the behavior or context of the assistant and is typically used at the beginning of a conversation, usually set once. This helps guide the model to align its responses with a desired style, context, or focus. The **user message** represents the input from the end-user and is used to provide questions, prompts, or instructions that require a response from the model. The **assistant message** is the model's response. This message type is particularly useful for maintaining consistency when multiple prompts are needed within the same session, allowing for follow-up questions and coherent dialogue.

**System message:** "You are a helpful assistant and you must base your answers on facts and list them as well."

User message: "What is your favorite food?"

**API Response:** "The capital of France is Paris. Here are some factual details about Paris

1. Paris is located in the north-central part of France along the Seine River.

2. ...

3 ...

**System message:** "You are a helpful assistant and you must base your answers on facts and list them as well."

User message: "What is your favorite food?"

Assistant message: "The capital of France is Paris. Here are some factual details about Paris

1. Paris is located in the north-central part of France along the Seine River.

2. ...

3 ...

User message: "I liked that format, please use that one as well for my next question: Who is the 42nd President of the United States?"

# 4.2.3 GPT Models

The **Context Window** refers to the maximum number of tokens (words or parts of words) that the model can consider at the same time, meaning this is how much information the model can "remember" when generating a response. This includes both the input prompt and the generated output.

The **Max Output** refers to the maximum number of tokens that the model is allowed to generate in response to a given prompt.

The **Training Data Cut-Off** is the date up to which the model has been trained on available data, which will affect the model's knowledge of events, facts, and trends.

Model	Context Window/Input (Tokens - Words)	Max Output (Tokens - Words)	Training Data Cutoff
GPT-3.5-Turbo-0125	16.385-10.000	4.096-3.000	Sep 2021
GPT-4-Turbo-2024-04-09	128,000-96.000	4.096-3.000	Dec 2023
GPT-4o-2024-05-13	128.000-96.000	4.096-3.000	Oct 2023

Table 4.1: Used Models (1 Token is roughly 4 characters, the average word length is 4.7 characters in English, so 1 Token is roughly 3/4 of a word in English.)

# 4.2.4 Global GPT settings for this thesis

Temperature: Set to 0.1.

**Reason**: A lower temperature reduces randomness, resulting in more consistent and predictable outputs. Since the goal is to generate factual questions where creativity is not a priority, a low-temperature setting is ideal.

# 4.3 Phase 1: Model Choice, Formatting, and MCQ Creation

#### 4.3.1 Naive Prompt Testing

In this first phase, I conducted the experiments as if I had no prior knowledge of using the API or understanding of prompt engineering techniques, but with a general idea of the desired outcome. This approach aimed to observe how effectively the models could perform basic multiple-choice question (MCQ) generation tasks with minimal guidance and expertise. The focus was particularly on the GPT-40 model, as it has not been previously studied for MCQ generation, to see how its performance evolves with more instructive prompts.

The primary objectives of this naive MCQ generation approach were to assess the baseline performance of the models and to choose an appropriate model early in the experimentation process. By establishing a baseline, the content relevance, accuracy, and model adaptability can be evaluated across different topics. This initial assessment also helps identify the model best suited for further development and refined prompt

engineering.

The experiments began with a simple and easily understandable topic: Geography, specifically within the context of the European Union. This choice allowed for a clear evaluation of the relevance and correctness of the generated content, as Geography is a well-defined subject with clear facts, making it suitable for testing the basic capabilities of the AI model in generating MCQs. The testing was structured to compare outputs across different versions of GPT models—GPT-40, GPT-4 Turbo, and GPT-3.5 Turbo—examining both their response times and the consistency of their replies.

#### Prompt 1: Basic MCQ Generation

Generate 3 Multiple-Choice Questions about Geography in the European Union.

GPT Response (only 1 example chosen for demonstration)
Question: What is the longest river in the European Union? A) Rhine B) Danube
C) Elbe D) Loire
Correct Answer: B) Danube

All models generated relevant questions. GPT-3.5 Turbo consistently produced faster results. The consistency in the replies varied, with some responses including unnecessary assistant messages and others failing to indicate the correct answer.

#### Prompt 2: Remove overhead

<Previous Prompt (4.3.1)>. Reply only with the questions

This prompt improved output consistency by removing the assistant message, but the indication of the correct choice was always missing. This might be due to GPT's literal interpretation of the added instruction.

#### **Prompt 3: Indicating Correct Answers**

<Previous Prompt (4.3.1)> and indicate the correct answer.

The additional instruction in the prompt led to the consistent inclusion of indicating

#### 4.3. PHASE 1: MODEL CHOICE, FORMATTING, AND MCQ CREATION

the correct answer, with GPT-4o and GPT-3.5 Turbo demonstrating efficient performance.

#### **Prompt 4: Insightful Feedback**

<Previous Prompt (4.3.1)> and provide insightful feedback as to why the correct answer is correct and why the incorrect answers are wrong.

#### **GPT** Response

Question: Which European Union country has the longest coastline?

A) Italy

B) France

C) Greece

D) Portugal

Correct Answer: C) Greece

#### Feedback:

- Correct: Greece has the longest coastline in the European Union due to its numerous islands.

- Incorrect: Italy and France have extensive coastlines but do not surpass Greece. Portugal's coastline is smaller in comparison.

The format of the MCQs and feedback was different from prompt to prompt, but the results seem decent.

#### **GPT** Models Performance

Prompt	Nr	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
Basic MCQ Generation	1	4.8s	6.9s	2.2s
Remove Overhead	2	2.8s	6.7s	2.3s
Indicating Correct Answers	3	3.1s	5.5s	2.7s
Insightful Feedback	4	12.1s	13.6s	3.86s*

Table 4.2: Model Average Speed Comparison Across Prompts. Each value is the average of 5 measurements.

\*Did not output requested number of MCQs

#### Conclusion

The naive testing approach revealed the strengths and limitations of each model in generating MCQs. GPT-3.5 Turbo consistently demonstrated rapid response times, but as seen in Table 4.2, the output of this model struggled to produce the requested amount of MCQs consistently in Prompt 4.

#### CHAPTER 4. DESIGN AND IMPLEMENTATION

The output of the final of this Phase seems decent. It fulfills some MCQ requirements 3.3 such as having a clear and relevant stem, and answers presented in logical order (a,b,c,d). But many of the aspects related to plausibility and relevancy of the distractors, whether there is a clear and single correct answer, etc. will not be verified yet. Additionally, the format of the output is not consistent, and a structured format such as JSON will be required for downstream processing and integration with external applications.

It is clear that with each prompt refinement, the output and consistency are improved across the models. These findings set the stage for more advanced prompt engineering techniques in subsequent experiments.

#### Phase 1: Final Prompt

Generate 3 Multiple-Choice Questions about Geography in the European Union. Reply only with the questions, indicate the correct answer, and provide insightful feedback as to why the correct answer is correct and why the incorrect answers are wrong.

# 4.3.2 Applying Prompt Engineering Techniques for Consistent Feedback Format

Building on the findings from Phase 1, this phase aims to enhance the consistency of the feedback format in the generated MCQs using advanced prompt engineering techniques. The objective is to produce a reliable output format that can facilitate downstream processing and integration with external applications.

#### **Prompt 5: Consistent Feedback Format**

Generate 3 Multiple-Choice Questions about Geography in the European Union. Reply only with the questions, and indicate the correct answer. **Provide, separately, insightful feedback as to why the correct answer is correct and why the incorrect answers are wrong.** 

#### 4.3. PHASE 1: MODEL CHOICE, FORMATTING, AND MCQ CREATION

Format 2
0.1
Q1:
Distractors
Correct Answer: B)
Feedback for Q1:
- Correct Answer (B): Explanation
- Incorrect Answers:
- A): Explanation
- C): Explanation
- D): Explanation

Reproducing the feedback format consistently without explicit prompt engineering is not possible. However, the refined prompt improved consistency in the output format, addressing some issues noted in Phase 1.

#### Prompt 6: Using Template Pattern for output consistency

The user message is getting quite loaded and I would now like to introduce an example to guide the response of the model. I will offload the instructions to the system message, whose purpose is to set the behavior or context of the assistant and keep the necessary information from the user to a minimum. Some minor adjustments will also be made.

#### CHAPTER 4. DESIGN AND IMPLEMENTATION

**System message:** Generate 3 MCQs for a given Topic. You should only reply with the questions. After each question, indicate the correct answer and provide insightful feedback that explains why the correct answer is correct and why the incorrect answers are wrong. Here is an example of an MCQ and the format that you should follow.

1. Which European Union country has the longest coastline?

A) Italy

B) France

C) Greece

D) Portugal

Correct Answer: C) Greece

Feedback:

- Correct: Greece has the longest coastline in the European Union, largely due to its numerous islands, which significantly extend its coastal length to about 13,676 km.

- Incorrect: Italy and France also have extensive coastlines, but they do not surpass Greece. Portugal, while having a significant coastline, is much smaller in comparison to the other options.

User message: Geography in the European Union.



Offloading instructions to the system message proved effective in getting consistent output that is also aligned with the format that is indicated in the prompt. GPT-3.5 Turbo remains the fastest model but always fails to generate the correct number (3) of MCQs. A consistent response format was achieved by employing the Template Pattern (3.2.2). It is important to note that the goal was not to finalize a specific format but to evaluate the effectiveness of this pattern in ensuring output consistency.

# Determining a suitable JSON format for downstream processing and integration with external applications.

In designing the JSON format for an MCQ, I aimed to create a structure that is both intuitive and flexible, suitable for integration with various external applications. This format includes key elements such as the question text, answer options, a subtle hint, and detailed feedback (which is a form of the Fact Check List Pattern 3.2.2) for each option, ensuring comprehensive data representation. By assigning a unique identifier ('id') to each question, tracking and management within databases will be easier. The design with clearly labeled fields for options and feedback, enhances readability and supports straightforward parsing and manipulation in different programming environments. Additionally, this format will allow for easy addition of extra keys if required.

```
{
"id": "Question Number",
"question": "Sample Question Text",
"options": {
   "A": "Option A",
   "B": "Option B",
    "C": "Option C",
    "D": "Option D"
},
"hint": "Subtle Hint for the correct answer",
"correct_answer": "A",
"feedback": {
    "correct": "Explanation for the correct answer",
    "incorrect": {
       "B": "Explanation for incorrect answer B",
       "C": "Explanation for incorrect answer C",
       "D": "Explanation for incorrect answer D"
       }
   }
}
```

#### **Prompt 7: Achieve JSON Format**

To avoid using up too much space for the JSON Format in all prompts from now on, I will use {example\_format} as a placeholder for the previously detailed JSON structure.

#### CHAPTER 4. DESIGN AND IMPLEMENTATION

**System message:** Generate 3 MCQs for a given Topic. You should only reply with the questions. After each question, indicate the correct answer and provide insightful feedback that explains why the correct answer is correct and why the incorrect answers are wrong. Use the following example JSON format for your output for each MCQ.

#### {example\_format}

#### User message:

Geography in the European Union.

```
GPT Response
```

```
{
    "id": 1,
    "question": "Which city hosts the headquarters of the

→ European Parliament?",

    "options": {
        "A": "Brussels",
        "B": "Strasbourg",
        "C": "Luxembourg City",
        "D": "Frankfurt"
   },
    "correct_answer": "B",
    "hint": "This city is located in northeastern France.",
    "feedback": {
        "correct": "The European Parliament has its official
            → seat in Strasbourg, France.",
        "incorrect": {
            "A": "Brussels hosts the European Commission and the
                \hookrightarrow Council of the European Union but not the
                ↔ official seat of the European Parliament.",
            "C": "Luxembourg City hosts the European Court of
                \hookrightarrow Justice and other EU institutions but not the

→ Parliament.",

            "D": "Frankfurt hosts the European Central Bank but
                \hookrightarrow not the European Parliament."
       }
   }
}
```

**Note:** The model now consistently produces the requested JSON format, but only one MCQ per iteration were generated for all models. Explicit instructions specifying the

number of questions might be needed for consistent output.

#### **GPT Models Performance**

Prompt	Nr	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
Consistent Feedback Format	5	9.7s	21.0s	2.7s*
Using Template Pattern	6	8.3s	12.2s	1.8s*
Achieving JSON Format	7	3.9s*	7.3s*	3.0s*

Table 4.3: Model Average Speed Comparison Across Prompts. Each value is the average of 5 measurements.

\*Did not output requested number of MCQs

#### Conclusion

Phase 2 demonstrated that using the Template Pattern effectively ensured a consistent feedback format in the generated MCQs. Although a desirable JSON Format was achieved, all models consistently produced only one MCQ instead of the requested three. In subsequent phases, the prompts must be refined to address this issue. However, before proceeding to that step, I will focus on selecting the most suitable model for further development.

# 4.3.3 Model Evaluation and Selection

Prompt	Nr	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
Basic MCQ Generation	1	4.8s	6.9s	2.2s
Remove Overhead	2	2.8s	6.7s	2.3s
Indicating Correct Answers	3	3.1s	5.5s	2.7s
Insightful Feedback	4	12.1s	13.6s	3.86s*

Table 4.4: Model Average Speed Comparison Across Prompts. Each value is the average of 5 measurements.

\*Did not output requested number of MCQs

Prompt	Nr	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
Consistent Feedback Format	5	9.7s	21.0s	2.7s*
Using Template Pattern	6	8.3s	12.2s	1.8s*
Achieve JSON Format	7	3.9s*	7.3s*	3.0s*

Table 4.5: Model Average Speed Comparison Across Prompts. Each value is the average of 5 measurements.

\*Did not output requested number of MCQs

#### CHAPTER 4. DESIGN AND IMPLEMENTATION

To see the performance of each iteration for each prompt, see Appendix .2.

In the process of evaluating and selecting the most suitable model for continued testing, several key factors were considered, including the speed, quality, and consistency of output across a series of prompts. GPT-40 emerged as the preferred choice due to its ability to consistently produce MCQs as instructed across various prompts, while also maintaining a balance between performance speed and reliability. Although GPT-3.5 Turbo demonstrated superior speed, it struggled to generate the correct number of MCQs from Prompt 4 onwards, indicating its failure to deal with more detailed prompts consistently. GPT-4 Turbo is similar to GPT-40 but was consistently slower. For instance, in the "Consistent Feedback Format" prompt, GPT-4 Turbo took 21.0 seconds on average compared to GPT-40's 9.7 seconds, which is a substantial difference.

GPT-40 is chosen for further testing, as it is not only adequate in speed compared to the other models, but because the output has been the most satisfactory as well. It is also known from other literature that GPT-40 performs better than GPT-4 in most task-specific benchmarks (3.1.1).

