Enhancing Automated Student Answer Marking:

Exploring Capabilities of LLMs Utilizing Prompt Engineering Techniques and Retrieval-Augmented Generation

By

M.A.V.V Wickramasinghe - Registration No: 2019/IS/091

T.H.H. Abeywardana - Registration No: 2019/IS/002

P.A.N.P. Nandadewa - Registration No: 2019/IS/049

Supervisor : Prof. K. P. Hewagamage

Co-supervisor : Dr. H.N.D. Thilini

This dissertation is submitted to the University of Colombo School of Computing
In partial fulfillment of the requirements for the
Degree of Bachelor of Science Honours in Information Systems

University of Colombo School of Computing
35, Reid Avenue, Colombo 07,
Sri Lanka

September 2024

Declaration

I, M.A.V.V Wickramasinghe (2019/IS/091) hereby certify that this dissertation entitled Enhancing Automated Student Answer Marking:Exploring Capabilities of LLMs Utilizing Prompt Engineering Techniques and Retrieval-Augmented Generation is entirely my own work and it has never been submitted nor is currently been submitted for any other degree.

Signature of Candidate Date :2024/09/17

I, T.H.H. Abeywardana (2019/IS/002) hereby certify that this dissertation entitled Enhancing Automated Student Answer Marking:Exploring Capabilities of LLMs Utilizing Prompt Engineering Techniques and Retrieval-Augmented Generation is entirely my own work and it has never been submitted nor is currently been submitted for any other degree.

Signature of Candidate Date : 2024/09/17

I, P.A.N.P. Nandadewa (2019/IS/049) hereby certify that this dissertation entitled Enhancing Automated Student Answer Marking:Exploring Capabilities of LLMs Utilizing Prompt Engineering Techniques and Retrieval-Augmented Generation is entirely my own work and it has never been submitted nor is currently been submitted for any other degree.

Signature of Candidate

Date :2024/09/17

I, K. P. Hewagamage, certify that I supervised this dissertation entitled Enhancing Automated Student Answer Marking:Exploring Capabilities of LLMs Utilizing Prompt Engineering Techniques and Retrieval-Augmented Generation conducted by M.A.V.V Wickramasinghe, T.H.H. Abeywardana, P.A.N.P. Nandadewa in partial fulfillment of the requirements for the degree of Bachelor of Science Honours in Information Systems.

K. P. Hewagamage 21/09/2024

Signature of Supervisor Date :

I, H. N. D. Thilini, certify that I supervised this dissertation entitled Enhancing Automated Student Answer Marking:Exploring Capabilities of LLMs Utilizing Prompt Engineering Techniques and Retrieval-Augmented Generation conducted by M.A.V.V Wickramasinghe, T.H.H. Abeywardana, P.A.N.P. Nandadewa in partial fulfillment of the requirements for the degree of Bachelor of Science Honours in Information Systems.

Dinusha

Signature of Co-Supervisor Date : 21/09/2024

Abstract

In this research, we explored the potential of Large Language Models (LLM) to enhance the automated marking process in education by utilizing LLMs' improved understanding of language and instructions following nature. We provided subject content as knowledge to increase accuracy in the marking of answers written for structured questions that are made up of theoretical subjects. Additionally, we examined the use of grading rubrics to maintain consistency and fairness in the marking process and, the use of prompt optimization techniques to enhance the accuracy by refining the prompts. Importantly, we examined the reliability and generalizability of prompts across various subjects and different questions, making the optimized prompt applicable to automated student answer marking of various theoretical subjects. Finally, detailed feedback generated utilizing the rubric grading scale that provide students with valuable insights to aid their learning journey.

The results of the study highlighted the importance of providing external knowledge within the prompt to improve the performance of Large Language Models (LLMs) like Generative Pre-Trained Transformers (GPT) in the automated grading of students' answers. The inclusion of grading rubrics, model answers, and course content significantly enhanced the accuracy of scores assigned by the LLM, reducing deviations from human evaluator scores. Particularly in theoretical subjects within the IT domain, where LLMs tend to apply vast knowledge beyond the scope of student expectations, providing course content or model answers helped define the expected answer scope and guide the LLM in determining other possible correct answers. This approach not only streamlines the marking process for academic staff but also promotes transparency and reduces human errors in marking. Additionally, prompt engineering techniques were used to further engineer the basic prompt. A detailed feedback was also provided to students at the end of the marking process. However, combining multiple prompt engineering techniques with a basic prompt did not outperform the basic prompt, suggesting the need for further exploration and refinement in prompt design strategies.

Preface

This research focuses on analyzing the capabilities of LLMs to assess the short answers written for theoretical exams. This process includes several components and some of them are optional and some are mandatory components. Question and student answers are mandatory components. With this research how the performance of LLMs would be diverse with different combinations of optional components was evaluated. The optional components that are analyzed here are the grading rubric, a model answer, and the course content.

Different OpenAI GPT models such as GPT-3.5 Turbo, GPT-3.5 Instruct, and GPT-4 were utilized for the student answer assessment. Since the course content was retrieved using a vector database, OpenAIs text-embedding-ada-002 model was used to create embeddings. This method of providing the course content falls under the Retrieval Augmented Generation (RAG) technique. Prompt engineering techniques were also incorporated to create prompts. Therefore, the impact of RAG and prompt engineering techniques is evaluated in this research. A basic prompt was used to perform student answer assessments across different subjects and different question types. How engineered prompts improved the student answer assessment process is also evaluated.

The results of the experiments conducted by our group are presented in Chapter 4. A pilot study was performed to identify the best-performing LLM and further experiments were conducted with the GPT-4 model for basic and engineered prompts with the guidance of our supervisors to generate valuable findings. With the constant guidance and supervision of our supervisor and co-supervisor, conclusions were drawn on the evaluation and final results. This piece of research work would be a great source of knowledge for future research on Automated Student Answer Assessment.

Acknowledgement

First, we would like to thank the university, University of Colombo School of Computing (UCSC) for giving us this great opportunity to carry out this research to develop our research and necessary academic skills. We would like to express our sincere gratitude to our research supervisor Prof K.P. Hewagamage, Senior professor at the University of Colombo School of Computing, and our co-supervisor Dr.H.N.D. Thilini, senior lecturer of the University of Colombo School of Computing, for providing us with valuable guidance and supervision throughout this research project. We are grateful to them for finding time to help us all the time, from the beginning to the end of this research project.

We also bestow our honor on Dr.Enosha Hettiarachchi, Mr.Kenneth Thilakarathna, and other lecturers for the guidance they offered us by providing their valuable feedback as examiners in proposal defense and interim presentations. Further, we would like to pay our sincere gratitude to Dr.Hiran Ekanayake and Dr.Lasanthi De Silva Information Systems project coordinators, and all other UCSC staff members for all the assistance they provided to make this project successful.

We are also grateful for our friends and university batchmates who assisted us in providing new insights and support for this project.

Table of Contents

List of Figures

List of Tables

List of Acronyms

AES Automated Essay Scoring

ARM Auxiliary Rationale Memory

ATM Automated Teller Machine

CoT Chain Of Thought

GPT Generative Pre-Trained Transformers

IEA Intelligent Essay Assessor

IT Information Technology

LLM Large Language Models

LSA Latent Semantic Analysis

LSTM Long Short-Term Memory

NLP Natural Language Processing

PEG Project Essay Grader

PIIT Professional Issues in IT

QWK Quadratic Weighted Kappa

RAG Retrieval Augmented Generation

UCSC University of Colombo School of Computing

Chapter 1

Introduction

1.1 Background

Assessment of student answers written for an examination is a tedious task. The traditional way of assessing student answers is performed by human evaluators. These human evaluators have their views on assessing a particular question based on their experience, preferences, and subject knowledge. Sometimes, the assessment is influenced by the student who submitted the answer, resulting in a biased assessment. Traditional student answer assessment results in many human errors which can occur due to various factors such as wrong interpretations of student answers, lack of instruction, fatigue, etc. Due to the time-consuming nature of this approach, students do not have enough time to understand their weaknesses and improve their understanding and knowledge of that particular subject. Different human evaluators have different thinking patterns which results in different interpretations of student answers and assessment criteria. Therefore, human assessment of student exam answers is subjective.

With the outbreak of covid-19 pandemic, all educational activities were transitioned to online platforms. The examination to evaluate the student's understanding of the subject knowledge was conducted via computer-based online examinations, in addition to the teaching process. Initially, the computer programs automatically marked the multiple choice questions, while the human evaluators marked the written questions through online platforms. Later, semantic similarity measures and NLP techniques were adopted to assess the written student answer. These platforms were also evolved as proctoring systems that detect students' unusual behavior to identify cheating during the examinations.

Automated student answer assessment, often referred to as automated student answer marking, belongs to the domain of Automated Essay Scoring (AES) and is employed to assess student answers. The research on AES has been started long before the outbreak of covid-19 pandemic. Project Essay Grader (PEG)[1] is considered one of the earliest studies on Automated Essay Scoring AES. Later, different kinds of feature extraction and NLP techniques such as extracting content-based features, statistical features, style-based features were adopted. Part-of-speech tagging, N-grams, semantic similarity, and sentiment analysis were some of those techniques. These strategies successfully eliminated the subjectivity and inherent bias in the human assessment of student answers. Traditional natural language processing (NLP) methods were gradually overtaken by the utilization of machine learning, vector embeddings, neural networks, and pre-trained transformer models like BERT and RoBERTa [2]. Machine learning approaches employed in the AES sector can be categorized into classification and regression tasks. Within the AES area, neural networks such as CNN, RNN, and Randomforest were also employed. The majority of the developed models achieved moderate performance in assessing student answers.

Large language models are the most recent advancement in natural language processing. Beyond BERT, The OpenAI Generative Pretrained Transformer (GPT) models have demonstrated remarkable proficiency in the understanding of languages and reasoning. GPT models are trained on vast amounts of text data, enabling them to understand and generate human-like responses across a wide range of topics and writing styles. Given the capabilities of GPT models, they are being employed for a wide range of tasks such as chatbots, data analysis, code generation, etc. Therefore, these OpenAI GPT models demonstrate significant potential in assessing written student answers.

There are other techniques such as Retrieval Augmented Generation RAG [3] which were developed with the evolution of LLM models such as OpenAI GPT models. This technique involves both retrieval and generation. An external knowledge source will be retrieved from a large dataset or a database and it will be provided as an input to LLMs. This approach significantly enhances the accuracy and reliability of LLMs. This technique is also referred to as in-context learning which helps LLMs to generate responses based on the provided context data within the prompt. This approach can be utilized to provide the latest knowledge and data without training the model specifically.

Various solutions were suggested and created in the field of AES (Automated Essay Scoring) by utilizing the reasoning capabilities of OpenAI GPT Models. The aim was to enhance the reliability and accuracy of automated assessment of student answers in language tests. [4] These models tend to generate misleading responses due to hallucinations which can be described as generating wrong responses and stating them as correct. Since there is a lack of AES research done in terms of marking short answers of students written for theoretical exams of the IT domain, this research explores the ways of enhancing this existing marking process for theoretical examinations with the capabilities of OpenAI GPT models utilizing different prompt engineering techniques and providing the subject knowledge via RAG approach. Course content is provided as it is necessary to answer the examination question based on subject knowledge. Additionally, how to provide proper feedback for each question is also explored as a feature. This procedure is useful to assess the student answers for different subjects or modules while addressing the shortcomings in manual student answer assessment such as subjectivity and biases.

1.2 Research Problem and Research Questions

1.2.1 Research Problem

There is a set of guidelines provided for assessing the student's answer to ensure the standardization of the marking process and it is called the Grading Rubric[5]. This is followed by the human evaluators when assessing the student's answers. A grading rubric is provided to guarantee that every evaluator follows a standardized marking approach, rather than using their methods. Despite adhering to these guidelines, human evaluators often make errors as a result of their different interpretations of grading rubrics and their biases. In addition, human assessors possess the necessary expertise to assess student answers, regardless of the presence of a model answer.

To enhance the student answer assessment process reduce subjectivity and biases and ensure a standardized procedure is followed throughout the student answer assessment process, the traditional procedure done with human evaluators was replaced with automated or computer-based student answer assessment. These AES methods have evolved through several phases such as traditional NLP techniques, machine learning techniques, vector embeddings, neural networks, and pre-trained transformer models such as BERT (Bidirectional Encoder Representations from Transformers). They assessed student answers based on different techniques such as keyword extraction, a bag of words, POS tagging, WordNet graphs, etc. Since LLMs and OpenAI GPT models were rapidly adopted around the world, researchers also adopted them for the AES domain due to their ability to generate human-like responses and their advanced reasoning skills. These research works mainly focused on marking language test answers written to assess the language proficiency of students.

There is a lack of research work done to assess short answers written for theoretical examinations. Despite possessing exceptional reasoning skills and language understanding, sometimes they generate hallucinations. Hallucinations refer to the inaccurate or deviated responses generated for the given input or the prompt. This reduces the amount of correct scores granted for student answers. This research explores the effectiveness of Retrieval Augmented Generation (RAG) to provide course content as subject knowledge and the use of prompt engineering techniques to reduce the amount of generated hallucinations during the AES process to mark student answers written for theoretical subjects in the IT domain. The assessment procedure of the answers varies based on the question type, but previous research failed to take this into account. Further, there are several high-performing OpenAI GPT models such as GPT-3.5-Turbo, GPT-4, and GPT-3.5-Instruct and their capabilities are different. Comparing them and finding the best-performing model for the AES process is also a useful finding. Utilizing OpenAI GPT models to provide feedback for each student's answer is also beneficial for students and is explored as an additional feature. This research will address these concerns by solving the following research questions.

1.2.2 Research Questions

There are several essential things needed for the student answer assessment and one of them is following a grading rubric. This serves as a guideline for determining how marks should be allocated for each section of the student's answer. The grading rubric should be followed while comparing the content that should be provided in the student's answer and what is already included in the students' answer. The model answer is an additional factor taken into account by human evaluators when assessing student answers. A model answer can be defined as a response that fulfills all of the requirements of the given question. It demonstrates the expected level of understanding that the students have about the subject. Evaluators can determine the level of analysis dedicated to the question by referencing the model answer. Following the same process done by the human evaluators, GPT models can be prompted by providing them with instructions, which include a question, student answer, grading rubric, and the model answer. GPT models will follow the specified instructions to assess the provided student answer given within the prompt and generate a score and feedback. Hence this research question will be able to enhance the automated student answer assessment utilizing the capabilities of OpenAI GPT models.

RQ 1: How to improve automated marking through Retrieval Augmented Generation (RAG) and using prompt engineering techniques.