# 4.3.4 Few-shot prompting, refinements, and MCQ Guidelines

While the desired JSON format is achieved and the decision of which model is the most suitable going forward has been made, it is still necessary to fix the issue of GPT not producing the requested number of MCQs. It always produces 1 instead of the requested 3. After multiple small variations of the prompts, GPT still only produced 1 MCQ of the requested 3, which then leads to the suspicion that this may be due to the fact that in Prompt 7 (4.3.2), only 1 example is given to indicate the desired structure of the output.

#### Prompt 8: 3-shot prompting

An example of a JSON with 3 MCQ objects under a key "mcqs" is added to the prompt, referred to as "{example\_mcqs}" from now on. I chose a 3-shot example because it provides a balanced amount of context and should offer enough examples to guide the model without overwhelming it [43, 4]. The full 3-shot examples can be found in Appendix .1. Some parts of the instructions will be omitted from the prompt as they can be derived from the 3-shot examples.

**The following instructions will be omitted:** "You should only reply with the questions. After each question, indicate the correct answer and provide insightful feedback that explains why the correct answer is correct and why the incorrect answers are wrong."

#### System message:

Create a JSON list of num\_questions MCQs for a given topic. Use this JSON format for your response:

{example\_format}.

Here is an example of the format I would like from you for 3 MCQs:

{example\_mcqs}

User message:

Geography in the European Union.

```
Model now correctly returns the correct number of MCQs
       {
       "mcqs": [
           {
               "id": 1,
               "question": "What is the highest mountain in the
                   → European Union?",
               . . .
               . . .
           {
               "id": 2,
               "question": "Which river is the longest in the
                   ↔ European Union?",
               . . .
               . . .
           },
           \{...\},\
           {...}
      ]
  }
```

See Appendix .1 for complete output

Over 30 prompt submissions were made to confirm that due to the addition of 3-shot prompting, GPT now correctly returns the correct amount of MCQs every single time.

#### **Prompt 9: User Message Flexibility**

A simple adjustment is made to the user message to be more flexible. Instead of "Geography in the European Union", they are separated into Field of Study and Subject/-

#### CHAPTER 4. DESIGN AND IMPLEMENTATION

Topic. This format is more flexible and the inputs can be easily adapted to other subjects, such as Field of study: Computer Science and Subject/Topic: Data Structures.

#### System message:

Create a JSON list of num\_questions MCQs for a given topic. Use this JSON format for your response:

{example\_format}.

Here is an example of the format I would like from you for 3 MCQs:

{example\_mcqs}

User message: Field of study: Geography Subject/Topic: European Union

#### Prompt 10: Context setting and Persona Pattern implementation.

I implement the Persona Pattern (3.2.2), by giving GPT the role of an assessment specialist who is specialized in creating high-quality MCQs that meet educational standards and goals. This helps the LLM to generate appropriate types of output and prioritize details and qualities that are relevant within the context of the role [72].

Additionally, I'll set the context (Context Manager Pattern 3.2.2) for the input by explaining what exactly they mean.

#### System message:

You are an educational assessment specialist specialized in creating highquality multiple-choice questions that meet educational standards and goals.

You must create a JSON list of {num\_questions} high-quality multiple-choice questions that follow this JSON format: {example\_format}.

Here is an example that you can use as a direction: {example\_mcqs}

You will be given the following information:

- Field of Study: The broad academic discipline encompassing the subject matter. Use this to set the context and scope for the questions.

- Subject/Topic: The specific focus within the field of study. Use this to narrow down the content to a particular area that the questions should address.

User message: Field of study: Geography Subject/Topic: European Union There is no clear difference in the output that can be determined without any objective measurements or testing, but it was previously established that the Persona Pattern is effective (3.2.2) and that additional context can result in more consistent output and less hallucination, as LLMs have more guidance and do not have to interpret by themselves what the user input means.

#### MCQ Guidelines

To further guide GPT in the creation of high-quality MCQs with minimal writing flaws, additional MCQ guidelines are added to the prompt. The additional guidelines come from the section 3.3, where characteristics of a good MCQ based on an aggregate of studies were outlined. The guidelines that can be derived through the 3-shot examples by GPT will be omitted.

## Guidelines for the Stem - The stem should be meaningful by itself and provide a complete → idea. - Avoid ambiguous statements and including irrelevant → information in the stem. - The stem should be in the form of a question or an incomplete → sentence that the options will complete. **##** Guidelines for the Distractors - Distractors must be plausible and relevant to the question. - Distractors must not overlap with the correct answer. - Distractors should be homogeneous in content to avoid giving → away clues about the correct answer. **##** General Guidelines - Ensure the correct answer cannot be easily guessed. - Use consistent formatting across all options (e.g., same tense  $\leftrightarrow$ , structure).

#### Prompt 11: Restructuring, Human Readability

The entire prompt has been restructured and put into a markdown format. There is some evidence that such a structure can help LLMs to understand and process information more effectively [48]. But the added benefit is also that the prompt will be more readable and adjustable for humans.

The new order of the instructions is intentional. Research has shown that setting the context before moving to specific guidelines or examples results in better accuracy [42, 61].

#### System message:

You are an educational assessment specialist specialized in creating high-quality multiple-choice questions (MCQs) that meet the highest educational standards and goals.

# Information Provided to You:

- Field of Study: The broad academic discipline encompassing the subject matter. Use this to set the context and scope for the questions.

- Subject/Topic: The specific focus within the field of study. Use this to narrow down the content to a particular area that the questions should address.

#### # Task:

Create a JSON list of '{num\_questions}' high-quality multiple-choice questions (MCQs) based on the provided information. Adhere to the specified format, examples, and guidelines.

# JSON Format: Ensure each MCQ follows this format: {example\_format}

# Example MCQs: Use this example as a reference: '{example\_mcqs}'

#### # Additional MCQ Guidelines:

#### Use these guidelines in addition to the Example MCQs when creating MCQs: '{mcq\_guidelines}'

The prompt has been further refined by restructuring it and implementing a markdown format. This format of the prompt will also be easier to adapt and modify in the future. An output with 5 MCQs can be found in Appendix .1. The additional guidelines MCQ guidelines have also been added to the prompt.

#### Conclusion

Using few-shot prompting, particularly 3-shot prompting (as 3 examples of the desired output are given), consistently returns the correct number of MCQs. This might raise the question as to why the generic JSON format is still included in the prompt when there are now 3 output examples. Including a generic JSON format is necessary to ensure standardization and consistency. It provides a clear template, reduces ambiguity, and guides the model in maintaining the correct structure and content, especially if there are variations or errors in the examples. The effects of the refinements in Prompts 10 & 11 are not directly visible, but best practices were implemented to improve the prompt and enhance the output of GPT.

# 4.3.5 Restricting Information with external source

#### Prompt 12: Simplified RAG and Context Manager Pattern

In this step, content about Geography in the European Union and Data structures in the context of Computer Science were generated with the help of the Llama-3.1 language model. Exact details of the content can be found on Github [16]. GPT should use this list of facts as its source of information, meaning that all MCQs must be created with the given content. The purpose is to allow the creation of MCQs based on the content of a particular lesson, subject, or course. The requested knowledge to answer the MCQs should be available to the students. MCQs that require information out of the scope of a lesson, are not effective in testing the students' knowledge.

```
System message:
<Previous prompt (4.3.4)>
...
# Information Provided to You:
...
```

- Content: Detailed information about the subject/topic, including key concepts and significant details. Restrict your MCQs to the information provided in this content.

#### Conclusion

Similar to Retrieval-Augmented Generation (3.2.4), without a retrieval step from a large external dataset, the generation can be augmented with specific content and restricted to this context with the Context Manager Pattern (3.2.2). Even though a set of facts is passed as context to GPT, it's not possible to strictly limit it to this information. GPT could still draw from its extensive pre-trained knowledge, although less when this technique is used.

# 4.4 Phase 2: Bloom's Taxonomy

Bloom's Taxonomy is a framework developed in 1956 by Bloom et al. to categorize educational goals and learning objectives. It divides learning into levels of complexity and specificity to encourage higher-order thinking skills [13]. The taxonomy was then revised in 2001 by Anderson & Krathwohl to make it more applicable to modern educational practices [6].

These are the 6 Taxonomy levels, ordered from lowest to highest:

- 1° Remember: Assess the ability to recall facts and basic concepts.
- 2° Understand: Assess how well information is understood by asking students to interpret or explain ideas.
- 3° Apply: Assess whether students can apply the information they learned in new situations.
- 4° Analyze: Assess the student's ability to break down information into parts and understand its structure.
- 5° Evaluate: Assess the ability to evaluate situations based on criteria and standards.
- 6° Create: Assess the ability to generate new ideas or produce original work.

MCQs are not suitable for testing all of these Taxonomy levels. The higher the Taxonomy level, the more challenging it gets to craft an MCQ that can effectively assess it. Therefore I have decided to include the generation of the first three levels of learning– remember, understand, and analyze–as MCQs are most suitable for assessing these levels.

The overwhelming amount of the generated MCQs so far are recall-based and fall into the Remember-Level of the Bloom Taxonomy. Occasionally, GPT also included MCQs that fall into the Understand/Apply category.

```
Examples of generated MCQs above the Remember Level.
See Appendix 5
```

**Understand(Set 2, Q4):** What is the primary advantage of using a linked list over an array?

**Apply(Set 1, Q10):** Which data structure is most suitable for implementing a priority queue?

**Apply(Set 5, Q6):** Which traversal method would you use to process all nodes in a binary search tree in ascending order?

# 4.4.1 Modifying the 3-shot examples

The current 3-shot example (see Appendix .1) does not represent MCQs for the Understand and Apply Levels at all, as currently all examples are recall-based. Therefore, new 3-shot examples for each cognitive level will be introduced in the same format based on the medical field. The full 3-shot examples can be found on Github [16].

#### Remember 3-Shot examples.

Q1: "What is the primary function of red blood cells?"

- Q2: "What is the name of the largest bone in the human body?"
- Q3: "Which vitamin is essential for blood clotting?"

#### Understand 3-Shot examples.

Q1: "Why is insulin important in regulating blood glucose levels?"

- Q2: 'What is the role of alveoli in the respiratory system?"
- Q3: "How does the body respond to a decrease in blood pressure?"

#### Apply 3-Shot examples.

**Q1:** "A patient with type 1 diabetes accidentally administers an excess dose of insulin. What immediate intervention should be applied to prevent hypoglycemia?"

**Q2:** "A patient arrives at the emergency department with symptoms of a heart attack. Which medication should be applied immediately to improve blood flow to the heart?"

**Q3:** "A patient is diagnosed with hypertension. Which lifestyle change can be applied to help manage their condition?"

# 4.4.2 Implementing in Prompt

With these new 3-Shot examples for each category, the current prompts will be expanded, leaving us with two final versions of our MCQ generation Prompts. One where no content is provided, meaning that GPT will create the MCQs solely based on its own knowledge, and one where content is provided. While Chain-of-Thought prompting has not been mentioned specifically in this Implementation phase, the presence of this technique is apparent due to the presence of step-by-step instructions, and the provision of contextual information to establish a clear understanding of the task, specifying the exact requirements for the MCQs. This encourages the model to follow a chain of thought involving:

- 1° Understanding the context and requirements
- 2° Analyzing the subject matter and content
- 3° Creating questions that align with the specified taxonomy level and format
4° Ensuring that the questions meet the guidelines and standards

#### Prompt with and without content provided.

You are an educational assessment specialist specialized in creating high-quality multiple-choice questions (MCQs) that meet the highest educational standards and goals.

# Information Provided to You:

- Field of Study: The broad academic discipline encompassing the subject matter. Use this to set the context and scope for the questions.

- Subject/Topic: The specific focus within the field of study. Use this to narrow down the content to a particular area that the questions should address.

- Content: Detailed information about the subject/topic, including key concepts and significant details. Restrict your MCQs to the information provided in this content.

- Bloom Taxonomy Level: Defines the cognitive process or skill that the MCQs aim to assess.

#### # Bloom's Taxonomy Level to follow: {taxonomy}

# Task:

Create a JSON list of '{num\_questions}' high-quality multiple-choice questions (MCQs) based on the provided information. Adhere to the specified format, examples, and guidelines.

# JSON Format: Ensure each MCQ follows this format: {example\_format}

# Example MCQs:

Use these examples for the '{taxonomy\_type}' level in Bloom's Taxonomy as a reference: '{example\_mcqs}'

# Additional MCQ Guidelines:

Use these guidelines in addition to the Example MCQs when creating MCQs:  ${mcq_guidelines}$ 

### 4.4.3 Results

The integration of Bloom's Taxonomy Levels is working effectively and means that GPT can be used to address different levels of cognitive complexity when creating MCQs. Here are a few examples of MCQs generated in the Understand & Apply Levels:

#### 4.5. DETERMINING MCQ SOLVABILITY

#### GPT Generated MCQs for 'Understand' level in Bloom's Taxonomy

Q1: What is a key advantage of using a linked list over an array?

Q2: What is the main purpose of a hash table?

Q3: What is a key application of a priority queue?

#### GPT Generated MCQs for 'Apply' level in Bloom's Taxonomy

**Q1:** You need to implement a spell checker that quickly verifies if a word is in a dictionary. Which data structure is most appropriate for this task?

**Q2:** To implement a priority queue for a task scheduling system, which data structure should you use?

**Q3:** To store and quickly retrieve key-value pairs, which data structure would you use?

The inclusion of the other 3-shot examples with comprehension-based questions such as 'What is the primary advantage of using a linked list over an array" and applicationbased questions such as 'Which data structure is most suitable for implementing a priority queue?", now instruct GPT to also create MCQs that challenge students' ability to comprehend conceptual differences but also apply their understanding to solve practical problems. These examples serve as templates to reinforce the model's ability to produce relevant questions aligned with educational objectives.

## 4.5 Determining MCQ Solvability

A separate prompt is created that tasks the model with determining whether an MCQ is solvable. This will be helpful to see how reliable the generated MCQs based on content are. The validity of this prompt will be determined in the Evaluation Chapter 5.

#### System message:

You are an educational assessment specialist specialized in analyzing the solvability of multiple-choice questions.

# Task: Evaluate if a Multiple Choice Question (MCQ) can be answered accurately based only on a provided description. Your task is to determine if the given description contains all the necessary information to solve the MCQ without requiring any prior or general knowledge beyond what is provided in the description.

# Description {description}.

If it can be solved, return TRUE and a direct quote of where the correct answer can be found in the Description, if not return FALSE and the reason why the MCQ cannot be solved.

Example:

{JSON Object Example}

## 4.6 User Interface and Prompt Assembly Framework

## 4.6.1 User Interface

A simple User Interface (UI) prototype was developed to demonstrate the ease of generating MCQs using the final prompt. The UI features a submission form where users can select between two output types: "Standard" and "Creative." These options correspond to different temperature settings for the GPT API in the backend, with "Standard" using a temperature (See figure 4.1) of 0.1 for more consistent and predictable outputs, and "Creative" using a temperature of 0.7 which is considered to be optimal for creative writing [53], in case the user is not satisfied with the variety of the standard settings. Upon submission, the UI displays the elapsed time since the prompt was submitted and presents all generated MCQs on the next page once the response from the API is received. Correct answers are highlighted in green, while incorrect ones are marked in red. Additionally, if a description is provided, users can check the solvability of the MCQs, which updates the solvability status and enables the download of the solvable MCQs as a JSON file. This interface ensures an intuitive and efficient experience for generating and reviewing MCQs.



Figure 4.2: Form Page

Figure 4.3: Result Page

#### 4.6. USER INTERFACE AND PROMPT ASSEMBLY FRAMEWORK

	Check Solvability	Download Solvable MCQ
Solvability Check Res	ults	
5 out of 6 MCQs are so	blvable based on the provided descrip	lon.
A patient present hypertension. Wl manage their cor	ts with elevated blood pressur hich lifestyle modification sho ndition?	e and is diagnosed with uld be applied to help
A: Increasing sodium	n intake	
B: Engaging in regu	lar physical exercise	
B: Engaging in regu C: Drinking more ca	lar physical exercise ffeinated beverages	
B: Engaging in regu C: Drinking more ca D: Smoking more fre	ffeinated beverages	
B: Engaging in regu C: Drinking more ca D: Smoking more fra Correct Answer: B	ffeinated beverages	
B: Engaging in regu C: Drinking more ca D: Smoking more fro Correct Answer: B Engaging in regular p	lar physical exercise ffeinated beverages equently hysical exercise helps lower blood pres	sure and improve heart health.
B: Engaging in regu C: Drinking more ca D: Smoking more fro Correct Answer: B Engaging in regular p Feedback for incorrect	lar physical exercise ffeinated beverages equently hysical exercise helps lower blood pres	sure and improve heart health.
B: Engaging in regu C: Drinking more ca D: Smoking more fro Correct Answer: B Engaging in regular p Feedback for incorrect A: Increasing sodiu	lar physical exercise ffeinated beverages equently hysical exercise helps lower blood pres ct answers: um intake can worsen hypertension.	sure and improve heart health.
B: Engaging in regu C: Drinking more ca D: Smoking more fro Correct Answer: B Engaging in regular p Feedback for incorrect A: Increasing sodiu C: Caffeinated bev	lar physical exercise ffeinated beverages equently hysical exercise helps lower blood pres ct answers: um intake can worsen hypertension. erages can raise blood pressure in som	sure and improve heart health.

Figure 4.4: Result Page: Solvability Checked

# 4.6.2 Framework to assemble the final MCQ generation Prompt



Figure 4.5: Final Prompt Assembly

#### 4.6. USER INTERFACE AND PROMPT ASSEMBLY FRAMEWORK

## 5 Evaluation

## 5.1 Using GPT to determine solvability of MCQs

In this phase of the research, I aim to address two research questions:

- 1° RQ1: How good is GPT at detecting recall-based MCQs that are not solvable based on the given content?
- 2° RQ2: How many recall-based MCQs that GPT creates in Prompt 12 (4.3.5) are solvable based on the given content? This will demonstrate how often GPT will draw from its extensive pre-training knowledge instead of adhering to the content.

The focus on recall-based questions in RQ1 and RQ2 is primarily due to my lack of subject matter expertise and the independent nature of this research. Manually assessing the solvability of recall-based MCQs is more feasible for me, as they are generally easier to evaluate for accuracy and relevance without requiring in-depth expertise in the subject area.

### 5.1.1 Research Question 1

**Research Question 1:** How good is GPT at detecting recall-based MCQs that are not solvable based on the given content?

#### RQ1: Methodology

I manually created 30 recall-based MCQs for each Field of Study & Subject/Topic Pair. The MCQs will always be around the same context, but some will clearly go out of the scope of the given content.

Data: 30 MCQs Field of Study: Geography Subject/Topic: European Union Solvable: 20 MCQs Unsolvable: 10 MCQs

#### 5.1. USING GPT TO DETERMINE SOLVABILITY OF MCQS

Data: 30 MCQs Field of Study: Computer Science Subject/Topic: Data Structures Solvable: 17 MCQs Unsolvable: 13 MCQs

#### **RQ1: Results**

Iteration	Found Solvable	Found Unsolvable
1	20	10
2	20	10
3	20	10
4	20	10
5	20	10
6	20	10
7	20	10
8	20	10
9	20	10
10	20	10

Table 5.1: GPT identifying solvability for Geography MCQs

Iteration	Found Solvable	Found Unsolvable
1	17	13
2	17	13
3	17	13
4	17	13
5	17	13
6	17	13
7	17	13
8	17	13
9	17	13
10	17	13

Table 5.2: GPT identifying solvability for Data Structures MCQs

**RQ1:** GPT demonstrated perfect classification accuracy in both tests, each test based on 10 iterations with 30 MCQs each. This suggests that GPT is highly effective at differentiating between solvable and unsolvable **recall-based** MCQs based on the provided content. This function of GPT will now be used to determine how many of the generated MCQs adhere to the given content.

## 5.1.2 Research Question 2

**Research Question 2:** How many MCQs of the generated MCQs are solvable based on the given content? This will demonstrate how often GPT will draw from its extensive pre-training knowledge instead of adhering to the content.

#### **RQ2:** Methodology

MCQs will be generated and the prompt from RQ1 will be used to determine their solvability. A set of 10 MCQs is generated per prompt. The total generated MCQs for this part is limited to 100, due to the limitation of manually inspecting the solvability of each MCQ.

Test Data: 5 Sets of 10 generated MCQs each Field of Study: Geography Subject/Topic: European Union Manual Assessment: All MCQs seem solvable. Full List of Questions: Appendix 4

Iteration	Found Solvable	Found Unsolvable	Question Number
1	50	0	None
2	50	0	None
3	50	0	None
4	50	0	None
5	50	0	None

Table 5.3: GPT classifying the generated MCQs' solvability

Test Data: 5 Sets of 10 generated MCQs each Field of Study: Computer Science Subject/Topic: Data Structures Manual Assessment: Found 4 MCQs to be not solvable. Full List of Questions: Appendix 5

#### 5.1. USING GPT TO DETERMINE SOLVABILITY OF MCQS

#### Manually found unsolvable MCQs

**Q1:** "Which graph traversal algorithm uses a queue to keep track of vertices to visit?"

A1: "Breadth-First Search (BFS)"

**Reason:** The description indicates that BFS uses queues, but there is no mention of how it works and that it keeps track of vertices to visit.

Q2: "What is the primary advantage of using a linked list over an array?"

**A2:** "Linked lists have a dynamic size, allowing for efficient insertions and deletions." **Reason:** The information in the Answer is present, but it's not mentioned as an 'advantage' per se, just as a characteristic of linked lists.

**Q3:** "Which operation in a circular queue connects the end of the queue back to the front?"

A3: "Wrap-around"

Reason: There is no mention of the term wrap-around.

**Q4:** 'Which traversal method would you use to process all nodes in a binary search tree in ascending order?''

A4: "In-order."

Reason: In-order method is mentioned but not explained how it works.

Iteration	Found Solvable	Found Unsolvable	Question Number
1	48	2	16, 43
2	48	2	16, 43
3	48	2	16, 43
4	48	2	16, 43
5	48	2	16, 43

Table 5.4: GPT classifying the generated MCQs' solvability

With our prompt used in RQ2, GPT detected that 48 out of 50 MCQs were solvable and 2 were not solvable. These are the questions that GPT deemed not solvable based on the Content for Data Structures.

**Question 16**: "Which operation in a circular queue connects the end of the queue back to the front?"

Answer 16: "Wrap-around"

**Question 43**: "In which scenario is a doubly linked list more advantageous than a singly linked list?"

Answer 43: "When frequent insertions and deletions are required at both ends"

#### CHAPTER 5. EVALUATION

#### GPT Result & Reasoning

**Question 16**: Which operation in a circular queue connects the end of the queue back to the front?

GPT Verdict: FALSE (not solvable)

**GPT Reason:** The description does not explicitly mention the term 'Wrap-around' or describe the specific operation in a circular queue.

**Question 43**: In which scenario is a doubly linked list more advantageous than a singly linked list?

GPT Verdict: FALSE (not solvable)

**GPT Reason:** The description does not provide a direct comparison between doubly linked lists and singly linked lists in terms of advantages for specific scenarios.

#### **RQ2:** Results

GPT was able to detect 2 out of the 4 MCQs (see 5.1.2) that I deduced were not solvable based on the given content (the exact content can be found on Github [16]). One could argue that Question 2 is solvable, but it might not be 100% clear to a student that these listed characteristics are an advantage. In Question 1, it seems like the Model used its pre-training knowledge to know that BFS keeps track of vertices to visit, and that's how it generated the unsolvable MCQ, and once again that's how it deduced that it is solvable based on the given content.

Otherwise, GPT's performance from iteration to iteration is consistent (meaning the amount of solvable/unsolvable MCQs) is constant.

Topic	Total Solvable	Total Unsolvable	GPT Solvable	GPT Unsolvable
European Union	50	0	50	0
Data Structures	46	4	48	2
Total	96	4	98	2

Table 5.5: Actual Results vs GPT Results

Out of 100 generated recall-based MCQs, 4 were not solvable, resulting in 96% of recall-based MCQs being solvable, notably all of these unsolvable ones were found in the Data Structures MCQs, indicating that the complexity of the topic may influence this statistic.

## 5.2 Assessing MCQ Quality with SAQUET

In the study by Moore et al. (see Section 3.3.2), the rule-based approach was more successful in determining the quality of common item-writing flaws compared to GPT-4 when compared to the evaluations of human experts. This led to further enhancements and the introduction of SAQUET [45] - an automatic question usability evaluation toolkit

#### 5.2. ASSESSING MCQ QUALITY WITH SAQUET

- which further refined the rule-based approach.

These are the 19 Item-Writing Flaws (IWFs) that SAQUET analyzes, many of which have been covered in my section about the characteristics of a good MCQ (see 3.3.1).

Item-Writing Flaw	An Item Is Flawed If
Longest Option Correct	The correct option is noticeably longer [>25%] and includes more
	detailed information than the other distractors, as this clues students
	to this option.
Ambiguous Information	The question text or any of the options are written in an unclear way
	that includes ambiguous language.
Implausible Distractors	Any included distractors are implausible, as good items depend on
	having effective distractors.
True or False	The options are a series of true/false statements.
Absolute Terms	It contains the use of absolute terms (e.g., never, always, all) in the
	question text or options.
Complex or K-type	It contains a range of correct responses that ask students to select
	from a number of possible combinations of the responses.
Negatively Worded	The question text is negatively worded, as it is less likely to measure
	important learning outcomes and can confuse students.
Convergence Cues	Convergence cues are present in the options, where there are different
	combinations of multiple components to the answer.
Lost Sequence	The options are not arranged in chronological or numerical order.
Unfocused Stem	The stem is not a clear and focused question that can be understood
	and answered without looking at the options.
None of the Above	One of the options is "none of the above", as it only really measures
	students' ability to detect incorrect answers.
Word Repeats	The question text and correct response contain words only repeated
	between the two.
More Than One Correct	There is not a single best answer, as there should be only one answer.
Logical Cues	It contains clues in the stem and the correct option that can help the
	test-wise student to identify the correct option.
All of the Above	One of the options is "all of the above", as students can guess correct
	responses based on partial information.
Fill in the Blank	The question text omits words in the middle of the stem that students
	must insert from the options provided.
Vague Terms	It uses vague terms (e.g., frequently, occasionally) in the options, as
	there is seldom agreement on their actual meaning.
Grammatical Cues	All options are not grammatically consistent with the stem, as they
	should be parallel in style and form.
Gratuitous Information	It contains unnecessary information in the stem that is not required
	to answer the question.

Figure 5.1: The 19 Item-Writing Flaw rubric criteria used by SAQUET. Taken from Moore et al. (2024), AIED 2024 [45]

SAQUET uses several techniques to determine whether an IWF is found in a multiplechoice question, including text-matching techniques, NLP-based information extraction such as Named Entity Recognition (see Section 2.3), and further enhancements provided

#### CHAPTER 5. EVALUATION

by GPT-4 [45].

- 1° RQ3: Moore et al. have identified areas for refinement in the "More than One Correct" Criteria when utilizing GPT. Building on their insights, SAQUET be enhanced further by leveraging the insights gained during this research, incorporating GPT-40 (instead of GPT-4), and more advanced prompt engineering techniques?
- 2° RQ4: Determining the MCQ Quality of the generated MCQs using SAQUET.

## 5.2.1 Research Question 3

**Research Question 3:** Moore et al. have identified areas for refinement in the "More than One Correct" Criteria when utilizing GPT. Building on their insights, can SAQUET be enhanced further by leveraging the insights gained during this research, incorporating GPT-40 (instead of GPT-4), and more advanced prompt engineering techniques?

As a quick refresher (see Section 3.3.2), Moore et al. (2024) [45] developed a tool called SAQUET to evaluate multiple-choice questions based on 19 Item-Writing Flaws (IWFs). Experts manually evaluated these IWFs for 271 MCQs and SAQUETs' evaluation is then compared to the expert's evaluation. They achieved a 94.13% overall matching rate and a 38% exact matching rate.

I will know whether this is achieved if I achieve a higher matching rate for the "more than one correct" criteria. This could also subsequently increase the overall and exact matching rate.

**Matching Rate:** This indicates how many of the flaws were detected by SAQUET that were also detected by the experts.

**Exact Matching Rate:** This indicates how many multiple-choice questions were evaluated exactly the same by SAQUET and by the human experts.

#### **RQ3: Motivation**

These are direct quotes from the paper by Moore et al. (2024) [45] and SAQUET code that was made available in the paper introducing SAQUET [45]. Full permission to use and modify this code was granted to me.

#### About the "More Than One Correct" Criteria

"Other flaws like More than one Correct, which relied heavily on GPT-4, presented significant challenges, notably impacting the overall exact matching ratio. This flaw saw a misclassification for 50 out of 271 questions (18.5%), making it the most problematic. The challenge arose from GPT-4's difficulty in reliably identifying the correct answer for an MCQ, frequently failing to determine if a single correct option exists. However, this limitation is not inherently negative, as it does not imply the question is flawed, just that the LLM has the inability to solve it" [45]

#### 5.2. ASSESSING MCQ QUALITY WITH SAQUET

#### **Code Comments:**

"There are two ways to do this, one being to present the options of the MCQ along with the question to the LLM and ask if more than one is correct. This has a ton of false positives however, so we're going with a more basic approach of "Can the LLM correctly answer the problem", however, this does not necessarily tell us if more than one is correct, but the assumption is if the LLM gets it, then that is likely the singular correct answer. However, the LLM incorrectly answering may just mean the question is difficult and requires higher Bloom's. At some point, this criteria should be refined."