Each question in a theoretical examination is formed by referring to a particular section within the course content to analyze the understanding of students. Therefore, the answer to that question resides within that particular section. Therefore, course content can be considered as a substitution for the model answer since it is also derived by referring to that section. Also providing relevant course content along with the model answer may define a better scope for the LLM model to gain knowledge and enhance the student answer assessment process. This approach can be achieved by the RAG technique. This is used to retrieve external sources of knowledge from a large database and utilize them to prompt LLMs. In addition, other prompt engineering techniques have been invented and presented as enhancements to the prompt, to generate more effective responses. This question is used to determine the potential of these techniques to enhance the automated student answer assessment process.

RQ 2: How well do different OpenAI GPT Models perform in automated marking across various subjects and different question types.

OpenAI models vary in the number of parameters they possess and the training data they are trained on. This has an impact on their performance. Employing them in the process of evaluating student answers and identifying their performance differences will lead to understanding which model excels in automated student answer assessment. As well as the model types, student answers are also available for different question types and subjects. This research question explores the ability of each model to assess student answers. Then utilize the best-performing models for the student answer assessment procedure and determine their performance across different question types and different subjects.

1.3 Aim and Objectives

1.3.1 Aim

Enhancing the automated marking process utilizing capabilities of large language models and providing individual feedback on students.

The traditional student answer marking process exhibits several drawbacks as a time-consuming and subjective process. Traditional AES models also suffer from lack of language understanding since they utilize bag-of-words, n-grams as traditional NLP techniques, regression and classification techniques as machine learning, WordNet graphs, etc. Large Language Models possess immense potential and are capable of effectively completing complex tasks that need logical thinking in a short timeframe. It helps to reduce the time taken for student answer assessment. Additionally, they incorporate logical reasoning steps that resemble the process of human decision making. At the end of the assessment process, providing feedback on student answers is also a crucial step. Students can incorporate the feedback given at the end of this process to improve themselves in that specific subject area. Since LLMs have text-generation skills, they can be utilized for giving feedback to students after assessing the answers based on the mistakes students have made when answering the questions. Feedback generation can be provided as an additional feature. This aim is intended to be accomplished through the following objectives.

1.3.2 Objectives

Following are the objectives that are in focus to achieve regarding each research question(RQ).

RQ 1: Evaluate the impact of prompt design on the accuracy of the automated marking process, identifying key elements within prompts that can improve automated marking.

A prompt serves as the means of communication with the OpenAI GPT models. These models may struggle to understand the provided prompt and sometimes interpret the same prompt differently due to the presence of ambiguous instructions. To reduce this, there are different prompt engineering techniques available and tried out by previous works. Popular techniques for prompt engineering include using a chain-of-thought approach to guide LLMs in thinking step by step, using delimiters to separate different parts of the prompt, and offering multiple examples within the prompt to help LLMs master the tasks in a few-shot approach [6]. After a basic prompt for student answer assessment has been developed, it can be engineered using the techniques discussed previously. The prompt provided is a basic prompt with simple directions, designed without employing any prompt engineering techniques. By comparing the performance of these engineered prompts with the basic prompt, we can determine the effects of prompt engineering techniques.

RQ 1: Evaluate the impact of Retrieval Augmented Generation (RAG) to enhance the automated marking process.

Based on the current state of the art, the majority of advancements in the field of AES have been achieved by assessing student answers through a comparison with a model answer. This model answer serves as a comprehensive and detailed answer that includes all the essential information required for students to receive full marks. As this is derived from the course information provided to the student, we can provide the relevant portion of the course content regarding the specific question, both with and without the model answer. The emergence of LLMs has made RAG a valuable technique for incorporating external knowledge from another data source into LLMs. This technique can be employed to provide subject knowledge to OpenAI GPT models for evaluating student answers. This objective examines the ability of this approach to improve the process of automatically assessing student answers.

RQ 2: Investigating the possibility of designing an improved prompts for all question categories.

Different types of questions expect different types of answers from the students. Sometimes the question expects the specific theory covered and sometimes the question expects an open answer. Answers to these two categories of questions are assessed using different methods. Instead of employing two separate prompts for assessing student answers for those question types, it is possible to consolidate them into a single prompt. Developing a single prompt that effectively assesses student answers for both sorts of questions minimizes unnecessary redundancies. This is investigated through this objective.

RQ 2: Examine how well GPT models can handle different subjects and types of questions in the marking context.

OpenAI introduces new GPT models as improvements to previous models. These models possess various amounts of parameters. Some models perform well in their ability to follow instructions. GPT-4 model demonstrates high performance across a diverse set of activities. The version accessible to the general public at no expense via chatGPT is GPT-3.5-Turbo. Therefore, different models will assess the student answers differently. This also relies on the number of dimensions, token limit, and training data. Their approach to conducting the student answer assessment procedure may vary depending on the subjects and types of questions. Therefore, this objective helps to determine the best-performing model in automated student answer assessment for various subjects and question types.

1.4 Justification for the Research

AES was introduced as a replacement for the human evaluation of student answers due to the inherent subjectivity and biases associated with the evaluation process. Subsequently, it underwent advancements using NLP techniques, Machine Learning, Neural networks, Pre trained transformer models such as BERT and LLMs. Previous research has demonstrated the impressive capabilities of pre-trained transformer models, specifically in the context of automated essay scoring. These achievements have been accomplished through the process of fine-tuning BERT models.

With the emergence of LLMs, several works have been done utilizing the capabilities of OpenAI GPT models. The potential of these models in text generation and logical reasoning has shown the potential for the AES process. Consequently, these have been employed for automated assessment of student answers. The primary elements included in AES include the Question, Student Answer, Model Answer, and a grading rubric serving as a marking guideline. Within some works, they have only utilized Questions, Student answers, and the grading rubric. The majority of research studies have mostly concentrated on assessing student answers written for language examinations designed to evaluate students' language proficiency.

Hallucinations pose a significant challenge in Language Model (LLM) research, particularly in scenarios where precise and contextually relevant responses are crucial, such as in Automated Essay Scoring (AES). While some studies have attempted to mitigate hallucinations by training smaller models and employing techniques like fine-tuning and few-shot learning, there remains a gap in addressing the incorporation of external knowledge. These approaches focus primarily on optimizing model performance within the existing pre-trained data framework but miss the potential benefits of integrating external information sources. Without considering external knowledge, LLMs may struggle to accurately comprehend and respond to inputs within specific domains, limiting their usefulness in tasks requiring complex understanding and contextual relevance. To address this gap, researchers need to explore methods that incorporate external knowledge into LLMs. By leveraging supplementary information from relevant sources such as textbooks, reference books, or lecture notes, LLMs can enhance their understanding of domain-specific content and mitigate the effects of hallucinations. This approach becomes particularly crucial in automated essay scoring, where student responses are expected to align closely with taught course content. By integrating external knowledge, LLMs can better discern the scope and context of student answers, leading to more accurate assessments and mitigating the impact of hallucinations on scoring accuracy. Overall, the integration of external knowledge represents a promising avenue for advancing LLM research and improving their performance in tasks requiring domain-specific understanding and contextual relevance.

However, GPT-4 exhibits a notable decrease in hallucinations compared to earlier GPT-3.5 models [7]. There is a lack of research conducted on the application of AES for theoretical subjects that require the subject knowledge to answer the questions. Therefore, this work aims to examine the procedure of AES for various question types and subjects by harnessing the capabilities of OpenAI GPT models. This research also incorporates the Retrieval Augmented Generation (RAG) approach for providing subject knowledge to these models. The purpose is to minimize hallucinations and facilitate learning without the need for training. In addition to that, feedback generation for each student answer and analyzing the effectiveness of prompt engineering techniques to enhance the AES is also included in this research.

1.5 Outline of the Dissertation

The significance of this research is thoroughly explained in this chapter. This chapter also addresses the research issue and its associated contexts. This chapter will discuss the scope and limitations of this study.

A comprehensive literature review is done and it is included in chapter 2. It discusses how Automated Essay Scoring has evolved through several phases of Natural Language Processing techniques. From the initial phase which was based on Bag of Words, Part-of-Speech tagging, and Semantic Similarity technique. Then later improvements were made utilizing machine learning, neural networks, and pre-trained models and they are described here. There is a huge improvement done by fine-tuning BERT for AES tasks. With the emergence of LLMs, they have shown great potential for AES. Then the literature review mainly focuses on the literature after the emergence of LLMs because this research specifically focuses on utilizing the capabilities of LLMs for automated student answer marking. Research works done regarding prompt engineering techniques and retrieval Augmented generation (RAG) are also explored since these areas will be integrated with the LLMs for this research.

Chapter 3 discusses the methodology of this research. It describes the dataset and gives a detailed view of the process of data collection and preprocessing, the experiment carried out, and the evaluation of the generated results.

In Chapter 4, comprehensive details of the final results and evaluation are presented and Chapter 5 demonstrates the conclusion of the research and outlines future work. The last chapter, Chapter 6 includes Appendices presenting the supplementary materials such as prompt versions, confusion matrices, hallucination exmaples, wrong assessment of students answers contain within the dataset, the chunking process of the course content, and grading rubrics utilized for this research.

1.6 Scope and Delimitations

Addressing the identified problem, the research involves assessing the student answers written for the Bachelor of Information Technology Degree Program of the University of Colombo School of Computing. The research seeks to provide insights into the acceptance, challenges, and potential benefits of automated assessment of student answers with the ultimate goal of informing the development of more efficient and effective assessment systems.

The student answers were scanned for two different subjects: IT 5105 Professional Issues in IT and IT 5306 Principles of Information Security. The research primarily focused on assessing the scanned student answers using OpenAI GPT models. This was done by utilizing a basic prompt template that was specifically created to analyze student answers. This prompt does not have prompt engineering techniques. The performance of the GPT models is evaluated using this prompt template, and the model that performs the best is selected to analyze a larger set of student answers across different subjects. The answers were assessed again utilizing prompt templates that were specifically designed using various prompt engineering techniques. This process helps to discover whether the student answer assessment process can be enhanced using the capabilities of OpenAI GPT models and then involving prompt engineering techniques. These prompt templates include the subject knowledge relevant to the particular question with the intention of analyzing the performance of these models for student answer assessment with the provision of subject knowledge. Additionally, feedback is also provided for each students' answer by using another prompt.

1.6.1 In scope

The following points are addressed in this study.

- Automated marking of short answers written for structured questions.
- Utilizing grading rubrics for the marking process of student answers.
- Providing detailed feedback using the rubric grading system.

1.6.2 Out of scope

The following points will not be addressed in this study.

- Automated assessment of Multiple Choice Questions.
- Automated assessment of answers with diagrams.
- Automated assessment of essays.
- Automated marking of coding-related question answers.

1.7 Summary

This chapter elucidates on research introduction. This chapter provides an explanation of the research introduction, which establishes the foundation for the fundamental understanding of the dissertation. The introductory chapter begins with the background which explains the motivation of the research and the need to enhance the automated marking process with the capabilities of OpenAI GPT models. Given the logical reasoning capabilities of OpenAI GPT models, it has been emphasized the need to utilize them in the AES domain.

The upcoming literature review section discusses that many studies have employed OpenAI GPT models to assess student answers by utilizing the Student Answer, grading rubric, and Model Answer. Through our research, we utilize the Retrieval Augmented Generation (RAG) technique to present the necessary subject knowledge or course content within the prompt, both with and without the Model Answer. The evaluation also includes an assessment of the performance of various models in automating the assessment of student answers for different subjects of questions. This chapter concludes with an explanation of the outline of the dissertation and follows up with a delimitation of scope. Further chapters explain the procedural aspects of the research and the evaluations conducted during the research.

Chapter 2

Literature Review

2.1 Overview

Researchers in the field of Automated Essay Scoring (AES) have investigated diverse strategies and techniques to efficiently and precisely analyze and score students' answers in the context of AES. In this literature review, we have gone through the related work in the AES domain under 4 sub-sections. First, we referred to the papers related to AES research work which were carried out before the rise of Large language models. The development of Automated Essay Scoring (AES) systems has progressed from rule-based algorithms like Project Essay Grader (PEG) in 1966 to more advanced methods that involve Natural Language Processing (NLP), Bayesian models, and deep learning techniques over more than fifty years.

In the second section, we have presented the work done in the AES context by utilizing the capabilities of large language models. Deep learning models, such as BERT and GPT, have had a significant impact on the development of Automated Essay Scoring (AES). Research has demonstrated that adjusting pre-trained language models like BERT for AES can enhance their effectiveness, leading to better performance compared to the most advanced models available. Moreover, GPT has emerged as a promising tool for AES, providing possibilities for accurate evaluation and feedback in educational contexts. GPT's integration has significantly changed the educational assessment, moving from prompt-based assessment to fine-tuned models. This transition promises more efficient and objective evaluation methods, leading to better accuracy.

In the next section, we have outlined the efforts made to improve AES tasks by employing prompt engineering techniques. In the initial studies, prompts were perceived as instructions to guide LLMs in predefined tasks. However, recent advancements have introduced prompt engineering frameworks, including pattern catalogs and task-specific prompting techniques, aimed at enhancing LLM performance across diverse applications such as Automated Essay Scoring.

In the final part, we discuss the work done related to use of RAG techniques. A study by Ovadia has evaluated knowledge injection via fine-tuning and Retrieval Augmented Generation (RAG), favoring RAG for its ability to integrate external knowledge. Another study has introduced In-Context Retrieval-Augmented Language Modeling (RALM), enhancing LLMs without altering their architecture. RAG proves crucial for question answering, and leveraging external knowledge sources effectively.

2.2 Automated Essay Scoring

The development of AES systems spans over half a century. In an early attempt at an automated marking context, the researchers used rule-based algorithms to assign marks [8]. The basic process was to match the keywords. Project Essay Grader(PEG)[1] started Automated Essay Scoring (AES) research in 1966. PEG graded the essay based on writing elements, including grammar, diction, composition, and others.

Then, the exam answers were evaluated using semantic analysis to produce an overall score for the student answers. Foltz, La- ham, and Landauer[9] created an Intelligent Essay Assessor (IEA) using the Latent Semantic Analysis (LSA). IEA was trained on domain-representative text, such as textbooks, samples of writings, or a large number of essays on a topic, and it has compared the semantic similarity of words and passages to assess essay quality, achieving accuracy comparable to human experts.

Later, the trend shifted towards using Natural Language Processing (NLP) (Natural Language Processing), which focused on the style and content of the answer to assign a score. Bayesian Essay Test Scoring System by Rudner and Liang[10] use Bayesian models for text classification in automated essay scoring. This study achieved an accuracy of over 80% with a small dataset of 462 essays. The results highlight the importance of feature selection and demonstrate the effectiveness of the Bayesian approach in achieving an accuracy of over 80% in essay scoring. So, with this, the rule-based scoring systems were refined with sentence structure-based automated grading. Machine learning models used supervised learning techniques, and the scoring process was conducted in 2 approaches: classification task and regression task. The regression tasks goal was to predict an essays score. The classification task was to classify the essays belonging (low, medium, or high) relevant to the questions topic [8]. A Ridge regression model was suggested by Sultan, Salazar, and Sumner[11], and among its features was Text Similarity, which measures how similar a students answer is to the reference answer. These text similarity features were Alignment, Semantic vector similarity, term weighting, length ratio, and question demoting.