#### **RQ3: Methodology**

The assessment of both the new overall and exact matching rates is possible thanks to the generosity of Steven Moore, a PhD candidate at the Human-Computer Interaction Institute in the School of Computer Science at Carnegie Mellon University [47] and the team [45], who kindly provided us with 100 MCQs of the 271 MCQs used in the paper, along with their classifications by experts in the 19 IWFs. I would like to express my sincere gratitude to them for their invaluable contribution.

#### **RQ3: Implementation**

The original prompt tasks GPT-4 with determining whether there is a possible correct answer to the question or not, prompting it to reply with "Yes" or "No". The reasoning behind this approach by Moore et al. is that the other approach of presenting the options of the MCQs along with the questions to the LLM and asking if more than one is correct leads to a ton of false positives [45], as highlighted in the "Code Comments" above.

I think that the latter approach was tackled with the wrong focus, as it requires GPT-4 to do multiple tasks (may be due to the constraints of GPT-4), and propose a modified alternative approach:

Instead of asking GPT to determine if there is more than one option that is correct when presenting him with the question and the options, GPT will be instructed to just solve the MCQ and return the choices that it thinks to be correct. Then it will be determined independently from GPT whether there was more than one option. In this approach, GPT only focuses on answering the question, and not on also determining whether there is more than one correct choice.

#### **RQ3: Result**

After manually verifying the original SAQUET's performance on the provided 100 MCQs, I got a 93.89% matching rate and a 35% exact matching rate, which closely aligns with the matching rates and exact matching rates of evaluating 271 MCQs mentioned in the paper [45].

#### CHAPTER 5. EVALUATION

The matching rate to the expert evaluation of SAQUET's evaluation in the "More than one correct" criteria is 68%. 8 of the 19 IWFs use GPT-4 in some way during classification. The majority use them as a last verification step if a good decision cannot be made purely based on the rule-based approach. Simply changing the Model used to GPT-40 actually yields considerably worse results overall, highlighting that GPT-40 is not superior to GPT-4 in this specific context. The worse results of GPT-40 also stem from the "More than one correct" criteria, dropping from a 68% matching rate to 60%. So, simply using GPT-40 will not improve the performance of SAQUET.

Due to GPT-4o's worse results in SAQUET when simply changing it as the go-to model for all criteria in SAQUET, GPT-4o will only be used for the "more than one correct choice" criteria. The reason that I will not test the performance with GPT-4 in this new approach is due to the limitation that GPT-4 does not support JSON output.

This modified version of SAQUET will now be referred to as mSAQUET. mSAQUET improved the evaluation of SAQUET in the "more than one correct" criteria, going from an original 68% matching rate to expert evaluation, to a 98% matching rate in this particular criteria. As this criterion was the most problematic one, the exact matching rate also increased drastically, as mSAQUET now has an exact matching rate of 44% compared to 35% before and the overall matching rate increased from 93.89% to 95.47%



Figure 5.2: Comparison of Matching Rates

As LLMs are inherently Black Boxes, one cannot exactly determine what the cause of this drastic improvement is. However, I hypothesize that the additional layer of decisionmaking involved in the approach, where GPT was tasked with determining if more than one correct answer exist, introduced unnecessary complexity. GPT not only had to assess

#### 5.2. ASSESSING MCQ QUALITY WITH SAQUET

the correctness of each individual option but also had to evaluate whether multiple options were correct, thereby increasing the cognitive load and potential for errors.

In contrast, the alternative approach simplifies the task by directing GPT-4 to focus solely on identifying the correct answers. This reduction in complexity likely minimizes cognitive strain on the model, allowing it to perform more effectively and with greater accuracy. By removing the secondary task of evaluating the number of correct answers, the model can concentrate fully on its core strength: understanding and answering the question based on its underlying knowledge and reasoning. This must not be confused with the approach taken in RQ1, as the model is only asked to determine if a given description is sufficient to answer an MCQ. This task is singular as the model only needs to assess the sufficiency of the information provided in the description.



Figure 5.3: Reducing cognitive load by focusing on a single task

## 5.2.2 Research Question 4

**Research Question 4:** Determining the MCQ Quality of the generated MCQs using the modified SAQUET from RQ3.

#### **RQ4:** Methodology

I decided to continue with our **Data Structures** content, but no longer use the Geography - European Union content as that content was not suitable for higher levels of Bloom's Taxonomy. Two distinct fields of study and subjects, with content that is suitable for higher levels of Bloom's Taxonomy are added: **Linear Algebra** and **General Medicine**. The content for these additional subjects was also generated with the help of the Llama-3.1 model.

These are the Data sets, divided into 6 categories, that will be used: (The generated MCQs can be found on Github [16])

## Category 1 Remember Level Bloom's Taxonomy - Content Provided 60 Generated MCQs 20 Data Structures, 20 General Medicine, 20 Linear Algebra

#### CHAPTER 5. EVALUATION

#### Category 2

Understand Level Bloom's Taxonomy - Content Provided

- 60 Generated MCQs
- 20 Data Structures, 20 General Medicine, 20 Linear Algebra

#### Category 3

Apply Level Bloom's Taxonomy - Content Provided 60 Generated MCQs 20 Data Structures, 20 General Medicine, 20 Linear Algebra

#### Category 4

Remember Level Bloom's Taxonomy - No Content Provided 60 Generated MCQs 20 Data Structures, 20 General Medicine, 20 Linear Algebra

#### Category 5

**Understand** Level Bloom's Taxonomy - **No Content Provided** 60 Generated MCQs

20 Data Structures, 20 General Medicine, 20 Linear Algebra

#### Category 6

Apply Level Bloom's Taxonomy - No Content Provided60 Generated MCQs20 Data Structures, 20 General Medicine, 20 Linear Algebra

#### 5.2. ASSESSING MCQ QUALITY WITH SAQUET

#### Results

IWF / Category	1	2	3	4	5	6	Total	Total(%)
Longest Option Correct	6	22	10	8	16	2	64	20.31
Ambiguous Information	9	5	8	4	5	8	39	12.83
Implausible Distractors	5	2	9	3	0	0	19	6.03
True or False	0	0	0	0	0	0	0	0.00
Absolute Terms	1	10	13	2	9	1	36	11.42
Complex or K-type	1	6	12	4	9	13	45	14.28
Negatively Worded	2	0	0	0	0	0	2	0.63
Convergence Cues	0	5	1	2	5	3	16	5.08
Lost Sequence	0	0	2	0	0	0	2	0.63
Unfocused Stem	0	0	0	0	0	0	0	0.00
None of the Above	0	0	0	0	0	0	0	0.00
Word Repeats	4	6	7	4	7	2	30	9.52
More Than One Correct	2	2	2	0	1	4	11	3.49
Logical Cues	2	3	5	0	1	2	13	4.12
All of the Above	0	0	0	0	0	0	0	0.00
Fill in the Blank	0	0	0	0	0	0	0	0.00
Vague Terms	0	2	15	3	3	3	26	8.25
Grammatical Cues	1	6	0	1	3	0	11	3.49
Gratuitous Information	0	0	1	0	0	0	1	0.31
Total	33	69	85	31	59	38	315	
Total(%)	10.47	21.9	26.98	9.84	18.73	12.06		

Table 5.6: Results of SAQUET for all 6 sets (View Sets here 5.2.2)

	IWFs	% of total	Most Common IWF
		flaws	
DS	86	27.3	Ambiguous Information
DS_WC	38	12.06	Ambiguous Information
DS_NC	48	15.24	Longest Answer Correct
LA	133	42.22	Longest Answer Correct
LA_WC	75	23.81	Longest Answer Correct
LA_NC	58	18.41	Complex K Type
GM	96	30.47	Longest Answer Correct
GM_WC	76	24.12	Longest Answer Correct
GM_NC	20	6.35	Longest Answer Correct

Table 5.7: Results of SAQUET grouped by Subject/Topic and Content or No Content. DS = Data Structures, LA = Linear Algebra, GM = General Medicine WC = With Content, NC = No Content

IWF / Set Number	Remember	Understand	Understand Apply		No
				Content	Content
Total MCQs	120	120	120	180	180
Longest Option Correct	14	38	12	38	26
Ambiguous Information	13	10	16	22	17
Implausible Distractors	8	2	9	16	3
True or False	0	0	0	0	0
Absolute Terms	3	19	14	24	12
Complex or K-type	5	15	25	19	26
Negatively Worded	2	0	0	2	0
Convergence Cues	2	10	4	6	10
Lost Sequence	0	0	2	2	0
Unfocused Stem	0	0	0	0	0
None of the Above	0	0	0	0	0
Word Repeats	8	13	9	17	13
More Than One Correct	2	3	6	6	5
Logical Cues	2	4	7	10	3
All of the Above	0	0	0	0	0
Fill in the Blank	0	0	0	0	0
Vague Terms	3	5	18	17	9
Grammatical Cues	2	9	0	7	4
Gratuitous Information	0	0	1	1	0
Total	64	128	123	187	128
Total(%)	20.29	40.65	39.06	59.35	40.65

#### CHAPTER 5. EVALUATION

Table 5.8: Comparing the amount of faults for each Taxonomy Level and With Content/No Content

flaws
5
5
2
7
)
5
õ

Table 5.9: How much % of the total flaws each category represents.

#### Observation

mSAQUET detected 315 flaws across 19 item-writing flaw categories in 360 generated multiple-choice questions (see Table 5.6). This corresponds to a total of 6,840 possible flaws

 $360 \text{ MCQs} \times 19 \text{ Item Writing Flaws} = 6840 \text{ possible flaws}$ 

mSAQUET's detection resulted in a flaw detection rate of approximately 4.60%, indicating that 95.4% of all possible flaws were not present in the MCQs.

#### 5.2. ASSESSING MCQ QUALITY WITH SAQUET

Considering that mSAQUET has a 95.47% matching rate with experts—meaning that it successfully identified 95.47% of the flaws detected by experts—the flaw absence rate can be adjusted accordingly. The combination of these two rates results in approximately

$$0.954 * 0.9547 \approx 91.07\%$$

of all possible item-writing flaws, as detailed in 5.1, not being present in our generated MCQs. This high flaw absence rate shows a high level of reliability and effectiveness in the quality of the generated MCQs, indicating that the majority of potential issues that could impact the educational validity and/or fairness of the questions were successfully avoided. The result suggests that our approach to automated MCQ generation, particularly when validated with mSAQUET, produces high-quality questions that align well with expert assessments. To further enhance the reliability, future work could focus on improving the model's ability to detect and correct the more subtle or sophisticated flaws.

20.29% of the total amount of flaws were detected in the Remember Bloom's Taxonomy Level, 40.65% and 39.06% for the Understand & Apply Bloom's Taxonomy Levels respectively, indicating that our MCQ generation Prompt has the least amount of flaws when creating MCQs at this Bloom's Taxonomy Level (see Table 5.8). This aligns with the fact that by default, GPT heavily favors Remember-based MCQs, unless the instructions indicate otherwise. It may be due to GPT being most confident in creating Rememberbased MCQs as these are possibly more prevalent in its training data. As a result, GPT is likely more adept at creating questions that rely on recall rather than the higher-order levels. This suggests a potential limitation of GPT when tasked with generating MCQs that require deeper cognitive engagement. Further prompt refinement may be needed or additional training data that emphasize these higher-order thinking skills.

59.35% of the total amount of flaws were detected in the MCQs generated with content and 40.65% in the MCQs generated only based on the pre-trained knowledge of GPT. This may be due to the complexity of integrating specific contents into the generation process, which might introduce more opportunities for errors and/or inconsistencies. When the model solely relies on its internal knowledge, the produced MCQs are clearly less likely to be flawed.

The most frequent item-writing flaw was "Longest Option Correct", which comprised 20.31% of all detected flaws and was most present in the "Understand" Bloom's Taxonomy Level MCQs. Future work could address this by refining prompts to avoid length bias or by incorporating additional validation steps to ensure option lengths are balanced. The next most frequent flaw was "Complex or K-type", accounting for 14.28% of the total flaws and most present in the "Application" Bloom's Taxonomy Level MCQs. These MCQs contain a range of correct responses and ask students to select from possible combinations of the responses.

Multiple IWF's did not appear at all, such as the **Unfocused Stem** (see Table 5.1), which indicates that our Prompt always results in a stem that is clear and focused and can be understood and answered without looking at the options. **Gratuituous Information** only appeared once in all 360 MCQs - meaning that the stem rarely contains unnecessary

#### CHAPTER 5. EVALUATION

information that is not required to answer to answer the question.

Out of the 360 generated MCQs, mSAQUET did not detect any IWFs for 167 MCQs. Here's an example for each category:

#### GPT RESPONSE - Category 1: Remember - Content Provided

**Question**: Which of the following is a common symptom of hypothyroidism? **Options**:

a) Weight loss

b) Increased heart rate

c) Fatigue

d) Hyperactivity

#### GPT RESPONSE - Category 2: Understand - Content Provided

**Question**: Summarize the key difference between a min-heap and a max-heap. **Options**:

a) A min-heap has a parent node with a value greater than or equal to its children, while a max-heap has a parent node with a value less than or equal to its children.b) A min-heap has a parent node with a value less than or equal to its children, while

a max-heap has a parent node with a value greater than or equal to its children.

c) A min-heap is used for implementing stacks, while a max-heap is used for implementing queues.

d) A min-heap is always balanced, while a max-heap is not.

#### GPT RESPONSE - Category 3: Apply - Content Provided

**Question**: If a matrix A has a determinant of 0, what can you apply to determine if A is invertible?

#### Options:

- a) A is not invertible.
- b) A is invertible.
- c) A is singular.
- d) A is non-singular.

#### GPT RESPONSE - Category 4: Remember - No Content Provided

**Question**: What is the time complexity of accessing an element in an array by index? **Options**:

a) O(1)
b) O(n)
c) O(log n)
d) O(n<sup>2</sup>)

#### 5.2. ASSESSING MCQ QUALITY WITH SAQUET

#### GPT RESPONSE - Category 5: Understand - No Content Provided

**Question**: Explain why a matrix is invertible if and only if its determinant is non-zero.

Options:

- a) A non-zero determinant indicates the matrix has full rank.
- b) A non-zero determinant means the matrix is singular.
- c) A non-zero determinant implies the matrix has dependent rows.
- d) A non-zero determinant shows the matrix is symmetric.

#### GPT RESPONSE - Category 6: Apply - No Content Provided

**Question**: A patient with acute myocardial infarction (MI) is being treated in the emergency department. Which intervention should be applied to restore coronary blood flow?