With the alignment feature, they measured the proportion of content words in two sentences. Semantic Vector Similarity - This feature has used pre-existing word embeddings to compute a sentence-level semantic vector for each input sentence. The cosine similarity between the semantic vectors of the correct answer and the students answer was then used as a feature to measure the correctness of the response. Term weighting was distinguishing between domain-specific keywords and general content words to grade short answers accurately. The models accuracy of 0.887 indicates that short response scoring is highly accurate.

WordNet is a lexical dictionary-type database developed by Princeton University for English. Previous work[12] utilized this database and generated the WordNet graphs to compare the ideal answer given by the teacher and the student answer. The number of common nodes of both graphs (N) is calculated to compare the generated WordNet graphs. The score for the student's answer is calculated as follows:

$$\boxed{Marks = \frac{|N| \times \text{Total marks}}{\text{no of nodes within the WordNet graph of the final answer}}}$$

The performance of the proposed method is higher than that of the existing techniques. They have compared the Root Mean Square Error (RMSE) and the evaluation time of the proposed method with other available methods in the literature. Based on that comparison, the proposed method achieves better results.

Another work[13] has been done using supervised and unsupervised learning. As the unsupervised learning method, they have clustered the student answers using k-means clustering with k =3 with clusters named as excellent, mixed, and weak. The excellent cluster includes students' answers that scored full marks, and the weak cluster includes students' answers with two marks. However, the mixed cluster does not contain student answers with the same scores. Some student answers with full marks are also included in the mixed cluster. It has been found that there are some keywords similar to the model answer keywords contained in those answers. This shows that a better knowledge of acceptable vocabulary (synonyms) is needed to cluster more effectively.

As the supervised method, they predict the scores for student answers by calculating the hamming distance between the student answer and the model answer. For that, they extract the keywords from both the model answer and the student answer using the Bag of Words method. They compared the results of teachers and their model; the disagreement between them is minor for three questions. Only for one question the implemented model gives contradictory results. They found that their model performed well when the student answers contained the exact keywords in the model answer.

Later, developed automated marking systems have improved with deep learning techniques and syntactic and semantic features, and they have been showing improved results. With the success of deep learning, researchers started to utilize various neural networks to learn text representations. Taghipour and Ng[14] explored several neural networks, such as Long Short-Term Memory (LSTM) and CNN. Finally, they found that the ensemble model combining Long Short-Term Memory (LSTM) and CNN performs best. The study has used recurrent neural networks, precisely long short-term memory (LSTM) networks to learn the relation between an essay and its assigned score. The research results show that LSTM performs significantly better than all other systems and outperforms the baseline by a large margin (4.1%). However, basic Recurrent Neural Network (RNN) falls behind other models and does not perform as accurately as GRU or LSTM.

2.3 LLM Applications in AES

With the deep learning models, transformers have revolutionized the automated marking process. The language understanding of the pre-trained language models, for instance, Bidirectional Encoder Representations from Transformers (BERT)[15] and GPT[16], has enabled the assessment procedure to use context understanding and semantic meaning. BERT is typically used for natural language understanding tasks, whereas GPT is for natural language generation tasks.

A study by Ruosong and team[17] has proposed a method to improve the performance of Automated Essay Scoring (AES) by fine-tuning pre-trained language models(BERT) with multiple losses, including mean square error loss and batch-wise ListNet loss, resulting in better scores compared to state-of-the-art neural models. The paper highlights the importance of utilizing pre-trained language models and multiple failures to capture deep semantic information and enhance the accuracy of AES. Another study by Sung, Dhamecha, and Mukhi[18] used BERT for short answer grading. The proposed approach is evaluated on two datasets: the ScientsBank-3way dataset of SemEval-2013 and two psychology domain datasets. The paper reports up to a 10% absolute improvement in macro-average-F1 over state-of-the-art results on the benchmarking dataset[19]. According to the study results, the fine-tuned model yields classification almost up to the human agreement levels on the two psychology domain datasets. The fine-tuned model based on BERT establishes state-of-the-art results on the SciEntsBank dataset, outperforming previous approaches such as InferSent. GPT is a recently released transformer-based language model. As the demand for efficient and objective assessment methods grows, LLM models like GPT have entered the realm of automated essay scoring, revolutionizing the landscape of education and assessment. A documentary analysis by Mhlanga[20] shows that GPT has many educational uses now and in the future and can significantly impact education. Mizumoto and Eguchi[4] conducted a study on the potential use of prompt-based GPT for AES. They used OpenAIs text DaVinci- 003 model to automatically score 12,100 essays from the ETS Corpus of Non-Native Written English (TOEFL11). They compared the scores to benchmark levels, and their results showed the feasibility of using GPT in AES. Another research finding was that adding linguistic features improved the prediction of the benchmark level significantly, indicating that the combination of GPT + research-based linguistic features may produce the best result in predicting professional ratings. They have used Quadratic Weighted Kappa (QWK), a variant of Cohens Kappa, to assess the reliability of AES.

Student learning can be improved by providing feedback and student marks after the automated essay scoring. The AERA (Automated Explainable Student Response Assessment) LLM-based framework has been developed for such scenarios by distilling the knowledge of ChatGPT. AERA[2] is a fine-tuned small language model such as the T5 model trained using the rationales generated by ChatGPT. With the distillation, they have removed the inaccurate rationales of the dataset before training the small language model. With simple instructions, it has achieved an overall QWK of 55.54; with complex instructions, it has achieved an overall QWK of 61.23. This has been further improved by further refinement of the rationale Augmented.

A study by Kevin et al[21]. has focused on assessing how well GPT 3.5 and GPT 4 rate short essays written in a Second Language. They have used the CEFR (Common European Framework of Reference) rubric and a human-rated dataset of short essay responses collected as part of the Duolingo English Test, a high-stakes test of English for L2 learners. The dataset comprises 10,000 responses, focusing on diverse native languages and genders. Responses were classified using a simple CEFR estimator to ensure representation across CEFR levels. Human raters have evaluated 1,961 essays based on a rubric aligned with the CEFR scale, achieving a high inter-rater agreement of 0.87.

This studys methodology involved instructing GPT to rate essays provided within eight predefined categories, with calibration examples provided based on GPT's token limit. They have initiated a fresh GPT chat for each assessment to minimize potential interactions between essays. Inter-rater reliability (IRR) between GPT and human rater one was then calculated. Additionally, this study has compared GPT's performance with two baseline approaches: a machine learning (ML) classifier based solely on the character length of responses and a firm baseline representative of current Automated Writing Evaluation (AWE) methods, which typically utilize feature engineering and statistical modeling. The methodology was designed to evaluate GPT's effectiveness in rating essays and to benchmark its performance against existing methods commonly used in the field.

During the experiment stage, they conducted it in 3 ways. In the first evaluation, they evaluated the GPTs ability to rate essay responses on the CEFR scale when provided with only a minimal rubric and varying calibration examples. In that experiment, they showed that when no calibration is provided, neither GPT 3.5 nor GPT-4 even outperformed the baseline classifier using character length only. However, by providing just one calibration example for each rating category, GPT-4 has almost matched the performance of the AWE baseline. They have shown that providing additional examples did not result in significant improvement.

In the second experiment conducted with prompt engineering, they tested two strategies for improving the performance of GPT-4. First, they used a detailed grading rubric instead of a simple grading rubric, and second, they asked the GPT to provide a rationale before providing its rating to elicit a chain of thought (COT) reasoning. However, they concluded that GPT 4 required only one calibration example per rating category to achieve near-optimal performance, and other prompt engineering techniques they used were not very helpful. Also, they have mentioned that there is more space for future research to explore the other prompt engineering strategies for improving GPT performance.

A recent study[6] has investigated the effectiveness of LLMs GPT 4 and finetuned GPT 3.5 as AES tools. Further, in this research, they discovered that the feedback given by the LLM can improve the human raters' performance. They have fine-tuned the GPT 3.5 model with the annotated dataset. For the datasets, they have utilized a publicly available dataset and a private dataset. As the public dataset, they used the ASAP dataset, and their proprietary private dataset was a Chinese student English essay dataset comprising 6559 essays.

These researchers have used two methods for their experiments: prompt engineering and further fine-tuning with the training dataset. The study has explored different grading approaches, including zero-shot with and without the rubrics, few-shot with rubrics, fine-tuning, and baseline methods. The prompt engineering method used has involved developing initial instructions and refining them using GPT-4. For the prompt engineering strategies, they have used COT (Chain Of Thoughts) to enhance the capabilities of LLM.

In the few-shot approach, in addition to rubrics, they have included sample essays and their corresponding scores to assist the model in understanding scoring patterns. This sample selection was done in 2 ways; the first approach is randomly selecting the samples. The second approach has followed a retrieval-based approach that has proven effective in enhancing LLM performance. Moreover, they have followed a similar approach to what we have used for context inclusion in our prompting. They have used OpenAIs text-embedding-ada-002 model to calculate the embeddings and find the top k similar essays to include as sample essays. In conclusion, the author has stated that models trained via supervised methods exhibited the best performances. In addition, when provided with detailed information, such as rubrics and examples, the performance of GPT 4 has improved.

2.4 Prompt Engineering Techniques in AES

Large Language models (LLM) have recently transformed the field of natural language processing. They are applied to different research areas, such as Automated Essay Scoring, Automated Software Engineering Tasks, and Automated Writing Evaluation. As LLM and their applications have become the discussion in the field, identifying and implementing ways of conversing with LLM has been a significant focus area. This led to emergent concepts such as prompting, prompt engineering and prompt tuning.

A previous work [22] described prompts as instructions to guide the Large Language models in following predefined rules, completing tasks automatically, and generating outputs with predefined characteristics. They proceed with this definition to further identify prompts as a form of programming, and they can have patterns similar to software design patterns. They introduced a catalogue of prompt patterns that enthusiasts can follow in prompting and prompt engineering to solve common problems while conversing with models related to automated software engineering tasks. Further, this work introduces a framework to document the prompt patterns and depict how several prompt patterns can be combined to enhance the performance of LLMs in the considered context. As they emphasised, the quality of the output generated by these models solely relies on the quality of the prompts. The introduced framework is followed through the catalogue to document prompt patterns describing the specific intent for each pattern, categorising to types: input semantics, Output customization, Error identification, Prompt improvement, Interaction, and Context control, along with example prompts and their limitations, providing a comprehensive guide to solve commonly occurred problems when communicating with LLMs in any context.

LLMs employed in different research areas are customized for each context to improve performance and accuracy. The finetuning approach and prompt engineering are frequently used in customizing LLMs. A recent study [23] explores the capabilities of both approaches, employing them in three primary software engineering tasks: code generation, comment generation, and code translation. The research analyses three prompt engineering techniques against 18 fine-tuned LLMs and the state-of-the-art GPT-4. The performance of basic prompting, in-context learning, and task-specific prompting were quantitatively analyzed for the selected tasks. At the same time, a qualitative study was conducted with the participation of 27 graduate students and 10 industry practitioners. Well-known and widely used benchmark datasets CodeXGLUE, HumanEval, and MBPP were utilized in the study. The performance was evaluated using the pass@k metrics for code generation, BLEU Score for comment generation, BLEU ACC, and CodeBLUE for code translation. Accordingly, the findings emphasized that the state-of-the-art LLMs GPT-4 could not significantly outperform the finetuned models in code generation (NL to SC) tasks. However, the task-specific prompting technique has demonstrated the most noticeable results in improving model performance by 8.33% from the baseline and 6.99% from the basic prompting technique. The task-specific prompting technique provides additional prompts to guide the model in generating better results than the basic prompt.

Further, the findings depict that GPT-4 can considerably perform tasks such as comment generation (SC to NL). Throughout the qualitative study, it was observed that most participants were satisfied with the initial responses generated by GPT-4. At the same time, some used conversational prompts to improve the initial response and achieved 24.3% enhancement of evaluation from the initial response. Concluding the work, they highlighted at the time of the study that even the most used typical prompt engineering techniques, such as in-context learning, did not aid GPT-4 to demonstrate huge improvement in accurately performing the selected tasks as none of the approaches could dominate the others in any task.

The communication methods to generate the expected output are crucial when employing LLMs across research areas. Even though the state-of-the-art LLMs are trained on a large number of parameters that induce advanced language understanding and reasoning capabilities, the performance of the models can still be uncertain, depending on the contexts. Hence, exploring the prompt engineering techniques that can be utilized to communicate with LLMs successfully is an emerging research area. Another reasearch work [24] explore how the reasoning ability of large language models can be enhanced by employing a method inspired by two existing concepts in natural language processing. The study refers back to the state of the art of the first concept, in which either the models were trained from scratch, or pre-trained models were finetuned to enable arithmetic reasoning in the LLMs. As they described, the second method utilizes in-context learning, widely known as few-shot learning, as a prompt engineering technique to guide the models. They explore the potential of incorporating few-shot-learning prompt engineering techniques into complex reasoning tasks. The study introduces the chain of thought prompting technique, which utilizes the concept of the chain of thoughts concept in the prompts and combines it with few-shot learning techniques. Chain of thought refers to the ability of humans to break down complex problems into smaller components and solve them step by step. The new approach was tested against the well-known LLMs: GPT-3, PaLM, and LaMDA in arithmetic, commonsense, and symbolic reasoning benchmarks. They utilized the GSM8K and MAWPS benchmarks of math word problems, the SVAMP, ASDiv dataset and AQuA dataset of various math problems for the study. The baseline was set to the standard few-shot learning prompt. Results of the study led to key takeaways revealing that Chain of thought ability emerges as the model scales up. Hence, it does not positively perform in small models but drastically improves performance in models with above 100B parameters.

Further, it was observed that CoT prompting is mainly favorable in more complicated tasks that require complex reasoning. During the study, the scaling curves were dramatically elevated by the chain of thought prompting compared to standard prompting when employed in various reasoning tasks. Furthermore, the PaLM 540B parameters model created a new state of the art for arithmetic reasoning with CoT prompting. At the same time, GPT-3 and PaLM favorably reached state-of-the-art in overall reasoning tasks. They observed that the reasons generated by CoT prompts were logically and mathematically correct when manually inspecting 50 random samples and their outputs. In conclusion, the studys main finding is that chain-of-thought prompting outperforms standard prompting in arithmetic, commonsense, and symbolic reasoning tasks. This study significantly contributes to the field by introducing a prompt-only approach as an alternative to the need for extensive training datasets to finetune models to enable reasoning ability in them related to specific contexts.

2.5 Retrieval Augmented Generation (RAG)

To address the issues of static and non-specific knowledge in Large Language Models (LLMs), regular updates to the model's training data should be implemented, incorporating recent information and domain-specific datasets. However, more than general pre-training is required for knowledge-intensive tasks; a post-processing step called knowledge injection is required. This step enhances the model's expertise in specific domains by integrating additional domain-specific knowledge into its responses, further improving its accuracy and relevance across various contexts.

Oded Ovadia [25] has worked on evaluating the knowledge injection capabilities of LLM by comparing Finetuning and RAG, which are two widely used techniques for knowledge injection. Fine-tuning involves adjusting a pre-trained model to perform better on specific tasks, while RAG expands the model's abilities by integrating external knowledge sources. In this study, they included tasks from different fields to ensure that the evaluation of LLM was not limited to specific areas of knowledge. They have compiled a dataset from Wikipedia articles to evaluate the LLM abilities, focusing on clean, manageable chunks of information. A new set of multiple-choice questions was also generated to accurately evaluate the models performance. The researchers used the LM-Evaluation-Harness repository to assess LLMs' performance in their experimental phase. They have evaluated three modelsLlama2-7B, Mistral-7B, and Orca2-7Balong with other baseline models and variations. Finally, their results showed that RAG consistently outperformed both base models and fine-tuning alone. It also has shown promising results in tasks related to current events, demonstrating its ability to incorporate real-time knowledge effectively.

In conclusion, the study highlights the importance of injecting knowledge into LLMs for better performance on knowledge-heavy tasks. By comparing fine-tuning and RAG, researchers found RAG to be a more reliable method. Its ability to utilize external knowledge sources through document retrieval significantly improved the models' understanding and manipulation of factual information.

A study by [26] proposed an In-Context Retrieval-Augmented Language Modeling(RALM) that does not need to modify the existing LM architecture like in other RALM approaches. In-context retrieval-augmented Language Modeling (RALM) adds relevant documents to the input of a language model without changing the model itself. Overall, this approach can make language models more accurate and useful when the model cannot be modified directly.

RAG is a large language model that refers to knowledge retrieved from an outside knowledge base before generating a response. [27] Another study analyzes the importance of RAG for question answering. To refer to an external knowledge base, they used the Pyserini index. GPT-3.5-turbo has been used as the large language model for the seven experiments done within this paper. Mainly, they evaluated the performance of the baseline experiment, which was prompted with each question in the training dataset exactly once. The other two main experiments were ARM-RAG and Obfuscated ARM-RAG, which involved providing correct examples within the prompt as question, answer, and reasoning as hinting with answers. The difference between these two is the examples because ARM-RAG used typical examples and Obfuscated ARM-RAG used examples that were changed for the nouns to be replaced with nonsensical words and proper names with rare names. This replacement has improved the accuracy of the model for answering questions. The main finding in this paper is that we can improve the performance of question answering with RAG by improving the quality of retrieved examples.

2.6 Conclusion

Automated Student Answer Assessment has evolved through several stages from traditional Natural Language Processing techniques. Bag of Words, Part-of-Speech tagging, and Semantic Similarity techniques were used at the initial stage. With the involvement of machine learning and deep learning, Automated Essay Scoring (AES) has been improved with classification models, regression models, text mining using WordNet graphs, etc. With the emergence of pre-trained language models such as BERT, the quality of AES was vastly increased by fine-tuning the BERT model. Then, the researcher found an interest in exploring the potential of large language models such as GPT-3 and GPT-4 in AES due to their exceptional performance in generating human-like responses and reasoning capabilities.

The existing literature mainly focuses only on the AES rather than focusing on a specific area of knowledge such as science, mathematics, etc. Applying student answer assessment to a specific area with model answers or key points to limit the potential area of knowledge needed to answer a question will improve the applicability and relevance. Existing literature has only utilized the model answer or the answer given by the teacher to grant a score to a students answer using large language models. There is a potential research area to apply particular course content with a wide range of acceptable responses and improve the student answer assessment quality instead of the model answer, which limits the knowledge area of a model. This concept can be applied to the automated student answer assessment domain using the Retrieval Augmented Generation with large language models.

Chapter 3

Methodology

Our research methodology adopted the Design Science approach which started with conducting the literature review and identifying the research gap. The research problem identification phase followed this phase. Incorporating the RAG technique to mark student answers is a major part of the identified research area. As the initiation step of the research implementation, an initial artifact was developed based on a basic prompt to assess the student answers utilizing the OpenAI GPT models along with only question and student answers. Subsequently, a pilot study was conducted utilizing the created artifact to determine the most effective GPT model. Once the best-performing model was identified, the artifact was enhanced to evaluate student answers using the same model. This was done by implementing five distinct ways outlined in section 3.4.2. With this, LLMs' capability to mark different question types is also evaluated. Finally, the student answer assessment process is performed for different subjects for all approaches to ensure LLMs can handle the marking process consistently for different subjects. Subsequently, the artifact undergoes further modification through the incorporation of prompt engineering techniques into the basic prompt. As a result, the prompt evolved into three prompt versions. These updated artifacts are then subjected to testing to evaluate the impact of prompt engineering techniques on the process of automated student answer assessment.

3.1 Design Overview

Retrieval Augmented Generation (RAG) is a technique that combines the power of information retrieval with text generation models like GPT. RAG can be considered as another approach to in-context learning, which is used to provide knowledge to the LLMs without training them specifically to use that knowledge. The latest information and specific knowledge can be provided to the LLMs, and they can generate context-aware responses without relying solely on their training data. Here's how it works in our research context: Retrieval: When given a question, instead of relying solely on the text generation model (LLM) to generate a response from scratch based on their training data, the system first searches the database for relevant information based on the input and retrieves them. (We can have a database of external knowledge we need to feed to LLM).

Augmented: Once the relevant information is retrieved, it's collaboratively fed into the text generation model as additional context or input. This model then synergistically uses the retrieved information along with the original prompt to generate a response.

Generation: The text generation model then produces a coherent and contextually relevant output based on the combined input of the original prompt and the retrieved information.

3.1.1 Our Approach

This section will explain the methodology utilized in the experimental phase of this research. The provided diagram illustrates the fundamental framework of the study design.

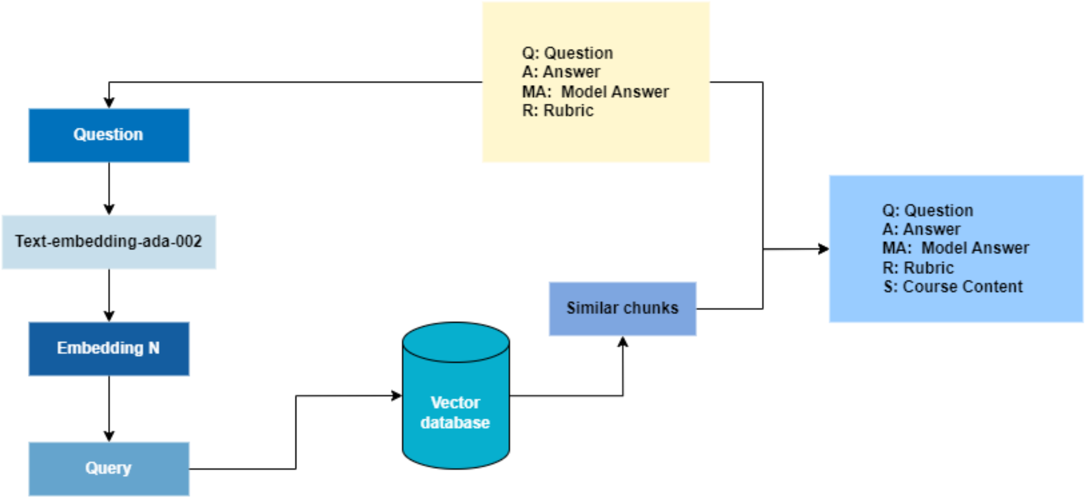


Figure 3.1: Fundamental approach of marking with course content and model answer

The proposed method for evaluating student answers using OpenAI GPT models by giving the course content chunks are, as shown in Figure 3.1

Preparing the vector store

Due to token constraints imposed by OpenAI GPT models, providing the entire course content in the prompt is not feasible. Hence, the suggested approach for transmitting the course information is to provide only the relevant course material about the specific exam question. In the first phase, we gathered the entire course content, which includes the reference books and lecture notes of the courses being considered, and divided them into smaller chunks. Chunking involves systematically dividing course content into smaller segments according to a predetermined criterion. The chunk size was determined logically by separating them into sub-topics based on the predetermined criteria. We had to consider the token limits, where the token count includes both the prompt and the response. Following the initial chunking procedure, a suitable number of tokens within a chunk was determined because certain chunks were excessively huge to be transmitted through the prompt. The chunking process was repeated following a manual chunking criteria according to preserve the logical structure of the information. Hence, decreasing the number of tokens transmitted via the prompt is necessary while staying below the prescribed token limit. The approach we used to store these chunks in the vector database is illustrated in Figure 3.2.

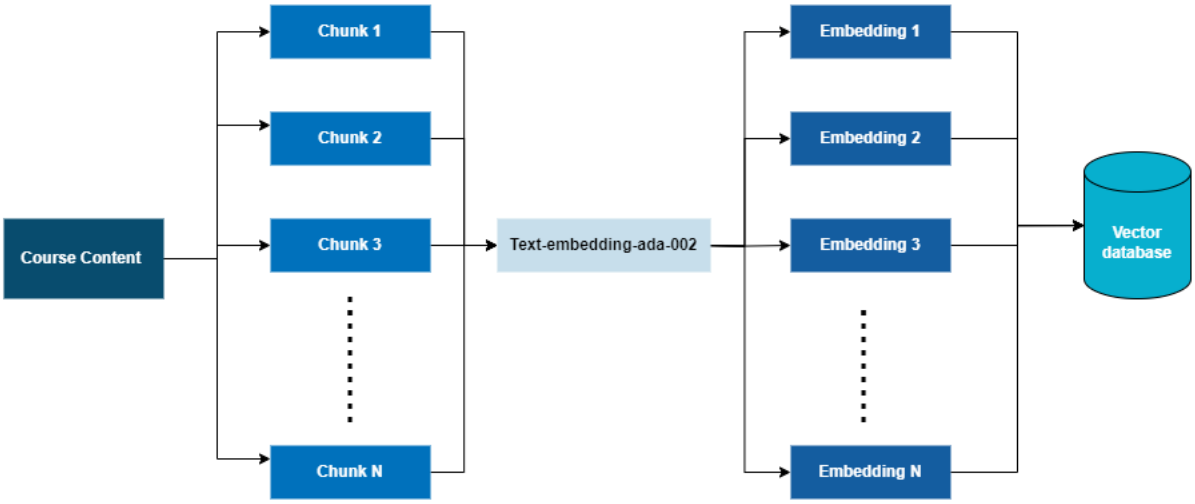


Figure 3.2: Process of creating the Vector Database

After dividing the course content as previously stated, the subsequent procedure was executed to determine and obtain the pertinent segment for a specific query rather than transmitting the entire course content in the prompt. We computed the embedding for each segment using OpenAI's ada-002 model for text embedding. These text embeddings are a depiction of words formed in a vector space with a large number of dimensions. Hence, these text embeddings are vectors that possess both magnitude and direction. Once the text embeddings for each chunk were generated, we utilized the ChromaDB vector database to store these embedding values and the primary subject and subtopic associated with each chunk. The research used the Chroma, an open-source database specifically designed for AI applications, as the vector database. We could determine the top k similar chunks by utilizing cosine similarity. Cosine similarity is a method used to quantify the similarity between two items in high-dimensional spaces. The procedure for selecting chunks will be discussed in the following section.

Prompting the LLM

As depicted in the initial illustration, we utilized the vector database to store the textual embeddings of the chunks. Next, we created the text embeddings of the question that we would evaluate. Using that embedding, we query the vector database and obtain the chunks with the highest similarity to that specific topic. Therefore, these chunks contain the relevant course material for that specific issue. Due to the constraint on the token limit for a single request, we must choose the number of chunks we will get. Previous experiments in AES tasks have shown that increasing the value of k does not consistently lead to improved performance. By including extra chunks, the prompt will also provide additional data from the course content that is irrelevant to answering the question. The approach employed for evaluating student answers recognizes the significance of important content but sometimes produces hallucinations. To minimize this, choosing an optimal quantity of chunks to be transmitted is necessary.

Consequently, we selected an appropriate value for k, specifically k=3, and obtained the three most similar chunks for a particular marking instance. After dividing the course content as previously stated, the subsequent procedure was executed to determine and obtain the pertinent segment for a specific query rather than transmitting the entire course content in the prompt. We computed the embedding for each segment using OpenAI's ada-002 model for text embedding. These text embeddings are a depiction of words formed in a vector space with a large number of dimensions. Hence, these text embeddings are vectors that possess both magnitude and direction. Once the text embeddings for each chunk were generated, we utilized the ChromaDB vector database to store these embedding values and the primary subject and subtopic associated with each chunk. The research used the Chroma, an open-source database specifically designed for AI applications, as the vector database. We could determine the top k similar chunks by utilizing cosine similarity. Cosine similarity is a method used to quantify the similarity between two items in high-dimensional spaces. The procedure for selecting chunks will be discussed in the following section.

Prompting is the method of interacting with the LLMs. Prompts comprise the required duties for LLMs and the corresponding directions to be followed. Once the necessary portions were extracted from the vector database, they were incorporated into the prompt, specifically created to evaluate the student's answer. The prompt includes retrieved chunks, questions, student answers, grading rubrics, and the essential procedures for doing the student answer grading procedure. The

GPT-4 model was instructed to evaluate the student's answer based on the prompt. We produced scores for the students' answers and explained each score during the assessment.