#### Options:

- a) Beta-blockers
- b) Thrombolytic therapy
- c) ACE inhibitors
- d) Statins

In the study by Moore et al. [45], an MCQ is categorized as acceptable if there is zero or one IWF, or unacceptable if there are two or more IWF. If the same categorization is applied to the 360 generated MCQs, this results in 284 (78.88%) acceptable and 76 (21.11%) unacceptable MCQs. (The files of the results can be found on Github [16])

## 6 | Conclusion

## 6.1 Discussion

#### 6.1.1 Design and Implementation

In this thesis, I designed and implemented a framework for generating Multiple-Choice Questions (MCQs) using GPT models (see Fig 4.5). The focus was on leveraging the basic capabilities of the models without any prior fine-tuning. The objective was to understand how well these models could perform in generating educational content, specifically in the form of MCQs, under various conditions and with the help of different prompt engineering strategies.

#### Model Selection and Experimentation

Due to practical constraints and the primary goal of exploring the models' default capabilities, the decision was made to use standard GPT models without doing any custom fine-tuning, including GPT-3.5 Turbo, GPT-4 Turbo, and GPT-4o. Throughout the early phases of experimentation, GPT-4o emerged as the most suitable model for generating MCQs, due to its consistent performance in adhering to specific instructions and comparable speed to GPT-3.5 Turbo. While GPT-3.5 Turbo exhibited faster response times, it started to struggle with consistency as the prompts became more complex. GPT-4 Turbo's output seemed on par with GPT-4o but the response times were considerably slower, making GPT-4o the choice for further experimentation.

#### **Prompt Engineering and Output Consistency**

A significant portion of the design and implementation focused on the implementation and refinements of prompt engineering techniques, which were extremely important in determining the quality and consistency of the MCQs generated by GPT. The implementation of few-shot (3.2.4) prompting, particularly 3-shot examples, played a critical role in guiding the model to produce the correct number of MCQs in the desired format. The Chain-of-Thought (CoT) (3.2.4) prompting technique encouraged the model to follow a logical chain of thought, due to the presence of step-by-step instructions and provision of contextual information.

The use of the Template Pattern (3.2.2) ensured that the outputs adhered to a standardized JSON structure for downstream processing and integration with other educational tools.

#### 6.1. DISCUSSION

#### **Content-Specific Constraints**

Another key aspect of the design and implementation is the integration of contentspecific constraints using a simplified version of Retrieval-Augmented Generation (RAG 3.2.4) and context restriction with the Context Manager Prompt Pattern (3.2.2. I provided specific content for GPT to base the MCQs on and found in the testing phase (RQ2) that it was possible to ensure that the questions remained relevant to the specific content that GPT was instructed with. This is extremely important in educational settings where assessments should align closely with the taught material. While GPT generally adhered to the provided content, it would occasionally draw from its pre-trained knowledge, which shows that it's still important to review and validate the generated questions to ensure their relevance.

#### Bloom's Taxonomy

By default, the generated MCQs are recall-based when not specified. Therefore, the framework was extended to include different cognitive levels as defined by Bloom's Taxonomy. New 3-shot examples were incorporated to represent the Remember, Understand, and Apply Level. With this, the models were now also able to generate questions that assessed not only basic recall but also comprehension and application skills. This was a significant advancement, as it demonstrated that GPT can also support higher-order thinking in educational assessments.

#### 6.1.2 Results

The results from our MCQ generation process are promising. In **Research Question 1**, our study indicated that GPT exhibits perfect accuracy in determining whether a recall-based MCQ is solvable or not. For the Geography (European Union) dataset, GPT correctly classified 20 out of 30 MCQs as solvable and 10 as unsolvable across 10 iterations. For the Computer Science (Data Structures) dataset, GPT accurately identified 17 MCQs as solvable and 13 as unsolvable in each iteration.

In Research Question 2, I then used the prompt in RQ1 to determine the solvability of the generated recall-based MCQs, as in Moore et al. [46, 45] the human experts do not directly assess the solvability and therefore SAQUET does not either. 50 MCQs were generated for Geography (European Union) and 50 MCQs for Computer Science (Data Structures). I also manually assessed the solvability of these 100 MCQs and found all to be solvable for Geography (European Union) and 4 to not be solvable for Computer Science (Data Structures), hinting that the solvability of MCQs may decrease as the content gets more complex. GPT determined that all were solvable for the ones in Geography, but only detected 2 unsolvable ones in Data Structures, incorrectly classifying 2 MCQs as solvable when they were not based on the given content. This suggests that GPT is generally effective, and may occasionally rely on pre-trained knowledge to bridge gaps in the content, leading to the generation of MCQs that are not fully aligned with the provided material. Out of 160 manually verified MCQs, 158 were classified correctly by

#### CHAPTER 6. CONCLUSION

GPT, which lowers the perfect record from RQ1, leading to an accuracy of

$$\frac{158}{160} = 98.75\%$$

The findings in **Research Question 3** once again emphasize the importance of prompt engineering 3.2 and understanding language models. By approaching the "more than one correct" criteria problem [45] from a different perspective and leveraging the knowledge gained on the cognitive load and behavior of language models, I was able to introduce a modified version of the SAQUET tool, named mSAQUET, which improved the Item-writing flaw (IWF) detection rate of SAQUET in the "more than one correct" criteria - an area that the paper identified as problematic [45].

Our results suggest that the original alternative approach used by SAQUET, which required GPT to determine if more than one correct answer exists, introduced unnecessary complexity and led to a substantial rate of misclassification. I simplified the task by directing GPT to focus solely on identifying correct answers and returning them in a structured format (a JSON object) and offloaded the task of determining if there were multiple correct answers, by simply using Python code to count the amount of correct answers from GPT's response. By doing this, mSAQUET dramatically improved the accuracy of the tool. The matching rate for the "more than one correct" criteria improved from 68% in the original SAQUET to 98% with mSAQUET.

This enhancement also led to a notable increase in the overall exact match rate, from 35% to 44%, and an overall matching rate increase from 93.89% to 95.47%. The success of mSAQUET shows the importance of targeted prompt engineering to leverage the strengths of large language models like GPT. By reducing the cognitive load on the model and focusing on its capabilities more precisely, one can achieve more accurate and reliable outcomes. This approach may be broadly applicable to other areas where LLMs are used for complex decision-making tasks, which can potentially lead to further improvements in automated educational assessment tools.

Through our experiments in **Research Question 3**, I also found that GPT-40 is not clear-cut better than GPT-4, as the original SAQUET uses GPT-4, as a last verification step if a good decision couldn't be made based on the purely rule-based method, for 8 of their classifications. Simply switching the model from GPT-4 to GPT-40 yielded the worst results (See comparison: 6.1).

Finally, in **Research Question 4**, the assessment of our 360 generated MCQs using mSAQUET was great. mSAQUET showed a flaw detection rate of 4.6% (315 flaws out of 6840 possible flaws), suggesting that the majority of potential flaws were successfully avoided. Combining that with mSAQUET's matching rate (how many % of flaws were detected compared to how many the experts detected) of 95.47%, results in a 91.07% of all possible flaws absent from the MCQs. This indicates high reliability in generating high-quality MCQs.

- Bloom's Taxonomy Levels: The "Remember" level questions had the fewest flaws



Figure 6.1: Comparison of Matching Rates

(20.29%), while "Understand" and "Apply" level questions had higher flaw rates of 40.65% and 39.06% respectively. This trend suggests that GPT is better at generating simpler, recall-based questions, likely due to this type of question aligning closer with its training data. As the cognitive demand increases, more item-writing flaws are introduced, indicating that there may be a need for further refinement of the prompts or training data that includes more higher-order cognitive tasks.

- Content-Dependent vs Content-Independent MCQs: MCQs generated with content had a higher proportion of flaws (59.35%) compared to those without content (40.65%). This may reflect the increased cognitive load and complexity when integrating specific content through the simplified version of RAG 3.2.4 into the generation process, introducing more opportunities for errors. This suggests that while GPT can generate reasonable questions solely based on its pre-trained knowledge, the incorporation of external content might require more sophisticated prompt engineering to maintain quality.
- Common Item-Writing Flaws: The most frequent IWFs were "Longest Option Correct" (20.31%) and "Complex or K-type" (14.28%). These flaws were particularly prevalent in the "Understand" and "Apply" levels, respectively. The presence of these flaws indicates areas where the MCQs could be improved to avoid leading students to the correct answer or overcomplicating the question structure.
- Least common Item-Writing Flaws: Some Item-Writing Flaws never occurred, such as the "True or False" Question Type. The "Unfocused Stem" criterion never appeared as well, indicating that the stem always is clear and focused, not requiring

#### CHAPTER 6. CONCLUSION

additional context or a look at the options to be understood. The undesired MCQ or Distractor types ("All of the Above", "None of the Above") never occurred.

- Acceptability of MCQs: Using the criteria from Moore et al. [45], which considers an MCQ acceptable if it has zero or one IWF, 78.88% (284) of the generated MCQs were categorized as acceptable, while 21.11% (76) were categorized as unacceptable.
- Zero Item-Writing Flaw MCQs: Out of 360 generated MCQs, 167 were detected to have no flaws by mSAQUET, and 117 had only one flaw. These examples of Zero IWF MCQs were found across all 3 cognitive levels (Remember, Understand, Apply), illustrating that high-quality MCQs are achievable for all cognitive levels.

## 6.2 Limitations and Future Work

### 6.2.1 Limitations

This study explored the capabilities of GPT in generating high-quality multiple-choice questions related to Geography, Data Structures, Linear Algebra, and General Medicine. While the results are promising, several limitations must be acknowledged:

#### Subject Matter Expertise

As a master's student in Computer Science and Information Technologies, I possess limited expertise in both assessment creation and specific subject matters. This constraint may affect the depth of analysis in the generated MCQs. Future work would benefit from collaboration with subject matter experts to validate the accuracy and relevance of the questions. I individually created recall-based MCQs in RQ1 and evaluated their solvability based on a given Description, while also manually verifying the solvability of 100 recallbased generated MCQs in RQ2. Naturally, I could have made errors when assessing their solvability.

#### Assessment of Output Quality

The evaluation of the MCQs, particularly regarding the plausibility and relevancy of distractors and the clarity of the correct answers, solely relies on SAQUET and no human expert reviewers. The overall quality assessment could be enhanced through expert valuation.

#### Limited Scope of Subject Areas

The study focused on generating MCQs in areas such as Geography, Data Structures, Linear Algebra, and General Medicine, so the scope was limited to these specific subjects. This selection does not fully represent the diversity and complexity of content that GPT models could encounter in broader educational contexts.

#### 6.2. LIMITATIONS AND FUTURE WORK

#### Lack of User-Centered Testing

The cognitive load imposed by the generated MCQs on students was not assessed. While the study focused on generating questions across different levels of Bloom's Taxonomy, it did not measure how these questions might affect students' cognitive load during assessments.

The generated MCQs were also not tested in real-world educational settings with actual students. Therefore, the study lacks empirical data on how well these questions perform in practice, including their effectiveness in assessing student knowledge and their impact on learning outcomes.

#### **Potential Bias in Content Generation**

This study did not explore potential biases in the content generated by GPT. Given that these models are trained on vast datasets that most certainly contain biased information, there is a risk that the generated MCQs could reflect these biases.

## 6.2.2 Future Work

The findings of this study have demonstrated the capabilities of GPT-40 in generating multiple-choice questions and assessing their quality through various metrics. However, several areas for future research and development have been identified to further enhance the reliability, applicability, and educational relevance of these generated MCQs.

#### Incorporating Newer Language Models

As large language models continue to evolve, it will be important to test the generation of MCQs with newer versions of these models. Future studies could investigate whether more recent or advanced LLMs offer improvements in terms of question quality, solvability, and alignment with educational standards.

#### Try a different number of examples for Few-Shot Prompting

All of the prompts use 3-shot (3 examples of the desired output) Prompting. I suggest for future research to include more examples where each multiple-choice question contains no item-writing flaw. This could help the model to produce fewer item-writing flaws when generating MCQs. Findings from studies such as from Lu et al. (2022) [44] could be used to assess the perfect amount and order of few-shot prompts.

#### Access up-to-date data with RAG

We used a simplified version of Retrieval-augmented generation (RAG), omitting the retrieval step of the prompting technique. This approach did not fully leverage the poten-

#### CHAPTER 6. CONCLUSION

tial of RAG to access and integrate real-time up-to-date data into the generation process. Future research should explore the integration of a complete RAG framework, to allow GPT to base its responses on the most recent and relevant information available at the time of the query.

#### Enhancing MCQ Generation with CoVe

One promising direction for future research is the integration of the Chain-of-Verification (CoVe) method into the MCQ generation process. CoVe has been shown to significantly improve the accuracy of language models by reducing factual hallucinations in complex text generation tasks. By applying CoVe to MCQ generation, it may be possible to ensure that correct answers are always factually accurate.

#### **Conducting User-Centered Testing**

To validate the practical application of the generated MCQs, future research should involve testing the questions with actual students. This would allow for the calculation of key educational metrics, such as the discrimination index (how well a question differentiates between high and low-performing students), difficulty level, gathering feedback on the practical usability, and testing the alignment with educational standards. These metrics are important for understanding the educational value of the MCQs and ensuring that they are appropriately challenging for the target audience.

#### **Communicating SAQUET Results**

The promising results achieved with the modification of the SAQUET tool, particularly in improving the evaluation of the "More than one correct" criterion, will be communicated to the research team led by Moore et al., hopefully leading to further refinements. Additionally, it would be valuable to test SAQUET against other similar toolkits to verify its effectiveness and explore opportunities for cross-validation.

#### Expanding Subject Matter and Interdisciplinary Research

Given that the solvability tests in RQ1 and RQ2 were limited to "Remember" level questions due to the limitations of this study, future work should include solvability assessments for "Understand" and "Apply" Taxonomy level questions as well. This will help determine whether the framework developed in this study is effective across different levels of cognitive complexity.

Future research should expand the range of subjects included in the study to more specialized or interdisciplinary topics. The goal would be to evaluate if the findings can be generalized, answering the question of whether GPT models can generate high-quality MCQs across diverse fields of study.

#### Exploring Image Generation Capabilities for Higher Cognitive Levels

In addition to text-based MCQs, future research could explore the use of advanced image-generation capabilities to create supplementary materials such as graphs, diagrams, and charts. These visual aids could be integrated into questions to target higher levels of Bloom's Taxonomy, such as "Analyze" and "Evaluate." By incorporating generated images, the complexity and depth of the questions can be increased, challenging students to interpret and analyze data or evaluate scenarios more comprehensively. Testing the integration of image-generation tools, such as those provided by state-of-the-art models, could significantly enhance the ability of Al-generated content to address higher-order cognitive skills, thus broadening the educational applications of these technologies.