3.2 Data collection and preprocessing

3.2.1 Data Source

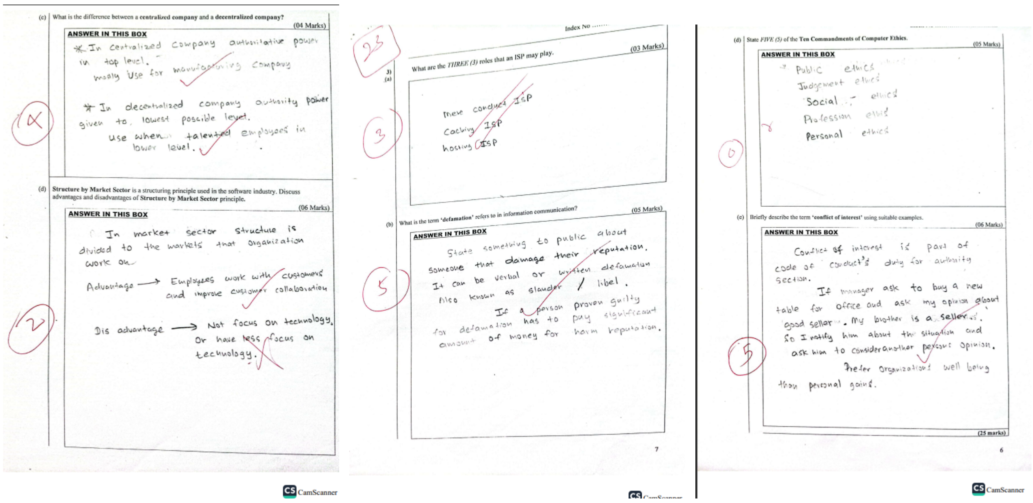
Since this research focuses on evaluating student answers written for a subject or a module which have separate course content, we used a private dataset that consists of student answers for structured essay questions from the University Of Colombo School Of Computing Degree Of Bachelor Of Information Technology (External) Academic Year 2021 and 2022 3rd Year Examination Semester 5 written by exam takers in IT 5105 Professional Issues in IT and IT 5306 Principles of Information Security subjects. Course content for each module was also used to create the chunks used during the automated student answer assessment process.

3.2.2 Data Collection

The data for this research was collected from the University of Colombo School of Computing. When collecting the data, first, we had to request consent from the IUD board of the University of Colombo School of Computing since these papers are standard exam papers written by the students manually during their university examinations. Exam papers, model answers, and student answers within the previous 10 years were requested with the aforementioned request. After permission was granted, suitable subjects or modules were selected by analyzing the nature of the questions and course content. The dataset consisted of the student answers marked by the university lecturers.

3.2.3 Anonymisation

It was stated that the student answers should only be collected without student index numbers as permission was granted to obtain student answers written for the past 10 years of BIT exams. Therefore, the student submissions had to be anonymized before being exposed to the OpenAI API. During the scanning process of student answers, we removed the student index numbers while scanning the student answers. This process has confirmed the anonymization of the students.



(c) What is the difference between a centralized company and a decentralized company?

(04 Marks)

ANSWER IN THIS BOX

* In centralized company authoritative power in top level.
mostly use for manufacturing company

* In decentralized company authority power given to, lowest possible level.
use when talented employees in lower level.

1

(d) Structure by Market Sector is a structuring principle used in the software industry. Discuss advantages and disadvantages of Structure by Market Sector principle.

(06 Marks)

ANSWER IN THIS BOX

In market sector structure is divided to the markets that organization work on

Advantage → Employees work with customers and improve customer collaboration

Dis advantage → Not focus on technology, Or have less focus on technology.

2

CamScanner

Index No ....

23

3) (a) What are the THREE (3) roles that an ISP may play.

(03 Marks)

mere conduct ISP
Caching ISP
hosting ISP

3

(b) What is the term 'defamation' refers to in information communication?

(05 Marks)

ANSWER IN THIS BOX

State something to public about someone that damage their reputation.
It can be verbal or written defamation
Also known as slander / libel.
If a person proven guilty for defamation has to pay significant amount of money for harm reputation.

5

7

CamScanner

(d) State FIVE (5) of the Ten Commandments of Computer Ethics.

(05 Marks)

ANSWER IN THIS BOX

* Public ethics
Judgement ethics
Social ethics
Profession ethics
Personal ethics

0

(e) Briefly describe the term 'conflict of interest' using suitable examples.

(06 Marks)

ANSWER IN THIS BOX

Conflict of interest is part of code of conduct's duty for authority section.
If manager ask to buy a new table for office and ask my opinion about good seller. My brother is a seller. So I notify him about the situation and ask him to consider another persons opinion.
Prefer organization's well being than personal gains.

5

(25 marks)

6

CamScanner

Figure 3.3: Scanned exam papers with student answers and lecturer's assigned marks.

3.2.4 Data Transformation

After getting the IUD approval, we received the students' paper bundles and started scanning the papers. We used a commercial scanning tool (Camscanner) for the scanning process. This is the initial phase of the scanning process. After selecting the question papers by carefully considering the course content of the modules, we selected 6 questions from the Professional Issues in IT and 4 questions from the Information Security subjects that we considered for our research. Here, we considered 100 student answers for each question. So overall, we had to convert 1000 data points.

These questions are selected by considering the student's answers provided level and the question type. How theories taught with the course content should be applied to answer the questions was also considered because this research focuses on assessing student answers written for theoretical examinations. Some students still need to complete their answers since this was an original University examination. Here, we had to avoid the questions that had diagrams. Coding questions were also avoided since there were not enough written answers to evaluate the answers. To gather additional findings on how these OpenAI GPT models assess student answers for different question types, questions selected from the Professional Issues in IT were divided into two types. Some questions expect an exact answer from a student, and some expect an open answer. This research considers these two scenarios when selecting the question types, calling them open-ended questions and questions that expect precise answers.

To assess them using the OpenAI GPT models, scanned answers should be converted into text format. Therefore, the previously selected 6 questions from professional Issues in IT and 4 questions from Information Security and the relevant answers are converted to text format using Google Lens which has an accuracy of 92.6%. After converting the scanned answers into the text format, we conducted a manual inspection to correct the transformation errors.

3.2.5 Data Cleaning

The data cleaning included fixing errors, language translations, clear formatting, and spelling issues. After converting physical documents into digital formats that are editable and searchable, we had to go through each students answer and the teachers mark to verify there were no errors in the dataset. At the same time, we had to mark the students answers with the grading rubric when we saw that the teacher had made some mistakes or given some unfair marks that did not align with the grading rubric. These are the drawbacks of the traditional marking process due to human error. We had to ignore some students answers because their handwriting was unrecognizable. Since there were more than 100 scanned papers, 100 student answers from each question were selected, considering the student answers with unclear handwriting as extra. Sometimes, spelling issues in the students answers may lead to a wrong evaluation of the students answers. Hence, data cleaning is a necessary process that should be followed during this research. Then, we created 2 CSV files separately for each subject or module containing student answers for all the selected questions. These files also included the score given after manually marking the students answers. Scanned questions were also added to 2 separate CSV files based on the subject and the model answer, as well as a grading rubric.

3.3 Implementation - Python Jupiter file

Following file was utilized to assess student answers for a particular question.

3.3.1 Loading Packages

```
import os
import numpy as np
import pandas as pd
import json
from sklearn.metrics import accuracy_score, precision_score , recall_score , f1_score , cohen_kappa_score
from dotenv import load_dotenv
from openai import OpenAI
import chromadb

load_dotenv()
os.environ['OPENAI_API_KEY'] = ”OPENAI−API−KEY”
client = OpenAI(api_key=os.environ.get(”OPENAI_API_KEY”))
```

3.3.2 Parameters

```
question_number = 4
answers_set_size = 100
#4b−1, 4a−2, 3a−3, 1c−4, 1a2021−5, 1c2021−6
```

3.3.3 Loading question data

```
def extract_question_data(row_number):
    df = pd.read_csv(”questions.csv”)
    question = df.iloc [row_number − 1, 0]
    model_answer = df.iloc[row_number − 1, 1]
    rubric = df.iloc [row_number − 1, 2]
    return question, model_answer, rubric

question, model_answer, rubric = extract_question_data(question_number)
```

3.3.4 Loading student answers and their scores

```
start = (question_number - 1) * 100
end = (question_number - 1) * 100 + answers_set_size

def extract_answers_and_scores(start, end):
    df = pd.read_csv("answers.csv")

    selected_rows = df.iloc[start:end]

    answers = np.array(selected_rows.iloc[:, 0])
    scores = np.array(selected_rows.iloc[:, 1])

    return answers, scores

student_answers, actual_scores = extract_answers_and_scores(start, end)
```

3.3.5 Load Vector Database

```
dbclient = chromadb.PersistentClient(path="./chromaDBs")
collection = dbclient.get_collection("CourseContents")
```

3.3.6 Embed the question

```
embeddings = client.embeddings.create(
  model="text-embedding-ada-002",
  input=question
)
```

3.3.7 Create the query embedding

```
query = embeddings.data[0].embedding
```

3.3.8 Query the vector database

```
context = collection.query(
    query_embeddings=[query],
    n_results=3
)
```

3.3.9 Arrange the course content to a string

```
subject_knowledge = ""
for x in range(len(context['documents'][0])):
    subject_knowledge = subject_knowledge + context['documents'][0][x] + "\n\n"
```

3.3.10 Prompt

```
prompt="""
Student answer is given to the question below.
————————————————————————
{Question}
————————————————————————
We have also provided some context information below.
————————————————————————
{Context}
————————————————————————
Set of rubrics to follow as the guideline is provided below.
————————————————————————
{Rubrics}
————————————————————————
Given the context information and rubric, assign a score to the student answer given below.
Follow the rubrics when assigning the score providing the steps taken.
Do not use your prior knowledge.
Follow JSON format to give the explanation first and then the score.
————————————————————————
{Student_Answer}
"""
```

3.3.11 Assessing student answers and predicting score

```
predictions = []
y=0
for student_answer in student_answers:
    completion = client.chat.completions.create(
        model=”gpt−4−0125−preview”,
        temperature=0,
        max_tokens=300,
        messages=[
            {”role”: ”system”, ”content”: ”As a teacher, your responsibility is to give a score to the
            students' answers based on the information provided.”},
            {”role”: ”user”, ”content”: prompt.format(Question=question, Context=subject_knowledge,
            Rubrics=rubric, Student_Answer=student_answer)}
        ]
    )

    response = completion.choices[0].message.content
    subResponse = response[7:len(response)−3]
    json_object = json.loads(subResponse)
    gptscore=int(json_object[”score”])
    Results = [student_answer,actual_scores[y],gptscore,subResponse]
    predictions.append(Results)
    print(y)
    y=y+1

df1=pd.DataFrame(predictions,columns=['Answer','Real_Score','GPT_Score','Final_Response'])
df1

df1.to_csv('./results.csv')
```

3.3.12 Calculate QWK and Accuracy

```
df = pd.read_csv('./results.csv')
actual_scores=df.Real_Score.tolist()
predicted_scores=df.GPT_Score.tolist()

# Calculate simple accuracy
accuracy = accuracy_score(actual_scores, predicted_scores)

# Calculate Quadratic Weighted Kappa
qwk_score = cohen_kappa_score(actual_scores, predicted_scores, weights='quadratic')

print(”Quadratic Weighted Kappa Score:”, qwk_score)
print(”Accuracy:”, accuracy)
```

3.4 Pilot Test

We did a pilot test to determine the most effective GPT model for marking student answers. Choosing an efficient model at the beginning helps to narrow down the focus of the research instead of evaluating all student answers using multiple accessible LLMs. To assess the efficacy of several models, we employed GPT-4, GPT-3.5-Turbo, and GPT-3.5-Turbo-Instruct. We employed a basic prompt consisting solely of the question and corresponding student answers. We employed a dataset consisting of 50 student answers for each topic using random sampling. When choosing the questions, we considered two types of questions: one open-ended structured question and one precise structured question. This was done to evaluate the capabilities of the model thoroughly. After obtaining the findings, we used the QWK evaluation matrix to assess the level of agreement between the scores given by the teacher and those generated by the GPT models. Upon analyzing and comparing the QWK findings, it was concluded that the GPT-4 model demonstrated the highest level of performance among the examined models. The results table that was generated can be found in section 5.1. We continued our investigation using the GPT 4 model based on that finding.

3.5 Assessing with GPT 4

3.5.1 Generating Results

Based on the data obtained from the pilot test, the GPT-4 model outperformed the other two models. Therefore, for our research, we utilized the GPT-4 API provided by OPEN AI as a paid service, with charges dependent on token consumption. GPT is a Language Model that generates responses to user inputs using conversation completions based on various types of users. GPT functions as a dialogue with three distinct roles: user, assistant, and system. Upon receiving a message from the user, the assistant generates a response, and the system can guide the assistant's response. This study involves the assistant evaluating the student's answer while the system verifies that the response adheres to a specific format. This study employed a temperature value of 0 and a maximum token length of 300 as parameter values to produce answers. The maximum token length parameter restricts the number of tokens produced and incorporated in the answer. Five fundamental prompts were created to facilitate communication with the model, encompassing the essential procedures for marking student answers. These prompts were explicitly crafted to assess the student's answer by analyzing their approach, followed by a human evaluation of the student's reply. The methodologies and prompts used in this research are explained in the following sections.

3.5.2 Marking Approaches

Student responses can be evaluated by considering multiple elements, including the questions asked, the student's answer, the model answer, the grading rubric, and the subject knowledge gained by the students. During our review of the available literature, we discovered that two marking procedures had been utilized to evaluate student answers using various combinations of the aforementioned components. Building upon that, in this study, we have examined 5 different grading methods, which include:

1. **Providing only the question and student answers in the prompt template (Q+A)**

In this setting, we provide the Question and the Students' answer in the prompt with the necessary instructions to assign a grade to the students' answer in a particular mark range. The model then evaluates the student's answers and assigns a score based on their comprehension within the specified score range, along with an explanation of the steps the model has taken to assign that particular score to the student's answers. This is the most straightforward approach which was followed in this research.

2. Providing the question, student answers, and grading rubric in the prompt template (Q+A+R)

Alongside the prompt, the Question, and the student answer, we also provide GPT-4 with explicit grading rubrics to improve the first approach. These rubrics serve as guidelines for evaluating the student's answer, outlining specific criteria such as adherence to instructions. By incorporating these rubrics, the evaluation process becomes more structured, transparent, and consistent, ensuring that the assessment aligns with predefined criteria.

3. Providing the question, student answers, grading rubric, and course content as context information in the prompt template (Q+A+R+C)

This prompt includes the subject knowledge extracted from the course syllabus, the question, the student answer, and the grading rubric. As subject knowledge, the relevant course content chunk for the particular question will be retrieved from the previously created vector database by querying the database with the question. This assists GPT-4 in giving the external knowledge to understand the scope of the knowledge it has to consider. Therefore, this approach is based on the RAG technique.

4. Providing the question, student answers, grading rubric, and model answer in the prompt template (Q+A+R+MA)

This prompt includes the Model Answer, which the teachers created to assist in the marking process. This assists GPT-4 in giving the idea of a perfect answer. This also helps the evaluator to understand the depth of the answer expected by the question. Student answers will be compared with the model answer using this approach.

5. Providing the question, student answers, grading rubric, model answer, and course content as context information in the prompt template (Q+A+R+MA+C)

In this final approach, we evaluated the model performance when we prompted LLM with the question, student answer, and grading rubric with both the model answer and the context information. With this approach, the performance and the behavior of the GPT-4 model are analyzed when it is provided with both the model answer and the subject knowledge along with the question, student answer, and grading rubric compared to previous approaches.

3.5.3 Prompt Levels

We employed many prompt levels, starting with fundamental prompts that provided basic instructions. Subsequently, we utilized prompts enhanced using prompt engineering techniques to stimulate the GPT model. We employed the basic prompts for each of the five ways listed above to obtain the desired outcomes.

Basic level prompts

In the initial methodological approach, without including a grading rubric or any external information, we direct the Large Language Model to evaluate students' answers based solely on their pre-existing knowledge. Due to the possibility of models assigning scores outside of the acceptable score range, the prompt included the appropriate score range for each question. This method is the most cost-effective and straightforward technique to stimulate a language model, as it does not require the user to provide specific instructions. Given this question, the input token count and token generation rate are minimal, making it the most cost-effective prompt in this research. The following prompt depicts the basic prompt we utilized.

```
Student answer is given to the question below.
——————————————————————
{Question}
——————————————————————
Assign a score between 0 to 4 to the student answer given below.
Follow JSON format to give the explanation first and then the score.
——————————————————————
{Student_Answer}
```

The second approach utilized a prompt constructed by including the grading rubric instructions from the previously employed prompt. In this methodology, the original prompt was altered by incorporating additional directives to evaluate the student's answer in accordance with the predetermined standards outlined in the grading rubric. This prompt does not include the score range for the student's answers. It will be provided separately, with the grading rubric, instructions, and prerequisites for assigning a specific score to the student's answer. The utilized grading rubrics are available in Appendix G. Below, we have presented the prompt that was used for this second approach.

```
Student answer is given to the question below.
————————————————————————
{Question}
————————————————————————
Set of rubrics to follow as the guideline is provided below.
————————————————————————
{Rubrics}
————————————————————————
Given the rubric, assign a score to the student answer given below. Follow the rubrics
when assigning the score providing the steps taken.
Follow JSON format to give the explanation first and then the score.
————————————————————————
{Student_Answer}
```

As the third approach, we integrated the prompt with the subject knowledge by incorporating the RAG technique. We requested the LLM to evaluate the student's answer by incorporating the relevant subject material provided. The course content chunk retrieved from the vector database provides subject knowledge. Subject knowledge can be regarded as a form of contextual learning. By doing this, we can modify the fixed information of the LLM and incorporate external knowledge that is pertinent to the question. The "Do not use your prior knowledge" statement in the prompt prevents the model from relying on its own knowledge, as this could lead to the generation of hallucinations. The prompt used for this third approach is shown below.

```
Student answer is given to the question below.
——————————————————————
{Question}
——————————————————————
We have also provided some context information below.
——————————————————————
{Context}
——————————————————————
Set of rubrics to follow as the guideline is provided below.
——————————————————————
{Rubrics}
——————————————————————
Given the context information and rubric, assign a score to the student answer given below.
Follow the rubrics when assigning the score providing the steps taken. Do not use your
prior knowledge.
Follow JSON format to give the explanation first and then the score.
——————————————————————
{Student_Answer}
```

In the following method, we prompted the LLM with the question, student answer, grading rubric, and Model Answer, which is utilized in the manual marking procedure. The LLM was directed to evaluate the model answer as an exemplary illustration of how the student's answer should be to achieve the maximum score. Hence, the following prompt enables the GPT-4 model to assess the student's answer by comparing it to the given model answer and adhering to the grading criteria.

Student answer is given to the question below.

————————————————————————

{Question}

————————————————————————

An answer that can score full marks is given below as the model answer .

————————————————————————

{Model_Answer}

————————————————————————

Set of rubrics to follow as the guideline is provided below.

————————————————————————

{Rubrics}

————————————————————————

Assign a score to the student answer given below. Compare the student answer with the "model answer" and follow the rubrics when assigning the score providing the steps taken.
Do not use your prior knowledge.
Follow JSON format to give the explanation first and then the score.

————————————————————————

{Student_Answer}

As the final approach, we employed the following prompt with subject knowledge and the model answer and asked to assign a grade to the given student answer.

```
We have provided a model answer below.
———————————————————————
{Model_answer}
———————————————————————
We have also provided some context information below.
———————————————————————
{Context}
———————————————————————
Set of Rubrics is below.
———————————————————————
{Rubrics}
———————————————————————
A student has provided an answer for the following question.
———————————————————————
{Question}
———————————————————————
Given the context information, model answer and not prior knowledge, Assess the student
answer given below explaining the how the score is formed before giving the total score.
Follow the Rubric provided for this task.
Follow JSON format to give the explanation first and then the score.
———————————————————————
{Student_answer}
———————————————————————
```

3.5.4 Prompt with Techniques

Various prompt engineering strategies can be employed during the creation of prompts. According to Open AI and many researchers, we incorporated their findings into our question. We generated multiple variations of the prompt and obtained the results. The prompt yielded the most favorable outcomes when incorporating engineering techniques. Additional variations of the prompt can be found in Appendix A.

```
As an evaluator of a university examination with expertise in professional practice in
information technology, your role is to analyze the student's answer to the given
 professional practice—related question and grade the
student's answer according to a predetermined set of rubrics. Let's think step by step.

Here are the specific guidelines for the scoring process:

Context information:
{Context}

Rubrics:
{Rubrics}

A student has provided an answer to the following question.
{Question}

Student's answer to evaluate:
{Student_Answer}

The following tasks should be performed to score a student's answer.

Task breakdown
1. Carefully read the provided question.
2. Read the student's answer and memorize it
3. Compare the student's answer with the provided context information.
4. Identify specific elements in the student's answer that align with the context information.
5. Without giving the score yet, let's think step by step and explain the scoring process
in steps referring to the provided rubrics.
6. Next, provide the score and the reason for the score.
7. Follow the JSON format with the following keys: explanation, score.
```

Three prompting techniques were utilized within this prompt and they can be explained as follows.

Chain-Of-Thought (CoT)

LLMs sometimes find it difficult to solve complex problems. This technique instructs the LLMs to think step by step to solve complex problems with natural language instructions. With this technique, the complex reasoning capabilities of LLMs can be enabled. Previous work utilizing this prompting technique shows that LLMs can be used to solve mathematical problems. This can be utilized by adding a Lets think step by step statement to a basic prompt.[24]

Persona Pattern

The Persona Pattern is a prompting technique designed to guide the output of LLMs by assigning them specific perspectives or roles, known as ”personas.” This approach serves two primary purposes: clarifying the intent and context of the generated output and assisting users in expressing their needs without detailed knowledge of the desired outputs. [22]

Similar to this, we have added to the basic prompt in this research the instruction that the LLM has to take the role of an evaluator of a university examination with expertise in professional practice in information technology. This persona directs the LLM to analyze student answers related to professional practice questions and grade them according to predetermined rubrics. By leveraging personas, users can effectively guide LLM outputs to align with their desired perspectives and objectives, even without precise knowledge of the necessary details.

Task Breakdown

In the realm of Automated Essay Scoring (AES), one effective technique employed is the task breakdown approach [28]. This systematic classification breaks tasks into smaller steps, ensuring the model goes through every step you want to follow. The primary objective is to minimize error rates by focusing on discrete components of the task in each prompt, thereby reducing the computational costs associated with larger prompts.

In our prompt, the task of scoring a student's essay:

1. Carefully analyze the provided question.
2. Extract and retain the student's answer.
3. Compare the response to contextual information.
4. Identify pertinent elements within the response that correlate with the context.
5. Provide a step-by-step breakdown of the scoring process utilizing predefined rubrics.
6. Assign a score accompanied by a detailed rationale.
7. Format the output using JSON, incorporating keys for explanation and score.

This breakdown exemplifies how tasks in AES can be systematically segmented, with tailored instructions for each stage, optimizing the scoring process for efficiency and accuracy. By employing this approach, AES systems can effectively handle the complexity of scoring essays by breaking them into manageable components.

3.6 Evaluation

3.6.1 Evaluation Metrics Used

The evaluation plan aims to assess the grading capabilities of the GPT-4 model by comparing its performance with human grading conducted by university lecturers. Two key metrics, Quadratic Weighted Kappa (QWK) and accuracy will be employed to evaluate the model's performance.

Accuracy Metrics

Accuracy measures the proportion of correctly assigned scores by GPT-4 compared to the scores given by human lecturers. It provides insight into the overall correctness of GPT-4's grading. Following is the formula to calculate the accuracy.

$$Accuracy = (TP + TN)/(TP + FP + TN + FN)$$

QWK Metrics

Quadratic Weighted Kappa (QWK) measures the agreement between two raters (GPT-4 and human lecturers) in assigning scores to the same set of student answers. It considers the agreement occurring by chance, thus providing a robust measure of inter-rater agreement. QWK helps us understand how closely their ratings match up.

The QWK value ranges from -1 to 1. QWK of 1: Perfect agreement. Both raters always give the same ratings. QWK of 0: No agreement. Ratings are as good as random guesses. Negative QWK: Less agreement than random guesses. Raters are even more inconsistent than if they were guessing randomly.

So, the higher the QWK value, the better the agreement between the raters.

Landis and Koch (1977) [29] have presented the guidelines for interpreting Kappa values. According to the guidelines, we can get an interpretation of the QWK values.

Table 3.1 presents the interpretation values.

QWK Range	**Interpretation**
0.00 - 0.20	Slight
0.21 - 0.40	Fair
0.41 - 0.60	Moderate
0.61 - 0.80	Substantial
> 0.80	Almost Perfect

Table 3.1: Landis and Koch (1977) guidelines for interpreting Kappa values.

The dataset consists of student answers to a set of questions, with each answer independently graded by both GPT-4 and university lecturers. The scores assigned by the lecturers are considered the ground truth for comparison with those generated by GPT-4. The evaluation procedure involves preprocessing the data, grading by GPT-4, calculating evaluation metrics, and analyzing results.

3.6.2 Feedback Generation

The student answers graded by the GPT can be awarded with feedback about the students' answers. As other researchers have shown in their research, these LLMs can generate personalized feedback based on the answers provided by the students. In this research, we prompted the LLM with the granted score and the explanation about how GPT has granted that score and asked it to generate feedback following the Hattie and Timperley (2007) Model of feedback [30]. Figure 3.4 describes the used model.

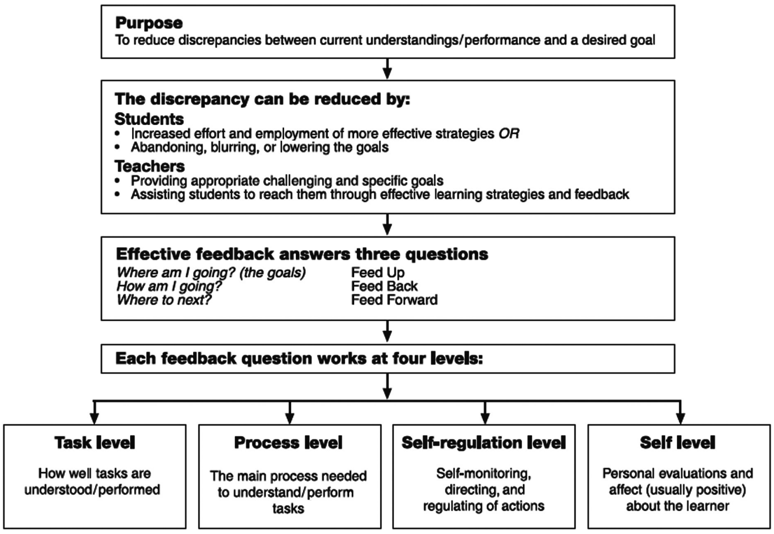

Purpose
To reduce discrepancies between current understandings/performance and a desired goal
The discrepancy can be reduced by:
Students
• Increased effort and employment of more effective strategies OR
• Abandoning, blurring, or lowering the goals
Teachers
• Providing appropriate challenging and specific goals
• Assisting students to reach them through effective learning strategies and feedback
Effective feedback answers three questions
Where am I going? (the goals)
Feed Up
How am I going?
Feed Back
Where to next?
Feed Forward
Each feedback question works at four levels:
Task level
How well tasks are understood/performed
Process level
The main process needed to understand/perform tasks
Self-regulation level
Self-monitoring, directing, and regulating of actions
Self level
Personal evaluations and affect (usually positive) about the learner

Figure 3.4: Hattie and Timperley (2007) Model of feedback

As shown in Figure 3.4, this model uses 4 levels to form feedback. According to the Hattie and Timperley (2007) Model of feedback, the first level is the task level. This level focuses on the task. It explains how well the task has been done. With this research, this level explains how well the student has answered the question. The second level focuses on the process. It suggests ways or strategies to complete the task and improve the answer. With this research, we focus on the things students should have focused on to answer the question accurately and improve the answer. The next level is the self-regulation level, which helps students keep track of their learning by giving them a gentle reminder to check whether they have used what they learned to answer the question. The last level is the self-level, which provides personal evaluation and affirmations to the students. This model can enhance student learning to align with their desired learning goals.

Below is the prompt we used.

```
A student answer was evaluated and granted a score which is provided below.
Score: {Score} out of {Full_Score}

An explanation for granting that score is given below.
Explanation: {Explanation}

Context information which is mentioned in the Explanation is the course content the student
should study.

Provide a feedback paragraph to the student by referring to Score and Explanation. Feedback
should include one sentence for each of the following
 levels .
   – How well student has answered the question
   – Suggest main things student should understand to answer the question and how to
   improve answer
   – Remind student to keep track of their own learning
   – Personal evaluation and affirmation to the student
```

3.7 Summary

This chapter mainly explained the methodology we followed in this research. Starting with the overview of the research, we outlined the design overview, emphasizing the use of the RAG for AES (Automated Essay Scoring) and our approach to the study. Subsequently, we delved into data collection and preprocessing, detailing the data source, collection methods, anonymization procedures, and data transformation techniques employed. Additionally, we addressed the crucial step of data cleaning to ensure the integrity and quality of our dataset.

A pilot test was conducted to validate our approach and refine our methodology, laying the foundation for subsequent stages of the research. We then moved on to assessing with GPT-4, a pivotal aspect of our study, wherein we generated results, marked approaches, and determined prompt levels to evaluate the effectiveness of our model. Furthermore, we discussed the methodology for evaluating accuracy, elucidating the metrics utilized, and generating feedback.

Chapter 4

Results and Evaluation

4.1 Pilot Test

In the pilot study, we tested three state-of-the-art GPT models: GPT-3.5-Turbo, GPT-3.5-Turbo-Instruct and GPT-4. The objective of this experiment was to comparatively assess the performance of these models in the automated assessment of students answers. Specifically, the pilot study tested two questions we selected from the Professional Issues in IT 2022 examination paper as shown in the Table 4.1. We utilized a basic prompt wherein we only provided the question and student answers in the prompt. This experiment was designed to compare the capabilities of several OpenAI models to perform automated marking.