## **List of Abbreviations**

- AI Artificial Intelligence
- ML Machine learning
- **NLP** Natural Language Processing
- MCQ Multiple-Choice Question
- **LLM** Large Language Model

- GPT Generative Pre-trained Transformer
- **SAQUET** Scalable Automatic Question Usability Evaluation Toolkit [45]
- **IWF** Item-Writing Flaw

## **Appendix**

## .1 Prompting Inputs & Outputs

```
3-shot example
  {
      "mcqs": [
          {
              "id": 1,
              "question": "What is the highest mountain in the

→ European Union?",

              "options": {
                  "A": "Mont Blanc",
                  "B": "Mount Elbrus",
                  "C": "Matterhorn",
                  "D": "Grossglockner"
              },
              "correct_answer": "A",
              "hint": "This mountain is located on the border of

→ France and Italy.",

              "feedback": {
                  "correct": "Mont Blanc, at 4,808 meters, is the
                      → highest mountain in the European Union.",
                  "incorrect": {
                      "B": "Mount Elbrus is the highest mountain in
                          \leftrightarrow Europe but not in the EU, as it is
                          → located in Russia.",
                      "C": "Matterhorn is a famous peak in the Alps
                          \rightarrow but not the highest in the EU.",
                      "D": "Grossglockner is the highest mountain
                          \leftrightarrow in Austria but not in the EU."
                  }
              }
          },
          {
              "id": 2,
              "question": "Which river is the longest in the
                  → European Union?",
```

```
"options": {
       "A": "Rhine",
       "B": "Elbe",
       "C": "Danube",
       "D": "Seine"
   },
    "correct_answer": "C",
    "hint": "This river flows through 10 countries
        \hookrightarrow before emptying into the Black Sea.",
    "feedback": {
        "correct": "The Danube River, at approximately
           → 2,860 kilometers, is the longest river in
           ↔ the European Union.",
       "incorrect": {
           "A": "The Rhine is an important river but
               → shorter than the Danube.",
           "B": "The Elbe is significant but not the
               \hookrightarrow longest river in the EU.",
           "D": "The Seine is mainly in France and much
               \rightarrow shorter than the Danube."
       }
   }
},
{
    "id": 3,
    "question": "What is the northernmost point of the
        → European Union?",
    "options": {
       "A": "Nuorgam, Finland",
       "B": "Reykjavik, Iceland",
       "C": "Oslo, Norway",
       "D": "Helsinki, Finland"
   },
    "correct_answer": "A",
    "hint": "This village is located near the border
       ↔ with Norway.",
    "feedback": {
       "correct": "Nuorgam, located in Finland, is the
           \hookrightarrow northernmost point of the European Union
           → .",
       "incorrect": {
           "B": "Reykjavik is the capital of Iceland,
               \hookrightarrow which is not part of the EU.",
           "C": "Oslo is the capital of Norway, which is
```

```
Full Output Example for Prompt 8
          {
      "mcqs": [
          {
              "id": 1,
              "question": "What is the highest mountain in the
                  → European Union?",
              "options": {
                 "A": "Mont Blanc",
                 "B": "Mount Elbrus",
                 "C": "Matterhorn",
                 "D": "Grossglockner"
             },
              "correct_answer": "A",
              "hint": "This mountain is located on the border of

→ France and Italy.",

              "feedback": {
                 "correct": "Mont Blanc, at 4,808 meters, is the
                     → highest mountain in the European Union.",
                 "incorrect": {
                     "B": "Mount Elbrus is the highest mountain in
                         \rightarrow Europe but not in the EU, as it is
                         → located in Russia.",
                     "C": "Matterhorn is a famous peak in the Alps
                         → but not the highest in the EU.",
                     "D": "Grossglockner is the highest mountain
                         \hookrightarrow in Austria but not in the EU."
                 }
             }
          },
          {
              "id": 2,
```

#### .1. PROMPTING INPUTS & OUTPUTS

```
"question": "Which river is the longest in the
        → European Union?",
    "options": {
        "A": "Rhine",
        "B": "Elbe",
        "C": "Danube",
        "D": "Seine"
   },
    "correct_answer": "C",
    "hint": "This river flows through 10 countries
        \hookrightarrow before emptying into the Black Sea.",
    "feedback": {
        "correct": "The Danube River, at approximately
            \hookrightarrow 2,860 kilometers, is the longest river in
           ↔ the European Union.",
        "incorrect": {
           "A": "The Rhine is an important river but
               ↔ shorter than the Danube.",
           "B": "The Elbe is significant but not the
               → longest river in the EU.",
           "D": "The Seine is mainly in France and much
               \hookrightarrow shorter than the Danube."
       }
   }
},
{
    "id": 3,
    "question": "What is the northernmost point of the

→ European Union?",

    "options": {
        "A": "Nuorgam, Finland",
        "B": "Reykjavik, Iceland",
        "C": "Oslo, Norway",
        "D": "Helsinki, Finland"
    },
    "correct_answer": "A",
    "hint": "This village is located near the border
        ↔ with Norway.",
    "feedback": {
        "correct": "Nuorgam, located in Finland, is the
            \hookrightarrow northernmost point of the European Union
           "incorrect": {
           "B": "Reykjavik is the capital of Iceland,
```
5 MCQ Example for Prompt 11

```
{
    "mcqs": [
       {
           "id": 1,
           "question": "Which organ is primarily responsible
               → for filtering blood and producing urine?",
           "options": {
               "A": "Liver",
               "B": "Kidney",
               "C": "Pancreas",
               "D": "Spleen"
           },
           "correct_answer": "B",
           "hint": "This organ is part of the urinary system.",
           "feedback": {
               "correct": "The kidney is responsible for
                   → filtering blood and producing urine.",
               "incorrect": {
                   "A": "The liver detoxifies chemicals and
                       → metabolizes drugs, but it does not

→ produce urine.",

                   "C": "The pancreas produces enzymes and
                       → hormones, but it does not filter blood
                       \rightarrow or produce urine.",
                   "D": "The spleen filters blood as part of the
                       \hookrightarrow immune system, but it does not

→ produce urine."

               }
           }
```

### .1. PROMPTING INPUTS & OUTPUTS

```
},
{
    "id": 2,
    "question": "Which organ is the primary site of
        \hookrightarrow nutrient absorption in the digestive system?",
    "options": {
        "A": "Stomach",
        "B": "Large Intestine",
        "C": "Small Intestine",
        "D": "Gallbladder"
   },
    "correct_answer": "C",
    "hint": "This organ is divided into three parts:
        → duodenum, jejunum, and ileum.",
    "feedback": {
        "correct": "The small intestine is the primary
            \hookrightarrow site of nutrient absorption in the
           → digestive system.",
        "incorrect": {
           "A": "The stomach is primarily responsible
               → for breaking down food, not nutrient

→ absorption.",

            "B": "The large intestine absorbs water and
               \hookrightarrow electrolytes, but not the majority of

→ nutrients.",

           "D": "The gallbladder stores bile, which aids
               → in digestion, but it is not involved
               ↔ in nutrient absorption."
       }
   }
},
{
    "id": 3,
    "question": "Which organ is responsible for pumping
        → blood throughout the body?",
    "options": {
       "A": "Lungs",
        "B": "Heart",
        "C": "Liver",
        "D": "Brain"
    },
    "correct_answer": "B",
    "hint": "This organ is part of the cardiovascular

→ system.",
```

```
"feedback": {
       "correct": "The heart is responsible for pumping
           → blood throughout the body.",
       "incorrect": {
           "A": "The lungs are responsible for gas
               → exchange, not pumping blood.",
           "C": "The liver processes nutrients and
               \hookrightarrow detoxifies the blood, but it does not
               → pump it.",
           "D": "The brain controls many functions of
               \hookrightarrow the body but does not pump blood."
       }
   }
},
{
   "id": 4,
    "question": "Which organ is primarily responsible
        → for detoxifying chemicals and metabolizing
       \rightarrow drugs?",
    "options": {
       "A": "Kidney",
       "B": "Liver",
       "C": "Pancreas",
       "D": "Spleen"
   },
    "correct_answer": "B",
   "hint": "This organ produces bile, which helps
        → digest fats.",
    "feedback": {
       "correct": "The liver is responsible for
           → detoxifying chemicals and metabolizing

→ drugs.",

       "incorrect": {
           "A": "The kidney filters blood and produces
               → urine, but it does not detoxify

→ chemicals.",

           "C": "The pancreas produces enzymes and

→ hormones, but it does not detoxify

→ chemicals.",

           "D": "The spleen filters blood as part of the
               → immune system, but it does not
               ↔ detoxify chemicals."
       }
   }
```

## .1. PROMPTING INPUTS & OUTPUTS

```
},
       {
           "id": 5,
           "question": "Which organ is responsible for

→ producing insulin?",

           "options": {
               "A": "Liver",
               "B": "Kidney",
               "C": "Pancreas",
               "D": "Thyroid"
           },
           "correct_answer": "C",
           "hint": "This organ also produces digestive enzymes
               "feedback": {
               "correct": "The pancreas is responsible for

→ producing insulin.",

               "incorrect": {
                   "A": "The liver processes nutrients and
                       \hookrightarrow detoxifies the blood, but it does not

→ produce insulin.",

                   "B": "The kidney filters blood and produces
                       → urine, but it does not produce insulin
                       ↔ .",
                   "D": "The thyroid produces hormones that
                       \rightarrow regulate metabolism, but it does not
                       \hookrightarrow produce insulin."
               }
           }
       }
   ]
}
```

## .2 Speed Tables for GPT models

Iteration	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
1	4.7s	6.2s	2.2s
2	4.1s	7.3s	2.1s
3	3.6s	4.9s	2.4s
4	3.9s	7.9s	2.1s
5	7.9s	8.2s	2.4s

Table 1: Model Speed Comparison for Prompt 1 - Basic MCQ Generation

Iteration	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
1	2.8s	7.4s	2.4s
2	2.7s	7.0s	2.5s
3	2.8s	9.6s	2.3s
4	3.2s	5.3s	2.3s
5	2.4s	4.1s	2.0s

Table 2: Model Speed Comparison for Prompt 2 - Streamlined Output

Iteration	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
1	2.8s	5.9s	2.5s
2	2.9s	4.5s	2.8s
3	2.9s	4.6s	2.6s
4	3.0s	4.9s	3.1s
5	4.0s	7.8s	2.6s

Table 3: Model Speed Comparison for Prompt 3 - Indicating Correct Answers

Iteration	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
1	17s	13.5s	4.9s
2	11.8s	12.7s	2.5s*
3	9.8s	17.2s	2.6s*
4	10.8s	12.5s	2.8s*
5	11.2s	12.0s	6.5s

Table 4: Model Speed Comparison for Prompt 4 - Insightful Feedback

\*Did not output requested number of MCQs

## .2. SPEED TABLES FOR GPT MODELS

Iteration	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
1	9.7s	21s	2.7s*
2	14.3s	17.9s	1.9s*
3	7.5s	23.3s	5.1s
4	7.6s	17s	5.2s
5	8.1s	28.3s	6.1s

Table 5: Model Speed Comparison for Prompt 5 - Consistent Feedback Format

\*Did not output requested number of MCQs

Iteration	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
1	8.3s	12.2s	1.8s*
2	10.5s	11.3s	4.1s*
3	10.6s	29.3s	2.2s*
4	7.8s	12.2s	2.0s*
5	8.2s	12.5s	2.1s

Table 6: Model Speed Comparison for Prompt 6 - Using Template Pattern

\*Did not output requested number of MCQs

Iteration	GPT-4o	GPT-4 Turbo	GPT-3.5 Turbo
1	3s	7.9s	5s*
2	3.9s	7.3s	3.5s*
3	5.4s	14s	3s*
4	4.9s	9.3s	3.3s*
5	3.8s	10.1s	3.4s

Table 7: Model Speed Comparison for Prompt 7 - Achieve JSON Format

\*Did not output requested number of MCQs

## .3 Output Testing

Q1: What is the name of the highest mountain in the EU?	A1: Mont Blanc	Solvable1: Yes
03: Which country in the European Union has the largest land area?	A2: France	Solvable3: No
Q4: Where is Mont Blanc located?	A4: On the border of France and Italy	Solvable4: Yes
Q5: Which mountain range is Mont Blanc part of?	A5: The Alps	Solvable5: Yes
Q6: Which EU country is the most populous?	A6: Germany	Solvable6: No
Q7: Which EU country is made up of over 1,400 islands?	A7: Denmark	Solvable7: No
Q8: What is the name of the longest river in the EU?	A8: Danube River	Solvable8: Yes
Q9: How long is the Danube River approximately?	A9: 2,860 kilometers	Solvable9: Yes
Q10: Where does the Danube River originate?	A10: Germany's Black Forest	Solvable10: Yes
Q11: Which country in the EU is known for its fjords?	A11: Sweden	Solvable11: No
Q12: Into which sea does the Danube River empty?	A12: The Black Sea	Solvable12: Yes
Q13: What is the largest lake in the EU?	A13: Lake VĤnern	Solvable13: Yes
Q14: What is the area of Lake VĤnern?	A14: Approximately 5,655 square kilometers	Solvable14: Yes
Q15: Where is Lake VĤnern located?	A15: Sweden	Solvable15: Yes
Q16: What is the northernmost point of the EU?	A16: Nuorgam, Finland	Solvable16: Yes
Q17: What are the coordinates of Nuorgam, Finland?	A17: Approximately 70Ű 5' N latitude	Solvable17: Yes
Q18: What is the capital city of the European Union?	A18: Brussels	Solvable18: No
Q19: What is the southernmost point of mainland Spain?	A19: Punta de Tarifa	Solvable19: Yes
Q20: What is the smallest country by area in the European Union?	A20: Malta	Solvable20: No
Q21: What is the westernmost point of the EU?	A21: Monchique Islet, Portugal (Azores)	Solvable21: Yes
Q22: Which EU country is famous for its tulip fields?	A22: Netherlands	Solvable22: No
Q23: Which EU country has the longest coastline?	A23: Greece	Solvable23: No
Q24: What is the easternmost point of the EU?	A24: Cape Greco, Cyprus	Solvable24: Yes
Q25: Which mountain range forms a natural border between France and Spain?	A25: Pyrenees	Solvable25: Yes
Q26: Which sea forms the southern boundary of the EU?	A26: Mediterranean Sea	Solvable26: Yes
Q27: Which ocean forms the western boundary of the EU?	A27: Atlantic Ocean	Solvable27: Yes
Q28: Which country borders Belgium, Germany and the Netherlands?	A28: Luxembourg	Solvable28: No
Q29: Which peninsula includes Sweden and part of Finland?	A29: Scandinavian Peninsula	Solvable29: Yes
Q30: Which EU country is known for the historic region of Transylvania?	A30: Romania	Solvable30: No