Q.No	Question	Type
3(a)	What is the difference between a centralized company and a decentralized company?	Precise Answer
4(a)	Differentiate Retail Software Agreements and Corporate Software Agreements by highlighting at least THREE (3) different aspects	Open-ended

Table 4.1: Questions assessed with the Pilot Test

This experiment tested one hundred data points, considering samples that included 50 students answers per question. Based on the results in the following Table 4.2, each questions performance metric, the Quadratic Weighted Kappa, was compared.

Model	**4(a)**	**3(a)**
GPT-4	0.4592	0.8811
GPT-3.5-Turbo	0.3713	0.5924
GPT-3.5-Turbo-Instruct	0.2378	0.5557

Table 4.2: Results of the Pilot Test

Upon analyzing the performance of each model, we observed that the GPT-3.5-Turbo-Instruct model exhibited the lowest level of performance, indicating minimal agreement between the score assigned by the model and the score assigned by the human evaluators for each students answer to selected questions. In contrast, the GPT-4 model showed considerably higher performance, especially for questions with precise answers, as it achieved approximately an agreement level of 0.9. Based on the findings, we selected the GPT-4 model for further examination during the final experimentthe final experiment aimed to test the capability of the GPT-4 model across several subjects.

4.2 Final Experiments

The GPT-4 model demonstrated higher performance during the pilot study. Hence, the GPT-4 model is employed for grading students answers in the final experiment. In this experiment, the grading capabilities of GPT-4 were tested using two subjects: Professional Issues in IT and Information Security. In addition to the approach tested during the pilot test, we experimented with several other new techniques built upon the initial method. These collectively formed five distinct approaches to provide prompts when automating students answers marking with GPT models. The five approaches are as follows,

- Providing only the question and student answer in the prompt template (Q+A)
- Providing the question, student answers, and grading rubric in the prompt template (Q+A+R)
- Providing the question, student answers, grading rubric, and model answer in the prompt template (Q+A+R+MA)
- Providing the question, student answers, grading rubric, and course content as context information in the prompt template (Q+A+R+C)
- Providing the question, student answers, grading rubric, model answer, and course content as context information in the prompt template (Q+A+R+MA+C)

Using the approaches described above, we tested two types of questions for Professional Issues in IT. The question types included open-ended questions and questions requiring precise answers. Then we marked student answers for the Information Security subject. The main focus of this experiment was to investigate three aspects aligning with the initial research questions. Firstly, we explored the potential of improving the automated assessment of students answers for different question types by employing Retrieval Augmented Generation. Secondly, we examined the performance of Open AI GPT-4 model in automated marking across various subjects by employing RAG. Moreover, the impact of various prompt engineering techniques on enhancing the performance of the GPT-4 model in the context of automated students answer marking was assessed. Accordingly, we discuss the results of the considered three aspects in the subsequent section, which describes relevant experiments,

1. The experiment conducted for professional issues in IT subject
2. The experiment conducted for Information Security subject
3. The experiment conducted with improved prompt employing prompt engineering techniques.

4.2.1 The experiment conducted for professional issues in IT subject

Table 4.3 contains the questions that were selected from the Professional Issues in IT subject 2021(1a, 1c) and 2022(3a, 1c, 4b, 4a) papers for the final experiment.

Q.No	Question	Type
1(c)	In most dynamic groups there are members with diverse personalities. List the four (04) main types of personalities.	Precise Answer
1(a)	List four (4) characteristics usually included in the elements of Group Dynamics	Precise Answer
3(a)	What are the THREE (3) roles that an ISP may play	Precise Answer
1(c)	What is the difference between a Centralized company and a Decentralized company?	Open-ended
4(b)	Explain the difference between Primary Infringement and Secondary Infringement of copyrights	Open-ended
4(a)	Differentiate Retail Software Agreements and Corporate Software Agreements by highlighting at least THREE (3) different aspects	Open-ended

Table 4.3: Questions selected from the Professional Issues in IT subject

The experiment was conducted with a dataset comprising 600 data points derived from the responses of 100 students answers per question. Following the automated marking of these responses, the resultant data were analyzed using specified evaluation metrics: Quadratic Weighted Kappa (QWK) and Accuracy. The outcome of this analysis is presented in Table 4.4 with QWK and Table 4.5 with Accuracy.

The basic approach, in which we provided Only the question and student answers in the prompt and guided the GPT model to grade student answers, demonstrated modest results with accuracy peaking at 0.48 and with a peak QWK value of 0.689. The results of this approach did not surpass the accuracy achieved by other research done using GPT models in Automated Essay Scoring such as [4].

Approach	**1a (2021)**	**1(c)2021**	**3a(2022)**	**1c(2022)**	**4b(2022)**	**4a(2022)**
Q+A+R+MA+C	0.983	0.928	0.943	0.843	0.868	0.538
Q+A+R+MA	0.997	0.937	0.969	0.899	0.841	0.505
Q+A+R+C	0.986	0.928	0.933	0.846	0.809	0.487
Q+A+R	-0.558	0.514	0.194	0.610	0.311	0.407
Q+A	-0.456	0.689	0.347	0.342	0.142	0.253

Table 4.4: Comparison of QWK values for assessment of student answers for Professional Issues in IT with basics prompts

Approach	**1a (2021)**	**1(c)2021**	**3a(2022)**	**1c(2022)**	**4b(2022)**	**4a(2022)**
Q+A+R+MA+C	0.94	0.88	0.96	0.76	0.81	0.59
Q+A+R+MA	0.98	0.88	0.94	0.8	0.84	0.61
Q+A+R+C	0.93	0.85	0.94	0.77	0.75	0.59
Q+A+R	0.14	0.61	0.36	0.69	0.48	0.58
Q+A	0.07	0.48	0.37	0.08	0.17	0.07

Table 4.5: Comparison of Accuracy values for assessment of student answers for Professional Issues in IT with basics prompts

Furthermore, the accuracy of scores assigned by GPT-4, compared to the teachers scores, significantly fluctuated across questions.

In the second approach, we incorporated a grading rubric along with the question and students answers to provide clear guidance on the marks distribution for each question. Though this approach led to an improvement in performance related to grading open-ended questions, performance for marking questions with precise answers declined across the considered evaluation metrics. Despite providing grading rubrics along with the questions and students answers, the accuracy peaked at 0.69 and the highest QWK value achieved was 0.610, yet the performance was unsteady.

As a further refinement, the course content to relevant questions was integrated into the prompt as external knowledge, thereby enhancing the prompting approach. The GPT model performed the marking process, adhering to the grading rubric and comparing the students answers with the provided course content. This approach produced promising outcomes, achieving accuracy exceeding 0.7 and more than 0.8 QWK values for five out of the six questions utilized in the experiment. Moreover, the model demonstrated stable accuracy and QWK levels without any major fluctuations for the majority of the questions. Nevertheless, the accuracy for one particular question still remained below 0.5 indicating an area that needed further optimization. Hence, as the next technique, we utilized the model answers prepared by teachers for each question, incorporating them into the prompt template instead of the course content. In this method, the GPT model compared the student's answers with the model answer while adhering to the grading rubric. We could observe improved performance demonstrating elevated accuracy and QWK values exceeding 0.5 for each question. Furthermore, this method exceeded the 0.8 QWK value for most of the questions that we considered in the experiment.

Proceeding further, we combined the previous two approaches to maximize the benefit in performance. We provided more comprehensive input related to the question in the prompt by including the question, students answers, relevant course content and model answers, and the grading rubric. Within this approach, the GPT model performed the marking process, adhering to the grading rubric and comparing the students answers against both model answers and the course content. However, the results revealed a decline in accuracy and agreement levels (QWK) compared to the approach we provided only the model answer. Nevertheless, this technique sustained the performance levels for each question above 0.5, and consistent stability of performance across the questions.

4.2.1.1 The Average Accuracies and QWK Values for each approach.

Approach	QWK	Accuracy
Q+A+R+MA+C	0.85	0.82
Q+A+R+MA	0.858	0.84
Q+A+R+C	0.832	0.81
Q+A+R	0.246	0.48
Q+A	0.22	0.21

Table 4.6: Average Accuracy and QWK values for each approach in Professional Issues in IT

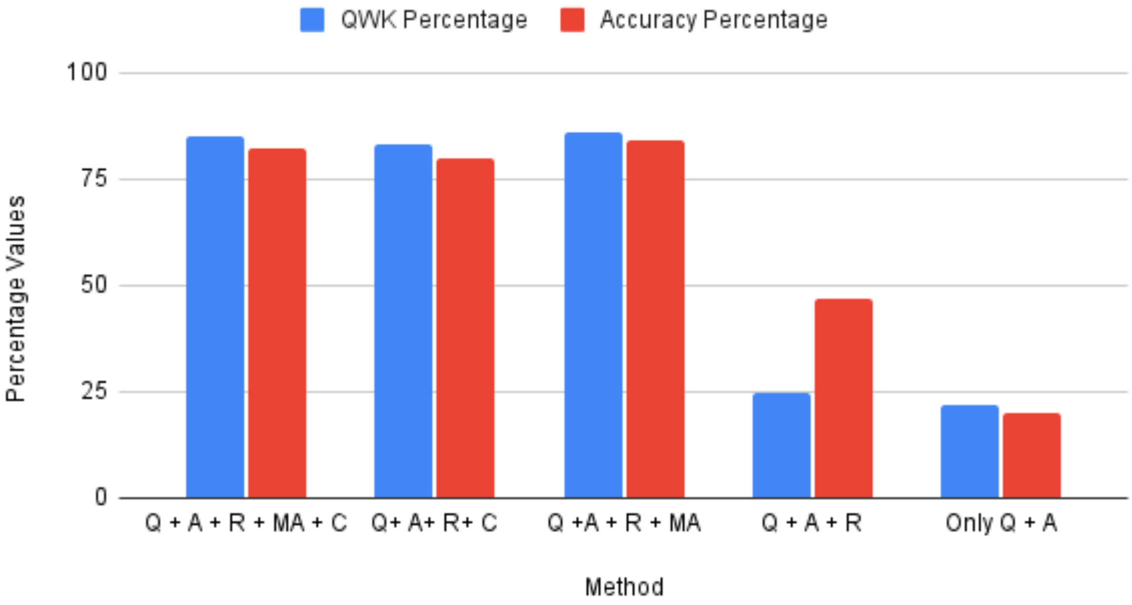


Figure 4.1: Average Accuracy and QWK values for each approach in Professional Issues in IT

Analyzing the average accuracies and QWK values associated with each approach for the six questions, as shown in Table 4.6, revealed significant findings. Conforming to the guidelines presented by Landis and Koch[29] it can be observed that, providing only the question and students answers demonstrates the fair level of agreement between the score generated by the model and those assigned by the human evaluator. Though incorporating grading rubrics enhanced the QWK value and accuracy of automated marking, the QWK level remained at a fair level of agreement. Conversely, integrating the model answer in the prompt or providing course content as external knowledge significantly elevated the QWK level peaking at 0.8 and above reaching the Almost perfect level of agreement.

4.2.1.2. Comparison Between the QWK values: open-ended questions vs. questions with precise answers.

Approach	Precise	Open
Q+A+R+MA+C	0.95	0.75
Q+A+R+MA	0.95	0.71
Q+A+R+C	0.97	0.75
Q+A+R	0.05	0.44
Q+A	0.19	0.25

Table 4.7: Comparison Between the QWK values of openended questions vs. questions with precise answers

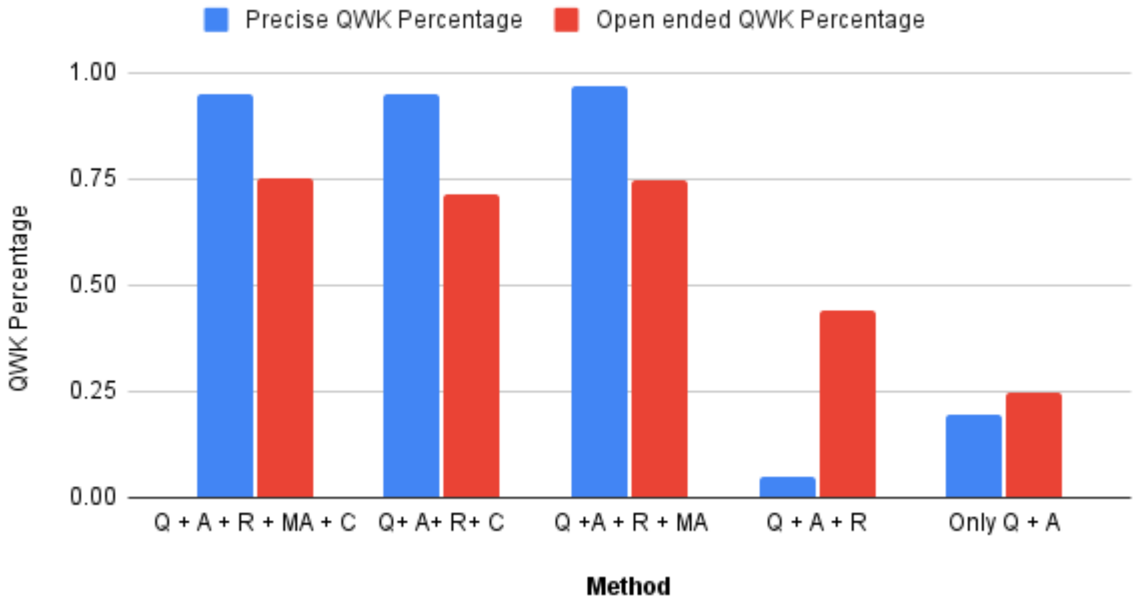


Figure 4.2: Comparison Between the QWK percentages of open ended questions vs. questions with precise answers

In our comparative analysis of average QWK values for open-ended questions and questions with precise answers as shown in Table 4.7, We observed that both question types demonstrated low QWK values below 0.5 in the basic approach. Therefore, to enhance the strategy we integrated a grading rubric into the prompt. While this exhibited an improvement of the QWK for open-ended questions, the performance for questions with precise answers, further declined reaching the lowest such of 0.05.

When we incorporated the model answer into the prompt we noticed a significant improvement in the performance. This adjustment resulted in a QWK value peaking at 0.97 for questions with precise answers, a performance that remained steady across the other approaches wherein we provided course content (QWK 0.95) and both components (QWK 0.95). Similarly in these approaches, the performance for grading open-ended questions improved considerably, peaking at 0.75 QWK value and remained stable across the three approaches.

4.2.2 The experiment conducted for the Information Security subject.

The following questions (Table 4.8) from the Information Security Subject 2023 paper were selected for the experiment. Our main intention was to investigate the capabilities of the GPT-4 model to conduct automated assesment of students answers across several subjects. In addition, we focused on exploring the performance of marking open-ended questions and stabilizing the findings we drew in the aforementioned experiment with the Professional Issues in IT (PIIT) subject.