Figure 2: 30 Human Crafted MCQs based on Geography, European Union

Q1: What is the time complexity of accessing an element in an array by index?	A1: O(1)	solvable1: Yes
Q2: Which data structure uses a FIFO (First In First Out) approach?	A2: Queue	solvable2: Yes
03: What is the main difference of a linked list over an array? 04: Which data structure is used for implemention recursion?	A3: Dynamic size and ease of insertion/deletion A4: stark	Solvable3: Yes
05: What is a balanced tree in data structures?	A5: A tree where the height difference between left and right subtrees is minimal	solvable5: No
Q6: What is the time complexity of searching in a binary search tree (BST)?	A6: O(log n) in average case	solvable6: Yes
Q7: Which data structure is used to implement a LIFO (Last In First Out) system?	A7: Stack	solvable7: Yes
Q8: What is the time complexity of inserting an element in a linked list?	A8: O(1) when inserting at the head	Solvable8: Yes
Q9: Which data structure allows elements to be added or removed at both ends?	A9: Deque	solvable9: No
Q10: What is a hash table used for in data structures?	A10: Efficiently searching, inserting, and deleting key-value pairs	Solvable10: No
Q11: What is the time complexity of accessing an element in a hash table?	A11: O(1) in average case	Solvable11: Yes
Q12: What is the primary characteristic of a circular queue?	A12: The last position is connected back to the first position	solvable12: Yes
Q13: What type of traversal is used to get the contents of a binary tree in ascending order?	A13: Inorder traversal	Solvable13: No
Q14: Which data structure is used to implement a priority queue?	A14: Heap	Solvable14: Yes
Q15: What is a characteristic of a doubly linked list?	A15: Each node points to both its previous and next node	Solvable15: Yes
Q16: What is the time complexity of depth-first search (DFS) in a graph?	A16: O(V + E) where V is vertices and E is edges	Solvable16: No
Q17: What is a complete binary tree?	A17: A binary tree in which all levels are completely filled except possibly the last	solvable17: No
Q18: What is the main characteristic of a self-balancing binary search tree?	A18: It automatically keeps its height small for efficient operations	solvable18: No
Q19: What is an adjacency list used for in graphs?	A19: To store lists of adjacent vertices for each vertex	Solvable19: No
Q20: What is a trie data structure used for?	A20: Storing a dynamic set of strings where keys are usually strings	solvable20: Yes
Q21: What is a min-heap used for?	A21: Quickly finding the smallest element in a set	solvable21: Yes
Q22: What is a key benefit of using a skip list?	A22: It allows fast search within an ordered sequence of elements	solvable22: No
Q23: What is a B-tree primarily used for?	A23: Efficiently storing and managing large amounts of data	solvable23: Yes
Q24: Which traversal method is used to visit nodes of a graph level by level?	A24: Breadth-first traversal	solvable24: No
Q25: What is a sparse matrix?	A25: A matrix with a majority of its elements being zero	solvable25: No
Q26: What is a common application of the bloom filter data structure?	A26: Testing whether an element is a member of a set	solvable26: No
Q27: What is the purpose of a circular linked list?	A27: To make the last node point to the first node, forming a circle	solvable27: Yes
Q28: What does the Pop operation do in Stacks?	A28: It removes the top element from the Stack	solvable28: Yes
Q29: Which tree is used for storing information about intervals?	A29: Fenwick Trees	solvable29: Yes
Q30: Which data structure is preferred for fast lookups?	A30: Hash tables.	olvable30: Yes

Figure 3: 30 Human Crafted MCQs based on Data Structures

uropean Union? ean Union? an Union? European Union? Sec cape Greeco, Cyprus A5: Mediternaean Sea A9: Mediternaean Sea A9: Mediternaean Sea A9: Mediternaean Sea A9: Mediternaean Sea A10: Scandinavian Peninsula A2: Rhink	A2: Rhine A3: Lake Văzttern A3: Lake Văzttern A5: Punta de Tarifa, Spain A6: Cape Finisterre, Spain A7: Nuorgam, Finiand A8: Alps A9: Atlantic Ocean A10: Scandinavian Peninsula	A1: Mont Blanc A2: Rhine A3: Germany A4: Nuorgam, Finland A5: Cape Greco A6: Cape Greco A8: Alps A10: Danube A10: Danube	<ul> <li>A1: Mont Blanc</li> <li>A2: Rhine</li> <li>A3: Lake Geneva</li> <li>A3: Lake Genavik, Iceland</li> <li>A5: Punta de Tarifa, Spain</li> <li>A5: Punta de Tarifa, Spain</li> <li>A6: Monchique Islet, Portugal</li> <li>A7: Cape Roca, Portugal</li> <li>A8: Moditerranean Sea</li> <li>A10: Iberian Peninsula</li> </ul>	A1: Mont Blanc A2: Rhine A3: Lake Geneva A3: Lake Geneva A4: Nuorgan Finland A5: Gibraltar A5: Gibraltar A5: Athens, Greece A3: Athens, Greece A3: Athens, Greece A3: Danube A10: Mediterranean Sea
uropean Union? ean Union? ean Union? European Union? European Union? European Union? European Union? I border between France and Spain? is including Sweden, Finland, and Germany? and Portugal? ean Union?				ng into the North Sea?
Q1: What is the highest mountain in the E Q2: Which river is the longest in the Europ Q3: What is the largest lake in the Europe Q4: Where is the northermmost point of the Q5: What is the southermmost point of the Q6: What is the eastermmost point of the Q8: Which mountain range forms an abural Q9: Which peninsula encompasses Spain Q10: Which peninsula encompasses Spain Q10: Which river is the longest in the Europe Q2: What is the highest mountain in the Europe	The first the longer turn in curron of the first the longer in the European Union? the largest lake in the European Union? the solutermost point of the European Union? is the westermmost point of the European Union? is the westermmost point of the European Union? ocation is the eastermmost point of the European Union? mountain range forms a natural border between France and Spain? meantain shordered by several EU countries including Sweden, Finland, and Germany? penilable accompasses Spain and Portugal?	(hat is the highest mountain in the European Union? (hich river is the longest in the European Union? (hich is the northermost point of the European Union? (hat is the northermost point of the European Union? (hich location marks the southermost point of mainland Spain and the European continent? (hich location marks the southermost point of mainland Spain and the European continent? (hich location is the eastermost point of the European Union? (hich nocation is the eastermost point of the European Union? (hich nocation is the eastermost point of the European Union? (hich noutien) range forms a natural border between France and Spain? (hich sea is bordered by several EU counting, including Sweden, Finland, and Germany?) (Which river originates in the Swiss Alps and flows through Germany and the Netherlands?)	<ol> <li>Which is the highest mountain in the European Union?</li> <li>Which river is the longest in the European Union?</li> <li>What is the largest lake in the European Union?</li> <li>What is the northermost point of the European Union located?</li> <li>What is the southermost point of the European Union?</li> <li>What is the southermost point of the European Union?</li> <li>Which is the westermost point of the European Union?</li> <li>Which is the eastermost point of the European Union?</li> <li>Which is the eastermost point of the European Union?</li> <li>Which mountain range forms a natural border between France and Spain?</li> <li>Which peninsula includes Sweden and part of Finland?</li> </ol>	Q1: What is the highest mountain in the European Union? Q2: What is the longest in the European Union? Q3: What is the longest lar in the European Union? Q3: What is the southermost point of the European Union? Q5: What is the southermost point of the European Union? Q5: What is the southermost point of the European Union? Q5: Which location represents the westermost point of the European Union? Q5: Which is the asstermost point of the European Union? Q5: Which is the asstermost point of the European Union? Q2: Which river originates in the Swiss Alps and flows through Germany and the Netherlands before emptyin Q0: Which river originates in the Swiss Alps and Rewsen France and Spain? Q10: Which river originates in the Swiss Alps and Letwein, Latvia, Lithuania, Poland, and Germany?

Figure 4: 50 generated MCQs based on Geography, European Union

f teS betereneo	<ol> <li>Which of the following data structures stores data in a sequential manner?</li> <li>What is the time complexity for accessing an element in an array by its index?</li> <li>a doubly linked list, each node contains pointers to:</li> <li>Which operation is NOT typically O(1) in a linked list.</li> <li>Which operation is NOT typically O(1) in a linked list.</li> <li>What is the following data structures follows a Last In, First Out (LIFO) access pattern?</li> <li>Which operation is Not vantage of vusing a hash table?</li> <li>Which type of tree ensures that the tree remains balanced after insertions and deletions?</li> <li>What is the time oundexity for inserting an element in a binary search tree (BST) in the average case?</li> <li>What structure is most suitable for inhomentin to a binary search.</li> </ol>	A1: Array 22: O(n) 22: D(n) 42: The next node only 43: Insertion at the beginning 43: Queue 55: Queue 77: Binary Search Tree 88: Depth-First Search (DFS) 40: O(1) 40: Array	Solvable1: Yes Solvable2: Yes Solvable2: Yes Solvable3: Yes Solvable5: Yes Solvable6: Yes Solvable6: Yes Solvable6: Yes Solvable8: No Solvable9: Yes
Generated Set 2	01: Which of the following data structures is characterized by a Last In, First Out (LIFO) access pattern? 02: What is the time complexity for accessing an element by index in an array? 03: What is the primary advantage of using a linked list over an array? 04: What is the primary advantage of using a linked list over an array? 05: Which of the following operations on a stack has a time complexity of 0(1)? 05: Which of the following operations on a stack start in an array? 06: In a binary search tree (BST), what is the inthe complexity for searching an element in a hear start element in a binary search operation in a circular queue complexity for searching an element in a binary such to the following operation in a circular queue back to the following a structure is specifically used for dynamic sets of strings and supports efficient prefix-based search operations? 03: Which data structure is most for implementing association a stray and provides average O(1) time complexity for search operations? 04: Which data structure is specifically used for dynamic sets of strings and supports efficient prefix-based searches? 05: Which data structure is most suitable for implementing a strated first search (BFS) algorithm?	11: Queue 22: O(1) 22: Stack 25: Stack 44: Fixed 55: Push 25: Push 20: O(1) 20: O(1) 20: Array 20: Bihary Search Tree 210: Stack 210: Stack	Solvable1: Yes Solvable2: Yes Solvable2: Yes Solvable4: No Solvable5: Yes Solvable5: Yes Solvable5: Yes Solvable5: Yes Solvable9: Yes Solvable9: Yes
Generated Set 3	Q1: Which data structure is best suited for implementing a LIFO (Last In, First Out) access pattern? Q2: Mich data structure does each node point to the next node, and possibly to the previous node? Q3: Which data structure is most appropriate for implementing a priority queue? Q4: Which data structure is most appropriate for implementing a priority queue? Q5: Which data structure is most appropriate for implementing a priority queue? Q5: Which data structure is most appropriate for implementing a priority queue? Q5: Which data structure is particularly useful for accessing an element by index in an array? Q5: Which data structure is particularly useful for dynamic sets of strings and efficient prefix-based searches? Q2: Which data structure uses a hierarchical parent-child relationship with a root node? Q8: Which data structure is used to implement associative arrays and provides average O(1) time complexity for search operations?	11: Queue 23: Insertion 24: Insertion 24: Stack 25: O(1) 25: O(1) 25: O(1) 26: O(1) 27: Hash Table 20: O(1) 20: Stack 210: Stack	Solvable1: Yes Solvable2: Yes Solvable2: Yes Solvable4: Yes Solvable4: Yes Solvable5: Yes Solvable5: Yes Solvable8: Yes Solvable9: Yes Solvable9: Yes
Generated Set 4	<ol> <li>Which of the following data structures is characterized by a Last In, First Out (LIFO) access pattem?</li> <li>In which data structure does the last node point back to the first node?</li> <li>What is the time complexity of accessing an element by index in an array?</li> <li>Which type of tree ensures that the left child has a smaller value and the right child has a greater value?</li> <li>Which type of tree ensures that the left child has a smaller value and the right child has a greater value?</li> <li>Which operation in a linked list has a time complexity of C(1) at the beginning?</li> <li>Which data structure is most suitable for implementing function call management?</li> <li>Which has a complexity of insertion?</li> <li>What is the time complexity of insertion?</li> <li>Which data structure is most suitable for implementing function?</li> <li>Which data structure is set suitable for implementing function?</li> <li>Which data structure is set suitable for implementing function?</li> <li>Which data structure is best suited for implementing principary esarch tree?</li> <li>Which data structure is best suited for implementing principary esarch tree?</li> </ol>	<ul> <li>A1: Queue</li> <li>A2: Singly Linked List</li> <li>A3: O(n)</li> <li>A4: Binary Tree</li> <li>A4: Efficient sorting</li> <li>Access</li> <li>A7: Queue</li> <li>A8: Directed Graph</li> <li>A10</li> <li>A10</li> <li>A10</li> <li>A10</li> <li>A10</li> <li>A11</li> <li>A12</li> <li>A12</li> <li>A13</li> <li>A14</li> <li>A17</li> <li>A14</li> <li>A14</li> <li>A17</li> <li>A14</li> <li>A14</li></ul>	Solvable1: Yes Solvable2: Yes Solvable2: Yes Solvable4: Yes Solvable5: Yes Solvable5: Yes Solvable6: Yes Solvable8: Yes Solvable8: Yes Solvable8: Yes
Generated Set 5	<ul> <li>Q1: Which of the following data structures is known for its Last In, First Out (LIFO) access pattern?</li> <li>Q2: What is the time complexity for accessing an element by index in an array?</li> <li>Q3: In which scenario is a doubly linked list more advantageous than a singly linked list?</li> <li>Q3: Which data structure is best suited for implementing a priority queue?</li> <li>Q5: Which data structure is nest suited for implementing a priority queue?</li> <li>Q5: Which data structure is typically used to process all nodes in a binary search tree in ascending order?</li> <li>Q5: Which data structure is typically used to implement into a binary search tree (BST) in the average case?</li> <li>Q6: Which data structure allows for efficient prefix-based searches?</li> <li>Q10: Which data structure allows for efficient a element into a binary search tree (BST) in the average case?</li> </ul>	<ul> <li>41: Queue</li> <li>42: O(n)</li> <li>42: O(n)</li> <li>43: When memory usage needs to be minimized</li> <li>44: Array</li> <li>44: Array&lt;</li></ul>	Solvable1: Yes Solvable2: Yes Solvable2: Yes Solvable3: Yes Solvable5: Yes Solvable6: No Solvable6: No Solvable9: Yes Solvable9: Yes Solvable10: Yes

Figure 5: 50 generated MCQs based on Data Structures

# Bibliography

- Tufan Adiguzel, Mehmet Haldun Kaya, and Fatih Kürşat Cansu. Revolutionizing education with ai: Exploring the transformative potential of chatgpt. *Contemporary Educational Technology*, 15(3):ep429, 2023. doi:10.30935/cedtech/13152.
- [2] Anakin AI. Gpt-40 mini vs gpt-40 vs gpt-4: Which one to choose from? 2024. URL: https://anakin.ai/blog/gpt-40-mini-vs-gpt-40-vs-gpt-4/.
- [3] TextCortex AI. Gpt-40 vs gpt-4: Which model is better? 2024. URL: https: //textcortex.com/post/gpt-40-vs-gpt-4.
- [4] Md Rashad Alam et al. Zero- and few-shot prompting with llms: A comparative study with fine-tuned models for bangla sentiment analysis. arXiv preprint arXiv:2308.10783, 2023.
- [5] Xavier Amatriain et al. Exploring advanced prompt engineering methods in ai-driven applications. *arXiv preprint arXiv:2403.04256*, 2024.
- [6] Lorin W Anderson, David R Krathwohl, Peter W Airasian, Kathleen A Cruikshank, Richard E Mayer, Paul R Pintrich, James Raths, and Merlin C Wittrock. A revision of bloom's taxonomy of educational objectives. A Taxonomy for Learning, Teaching and Assessing. Longman, New York, 2001.
- [7] I. Ayub, D. Hamann, C. R. Hamann, et al. Exploring the potential and limitations of chat generative pre-trained transformer (chatgpt) in generating board-style dermatology questions: a qualitative analysis. *Cureus*, 15:e43717, 2023.
- [8] Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Zihao Ji, Tiezheng Yu, Wanjun Chung, et al. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. arXiv preprint arXiv:2302.04023, 2023.
- [9] Matthias Bastian. Gpt-4 has more than a trillion parameters report. https: //the-decoder.com/gpt-4-has-a-trillion-parameters/, 2024.
- [10] Staphord Bengesi, Hoda El-Sayed, Md Kamruzzaman Sarker, Yao Houkpati, John Irungu, and Timothy Oladunni. Advancements in generative ai: A comprehensive review of gans, gpt, autoencoders, diffusion model, and transformers, 2023. URL: https://arxiv.org/abs/2311.10242, arXiv:2311.10242.
- [11] T. M. Benítez, Y. Xu, J. D. Boudreau, et al. Harnessing the potential of large language models in medical education: promise and pitfalls. J Am Med Inform Assoc, 31:776–783, 2024.
- [12] S. Biswas. Passing is great: can chatgpt conduct usmle exams? Ann Biomed Eng, 51:1885–1886, 2023.

- [13] B Bloom. Taxonomy of educational objectives: The classification of educational goals handbook 1: The cognitive domain. *David Mckay*, 1956.
- [14] Cynthia Brame. Writing good multiple choice test questions, 2013. URL: https://cft.vanderbilt.edu/guides-sub-pages/ writing-good-multiple-choice-test-questions/.
- [15] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. arXiv preprint arXiv:2005.14165, 2020.
- [16] Gilles Chen. Resources of this master thesis, 2024. URL: https://github.com/ gilles-chen/MiCS\_Master\_Thesis\_Resources.
- [17] Billy Ho Hung Cheung, Gary Kui Kai Lau, Gordon Tin Chun Wong, Elaine Yuen Phin Lee, Dhananjay Kulkarni, Choon Sheong Seow, Ruby Wong, and Michael Tiong-Hong Co. Chatgpt versus human in generating medical graduate exam multiple choice questions—a multinational prospective study (hong kong s.a.r., singapore, ireland, and the united kingdom). PLOS ONE, 18(8):1–12, 08 2023. doi:10.1371/ journal.pone.0290691.
- [18] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways, 2022. URL: https://arxiv.org/abs/2204.02311, arXiv:2204.02311.
- [19] Coursera. How educators can leverage generative ai to augment teaching and learning, 2024. URL: https://blog.coursera.org/ how-educators-can-leverage-generative-ai-to-augment-teaching-and-learning/.
- [20] Coursera. Unleashing the next chapter of personalized and interactive online learning with generative ai, machine learning, and virtual reality, 2024. URL: https://blog. coursera.org/new-products-tools-and-features-2023/.
- [21] Steven Coyne, Keisuke Sakaguchi, Diana Galvan-Sosa, Michael Zock, and Kentaro Inui. Analyzing the performance of gpt-3.5 and gpt-4 in grammatical error correction, 2023. URL: https://arxiv.org/abs/2303.14342, arXiv:2303.14342.

#### **BIBLIOGRAPHY**

- [22] DataCamp. What is openai's gpt-4o? launch date, how it works, use cases more. 2024. URL: https://www.datacamp.com/blog/what-is-gpt-4o.
- [23] Jacob Doughty, Zipiao Wan, Anishka Bompelli, Jubahed Qayum, Taozhi Wang, Juran Zhang, Yujia Zheng, Aidan Doyle, Pragnya Sridhar, Arav Agarwal, Christopher Bogart, Eric Keylor, Can Kultur, Jaromir Savelka, and Majd Sakr. A comparative study of ai-generated (gpt-4) and human-crafted mcqs in programming education. ACE 2024, 2024. doi:10.1145/3636243.3636256.
- [24] G Eysenbach. The role of chatgpt, generative language models, and artificial intelligence in medical education: a conversation with chatgpt and a call for papers. JMIR Medical Education, 9:e46885, 2023.
- [25] Susan Fourtané. The three types of artificial intelligence: Understanding ai, 2019.
- [26] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024. URL: https://arxiv.org/abs/2312.10997, arXiv:2312.10997.
- [27] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C Courville, and Yoshua Bengio. Generative adversarial networks. corr abs/1406.2661 (2014). arXiv preprint arXiv:1406.2661, 2014.
- [28] T. Hirosawa and T. Shimizu. Enhancing clinical reasoning with chat generative pre-trained transformer: a practical guide. *Diagnosis (Berl)*, 11:102–105, 2024.
- [29] HyperQuiz. Hyperquiz ai question generator for teachers, 2024. URL: https: //hyperquiz.io/.
- [30] IBM. What is nlp (natural language processing)?, 2023. URL: https://www.ibm. com/cloud/learn/natural-language-processing.
- [31] IBM. How bert and gpt models change the game for nlp, 2024. URL: https:// www.ibm.com/blog/how-bert-and-gpt-models-change-the-game-for-nlp/.
- [32] HB Ilgaz and Z Çelik. The significance of artificial intelligence platforms in anatomy education: an experience with chatgpt and google bard. *Cureus*, 15:e45301, 2023.
- [33] Walturn Insight. Comparing gpt-4o, Ilama 3.1, and claude 3.5 sonnet. Walturn Insight, 2024. URL: https://www.walturn.com/insights/ comparing-gpt-4o-llama-3-1-and-claude-3-5-sonnet.
- [34] Zihao Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yanlin Xu, Etsuko Ishii, Yejin Bang, Andrea Madotto, and Pascale Fung. A survey of hallucination in natural language generation. ACM Computing Surveys, 55(12):1–38, 2023.
- [35] MDPI Information Journal. Translation performance from the user's perspective of large language models and neural machine translation systems. 2023. URL: https://www.mdpi.com/2078-2489/14/10/574.

- [36] Diksha Khurana, Aditya Koli, Kiran Khatter, and Sukhdev Singh. Natural language processing: state of the art, current trends and challenges. *Multimedia Tools and Applications*, 82(3):3713–3744, 2023. doi:10.1007/s11042-022-13428-4.
- [37] E. Klang, S. Portugez, R. Gross, R. Kassif Lerner, A. Brenner, M. Gilboa, T. Ortal, S. Ron, V. Robinzon, H. Meiri, and G. Segal. Advantages and pitfalls in utilizing artificial intelligence for crafting medical examinations: A medical education pilot study with gpt-4. *BMC Medical Education*, 23(772), 2023. doi: 10.1186/s12909-023-04752-w.
- [38] Y. S. Kıyak. A chatgpt prompt for writing case-based multiple-choice questions. *Rev Esp Edu Med*, 4:98–103, 2023.
- [39] Y. S. Kıyak, Ö. Coşkun, and I. Budakoglu. Chatgpt for generating multiple-choice questions: evidence on the use of artificial. 2023. Incomplete reference.
- [40] Yavuz Selim Kıyak and Emre Emekli. ChatGPT prompts for generating multiplechoice questions in medical education and evidence on their validity: a literature review. Postgraduate Medical Journal, page qgae065, 06 2024. arXiv:https: //academic.oup.com/pmj/advance-article-pdf/doi/10.1093/postmj/ qgae065/58120331/qgae065.pdf, doi:10.1093/postmj/qgae065.
- [41] MC Laupichler, JF Rother, IC Grunwald Kadow, et al. Large language models in medical education: comparing chatgpt- to human-generated exam questions. Academic Medicine, 99:508–512, 2024. doi:10.1097/ACM.00000000005626.
- [42] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameterefficient prompt tuning, 2021. URL: https://arxiv.org/abs/2104.08691, arXiv:2104.08691.
- [43] Shengquan Liu and Jianming Cheng. Exploring prompts in few-shot cross-linguistic topic classification scenarios. *Applied Sciences*, 13(17):9944, 2023.
- [44] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity, 2022. URL: https://arxiv.org/abs/2104.08786, arXiv:2104. 08786.
- [45] Steven Moore, Eamon Costello, Huy A Nguyen, and John Stamper. An automatic question usability evaluation toolkit. In *International Conference on Artificial Intelli*gence in Education, pages 31–46. Springer, 2024.
- [46] Steven Moore, Huy A Nguyen, Tianying Chen, and John Stamper. Assessing the quality of multiple-choice questions using gpt-4 and rule-based methods. In *European Conference on Technology Enhanced Learning*, pages 229–245. Springer, 2023.
- [47] Steven James Moore. Website of steven james moore, 2024. URL: https: //stevenjamesmoore.com/.

#### **BIBLIOGRAPHY**

- [48] Dhaval Nagar. Llms love structure: Using markdown for better pdf analysis, 2024. URL: https://www.appgambit.com/blog/ llms-love-structure-using-markdown-for-pdf-analysis.
- [49] A Ngo, S Gupta, O Perrine, et al. Chatgpt 3.5 fails to write appropriate multiple choice practice exam questions. *Academic Pathology*, 11:100099, 2024.
- [50] OpenAl. Gpt-4 | openai. https://openai.com/product/gpt-4/, 2024.
- [51] OpenAI. Hello gpt-4o, 2024. URL: https://openai.com/blog/ introducing-gpt-4o/.
- [52] OpenAI. Openai gpt documentation, 2024. URL: https://platform.openai. com/docs/.
- [53] OpenAI Developer Forum. Cheat sheet: Mastering temperature and top\_p in chatgpt api, 2023. URL: https://community.openai.com/t/ cheat-sheet-mastering-temperature-and-top-p-in-chatgpt-api/172683.
- [54] OpenAI Developer Forum. Temperature, top\_p and top\_k for chatbot responses, 2023. URL: https://community.openai.com/t/ temperature-top-p-and-top-k-for-chatbot-responses/295542.
- [55] Valentine Joseph Owan, Kingsley Bekom Abang, Delight Omoji Idika, Eugene Onor Etta, and Bassey Asuquo Bassey. Exploring the potential of artificial intelligence tools in educational measurement and assessment. *EURASIA Journal of Mathematics, Science and Technology Education*, 19(8):em2307, 2023. doi:10.29333/ejmste/ 13428.
- [56] Johan Perols. Financial Statement Fraud Detection: An Analysis of Statistical and Machine Learning Algorithms. AUDITING: A Journal of Practice Theory, 30(2):19– 50, 05 2011. arXiv:https://publications.aaahq.org/ajpt/article-pdf/ 30/2/19/46338/ajpt-50009.pdf, doi:10.2308/ajpt-50009.
- [57] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.
- [58] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.
- [59] Aditya Raghunath. Gpt-4 parameters explained: How many parameters in gpt-4 & more. https://hix.ai/hub/chatgpt/gpt-4-parameters, 2024.
- [60] ResearchGate. Gpt-4 vs gpt-3.5: A concise showdown, 2023. URL: https://www.researchgate.net/publication/369897711\_GPT-4\_vs\_ GPT-35\_A\_Concise\_Showdown.
- [61] Laria Reynolds and Kyle McDonell. Prompt programming for large language models: Beyond the few-shot paradigm. In *Extended Abstracts of the 2021 CHI Conference* on Human Factors in Computing Systems, CHI EA '21, New York, NY, USA, 2021. Association for Computing Machinery. doi:10.1145/3411763.3451760.

- [62] Pranab Sahoo, Ayush Kumar Singh, Sriparna Saha, Vinija Jain, Samrat Mondal, and Aman Chadha. A systematic survey of prompt engineering in large language models: Techniques and applications, 2024. URL: https://arxiv.org/abs/2402.07927, arXiv:2402.07927.
- [63] Iqbal H. Sarker. Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2(3):160, 2021. doi:10.1007/ s42979-021-00592-x.
- [64] Jaromir Savelka, Arav Agarwal, Christopher Bogart, and Majd Sakr. From gpt-3 to gpt-4: On the evolving efficacy of Ilms to answer multiple-choice questions for programming classes in higher education. In *International Conference on Computer Supported Education*, pages 160–182. Springer, 2023.
- [65] Maximilian Schreiner. Gpt-4 architecture, datasets, costs and more leaked, 2023. URL: https://the-decoder.com/ gpt-4-architecture-datasets-costs-and-more-leaked/.
- [66] UT Sevgi, G Erol, Y Dogruel, et al. The role of an open artificial intelligence platform in modern neurosurgical education: a preliminary study. *Neurosurgical Review*, 46:86, 2023.
- [67] David Silver, Aja Huang, Chris J Maddison, et al. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. doi: 10.1038/nature16961.
- [68] Thinkific. 22 top ai quiz and exam generators to test your students' knowledge with ease. 2024. URL: https://www.thinkific.com/.
- [69] Jorge Valente, João António, Carlos Mora, and Sandra Jardim. Developments in image processing using deep learning and reinforcement learning. *Journal of Imaging*, 9(10), 2023. URL: https://www.mdpi.com/2313-433X/9/10/207, doi: 10.3390/jimaging9100207.
- [70] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2017. URL: https://arxiv.org/abs/1706.03762, arXiv:1706.03762.
- [71] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc., 2022. URL: https://proceedings.neurips.cc/paper\_files/paper/2022/ file/9d5609613524ecf4f15af0f7b31abca4-Paper-Conference.pdf.
- [72] Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt, 2023. URL: https://arxiv. org/abs/2302.11382, arXiv:2302.11382.

## **BIBLIOGRAPHY**

- [73] Analytics Yogi. Large language models (Ilms): Types, examples, 2024. URL: https: //vitalflux.com/large-language-models-concepts-examples/.
- [74] M Zuckerman, R Flood, RJB Tan, et al. Chatgpt for assessment writing. Medical Teacher, 45:1224–1227, 2023.