Q.No	**Question**	**Type**
4(e)	What is the difference between computer crime and cybercrime?	Open-ended
3(d)	Briefly explain the difference between making data backups and making data archives as apart of security maintenance of systems?	Open-ended
3(f)	The two SQL commands GRANT and REVOKE can be used in managing database security.Briefly describe the functionality of them.	Open-ended
4(a)	Briefly describe what is meant by malware?	Open-ended

Table 4.8: Questions selected from the Information Security subject.

The experiment was conducted using 400 data points, considering 100 students answers per question, and it was concluded after marking students answers using the previously described five approaches and receiving the following results.

Approach	**4e**	**3d**	**3f**	**4a**
Q+A+R+MA+C	0.878	0.745	0.888	0.821
Q+A+R+MA	0.813	0.783	0.872	0.831
Q+A+R+C	0.891	0.893	0.921	0.941
Q+A+R	0.281	0.494	0.881	0.149
Q+A	0.105	0.504	0.639	0.336

Table 4.9: Comparison of QWK values for assessment of student answers for PIIT with basics prompts

Approach	**4e**	**3d**	**3f**	**4a**
Q+A+R+MA+C	0.79	0.74	0.88	0.78
Q+A+R+MA	0.79	0.77	0.89	0.88
Q+A+R+C	0.88	0.9	0.93	0.96
Q+A+R	0.51	0.59	0.93	0.37
Q+A	0.11	0.26	0.17	0.57

Table 4.10: Comparison of Accuracy values for assessment of student answers for Information Security with basics prompts

When analyzing the above performance outcomes, we observed The QWK in the Table 4.9 and the accuracy in the Table 4.10 for first two approaches were low and drastically fluctuating across the questions. Even though the grading rubrics were provided as the second step, they did not significantly improve performance. As we proceeded with the other three approaches, which showed promising results in the experiment with PIIT, the performance of the GPT-4 model was improved for the Information Security subject as well. Not only did these approaches reach above 0.8 in performance metrics, but the performance level was maintained within the same range for each question without major fluctuations. Subsequently, we discuss this experiments average accuracy levels and QWK values.

4.2.2.1 Average accuracies and QWK values for each approach

Approach	QWK	Accuracy
Q+A+R+MA+C	0.833	0.8
Q+A+R+MA	0.825	0.83
Q+A+R+C	0.912	0.92
Q+A+R	0.451	0.6
Q+A	0.396	0.28

Table 4.11: Average Accuracy and QWK values for each approach in Information Security

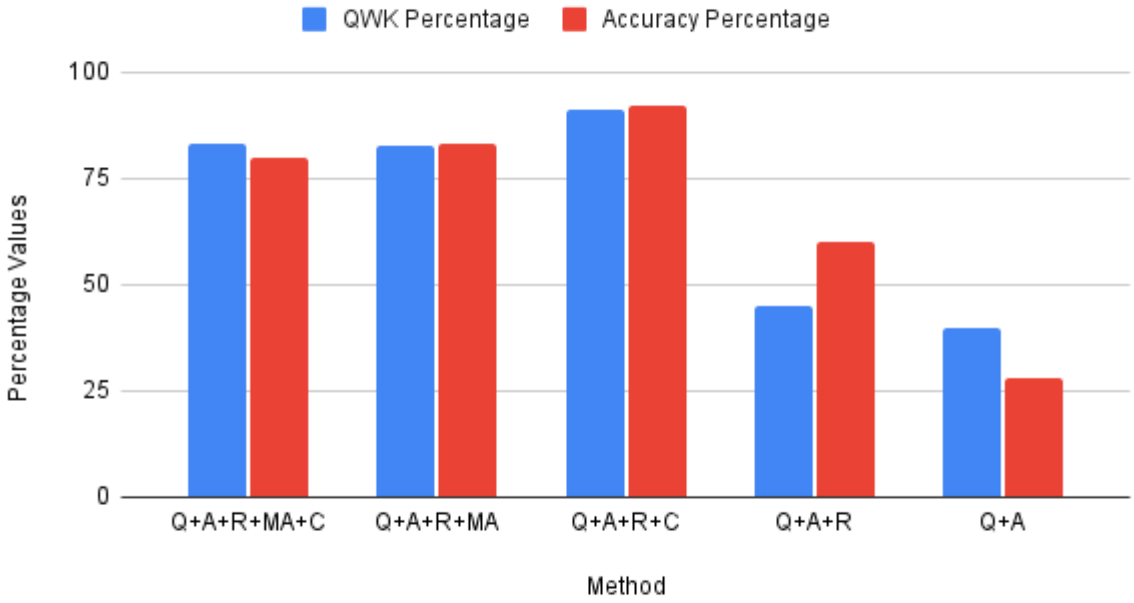


Figure 4.3: Comparison of QWK Percentage and Accuracy Percentage for each approach in Information Security

In this experiment, a similar pattern was demonstrated in how QWK and accuracy levels improved as shown in the Table 4.11 and the Figure 4.3. Providing only questions and answers in the prompt achieved a fair level of agreement between the actual scores assigned by human evaluators and predicted scores by the model. We observed the QWK level being enhanced to a moderate level of agreement by the grading rubric provided in the next approach. Similar to the previous experiment, providing model answers or course contents could enhance the performance of the GPT-4 model in students answer marking in the Information Security domain. As the table presents, those approaches achieve above 0.8 accuracy and QWK value, which depicts an almost perfect agreement between the human evaluators scores and the GPT model-generated scores.

4.2.3 Prompt-enhancing Experiment

The main focus of this experiment was to examine the potential to enhance the performance of the GPT model in assessing student answers through the implementation of prompt engineering techniques. These techniques included establishing a chain-of-thought technique that refined the models logical reasoning ability, providing a persona and a task breakdown that guided the GPT model step by step. By integrating these prompt engineering strategies, the study intended to explore their collective impact on enhancing the model's ability to grade students answers accurately. The following table 4.12 and table 4.13 presents the QWK and accuracy results respectively. It contains the results of each question selected from Professional Issues in IT papers.

Prompt Version	1a (2021)	1(c)2021	3a(2022)	1c(2022)	4b(2022)	4a(2022)
Basic Prompt	0.986	0.928	0.933	0.846	0.809	0.487
Version 1	0.986	0.926	0.909	0.695	0.567	0.479
Version 2	0.987	0.926	0.905	0.708	0.497	0.5
Version 3	0.967	0.922	0.885	0.828	: 0.686	0.491

Table 4.12: Comparison of QWK values for assessment of student answers for Professional Issues in IT with different prompt versions for Q+A+R+C approach

Prompt Version	1a (2021)	1(c)2021	3a(2022)	1c(2022)	4b(2022)	4a(2022)
Basic Prompt	0.93	0.85	0.94	0.77	0.75	0.59
Version 1	0.84	0.84	0.91	0.73	0.58	0.55
Version 2	0.94	0.87	0.9	0.73	0.59	0.58
Version 3	0.88	0.83	0.88	0.75	0.62	0.5

Table 4.13: Comparison of Accuracy values for assessment of student answers for Professional Issues in IT with different prompt versions for Q+A+R+C approach

As the initial step towards enhancing the basic prompt, we included the chain of thought prompting technique in the prompt forming version one. The objective was to guide the GPT models through the grading process by allowing the model to think step by step. Implementing this technique the model achieved a QWK value exceeding 0.9 and an accuracy level above 0.8 for the questions with precise answers. However, for open-ended questions, the QWK level remained within the range of 0.5 - 0.7 accompanied by accuracy ranging from 0.6 - 0.7.

Advancing to the second prompt version we employed a combination of persona pattern and Chain of thought technique. The persona pattern introduced the persona of an examiner with expertise in the subject that was being graded, aimed at providing the model with an understanding of the scope and responsibilities associated with the assigned role. This technique did maintain the QWK level and accuracy for questions with precise answers at a consistent level with the previous prompt version, by maintaining a QWK level exceeding 0.9. Additionally, it depicted a marginal enhancement of open-ended questions grading performance for two questions although one question(4b) demonstrated a decline in performance.

Prompt version 3 was developed through the integration of a task breakdown into prompt version 2 comprising CoT and persona pattern. Conforming to the outcome of the experiment, it was observed that the QWK values for grading questions with precise answers demonstrated an almost perfect level of agreement while two of three open-ended questions achieved substantial levels of agreement exceeding 0.7 as the QWK level.

Chapter 5

Discussion

5.1 Discussion

5.1.1 Results Discussion

5.1.1. Pilot Test

The pilot test was conducted to explore the potential GPT model to be employed in the final experiment. We used the latest GPT models introduced by Open AI, including GPT instruct, GPT 3.5 Turbo, and GPT 4, to grade one question per question type. The analysis of the outcomes depicted that the GPT Instruct model exhibited the lowest level of performance among the three. It indicated the minimal agreement between the score assigned by the model and those assigned by the human evaluators for each students answer to selected questions. The GPT Instruct model tended to generate incorrect responses frequently.

The subsequent model, GPT 3.5 Turbo, displayed improved performance compared to the GPT instruct model. However, it still yielded a low QWK level when grading the open-ended question. In contrast, the GPT 4 model showed considerably higher performance, especially for questions with precise answers, as it achieved a perfect level of agreement according to the QWK interpretation guidelines. Even for open-ended questions, the GPT 4 model demonstrated a moderate level of agreement, surpassing the predecessors that could reach a fair level of agreement. The findings depicted the GPT 4 models potential for enhancing automated grading, and it was employed for further examination during the final experiment. Nevertheless, another noteworthy finding was observed across all models, that they demonstrated significantly better performance in grading questions with precise answers than in grading open-ended questions.

5.1.2 Final experiments

When analyzing the results, it can be noted that providing only the question and answer in the prompt is insufficient to guide the Large Language model in grading students answers. This approach not only holds low accuracy, but the scores assigned by the GPT model in this approach far deviate from the scores assigned by the human evaluators. This is indicated by the low QWK value of this approach. Grading rubrics provide guidelines when grading students answers by defining the possible instances of a students answer and their respective scores. Hence, including grading rubrics in the prompt has significantly improved the accuracy of scores provided by the GPT model. Grading rubrics did not considerably enhance the agreement between the GPT-assigned scores and the human evaluators scores, while including the model answer for each question upsurges the models performance. The lecturers prepare these model answers based on the course content, defining the scope of the expected answer for each question. It can further guide the large language model, indicating the knowledge margins that should be considered when grading the students answers. Similarly, providing course content in the prompt can enhance the models performance significantly. The course content provides the knowledge base that should be considered during students answers grading. The GPT models can understand the content and identify all the possible answers to the given questions. This approach performs the same as the previous approach in which only the model answer is provided. Further proceeding with the techniques, it can be observed that including the course content and model answers in the same prompt also reaches a significantly improved performance level. In this approach, the knowledge margins and the knowledge base are provided together so that the model can consider the expected answer and draw other possible answers from the course content.

Analyzing the average performance levels for open-ended questions and questions with precise answers reveals notable findings. The basic approach involving only the question and students answers yielded modest results for both question types. Furthermore, Incorporating grading rubrics significantly enhanced the QWK value for open-ended questions; it appeared to decline the performance when grading questions with precise answers.

After thorough consideration of outcomes, the noteworthy observation was, regardless of the question type the three approaches in which the model answer and course content were provided demonstrated promising performance. Consequently, the main finding of this study evidently demonstrates that providing course content as external knowledge to the model within the prompt using the RAG approach can significantly improve the agreement level and accuracy of GPT models in the context of automated grading of students answers.

We received the same pattern when applying the approaches to the subject of Information Security. When the model answers and the relevant course content were provided in the prompt, the performance was enhanced, increasing the accuracy and reducing the deviations from the scores assigned by the human evaluator. As LLMs are trained on a wide range of knowledge, they apply that knowledge when grading the students answers, which can affect the grade as it is considered a vast knowledge area than what is expected from the students. Providing the model answer or course content as external knowledge provides a solution as it defines the scope of the expected answer and provides the LLM with access to the knowledge scope from which it can determine other possible correct answers.

When it comes to employing prompt engineering techniques, the chain of thought technique and task breakdown enable the reasoning ability of the model. Most importantly, it provided the model with space to think clearly and reduced hallucinations or incorrect response generations. It not only supported deriving the reasons behind the grade assigned by the model to a particular students' answer but also played a crucial role in providing detailed feedback. Especially the task breakdown technique led to a significant improvement in grading open ended questions than previously employed techniques. As task breakdown technique specifically defined the clear tasks in the grading process it reduced the incorrect response generation when grading open-ended questions. The prompt versions with different combinations of the aforementioned prompt engineering techniques performed promisingly in the automated grading of students answers during the experiment. Nevertheless the outcomes demonstrating combining all prompt engineering techniques (Prompt version 3) could not outperform the basic prompt. Instead, it maintained a similar level of performance for grading questions with precise answers. For open ended questions the performance was improved across the three prompts with prompt engineering techniques, but prompt version 3 did not perform up to the level of the basic prompt.

When considering the nature of the selected subjects, both lie within the category of theoretical subjects in the IT domain. As per the experiment outcome and their analysis, we can state that it is possible to successfully employ Large Language models such as GPT for automated grading of students answers during examinations or assignments for theoretical subjects in the IT domain. Since providing course content has proven effective in enhancing accuracy, it allows the academic staff to grade students answers even without deriving model answers for every question. It benefits the academic staff by reducing the time invested in grading student answers manually and reducing human errors by employing these methods. Further, it enforces more transparency as the LLM provides logical reasoning of how the grading was carried out in the feedback provided to the students.

Chapter 6

Conclusion

6.1 Introduction

The chapter focuses on the conclusions drawn upon the completion of the research. Explaining the results which were presented in chapter 5 and the contribution towards the research community. In section 6.4, the limitations of the proposed model will be explained. Finally, the future work will be discussed.

6.2 Research conclusions by objectives

6.2.1 Objective 1

Evaluate the impact of prompt design on the accuracy of the automated marking process, identifying key elements within prompts that can improve automated marking.

Table 4.12 and Table 4.13 show the comparisons of QWK and Accuracy for each prompt version created utilizing the different prompt engineering techniques such as chain-of-thought, persona pattern and task breakdown. All these techniques were able to improve the reasoning capabilities of GPT-4 and reduce hallucinations. When analyzing these tables, it can be observed that precise answers were marked consistently by all the prompt versions but there is slight decrease in assessing the open ended answers when basic prompts are modified to version prompt. But it was enhanced to a certain extent significantly by the task breakdown prompt engineering technique . However, these engineered prompts were not able to surpass the basic prompts.

6.2.2 Objective 2

Evaluate the impact of Retrieval Augmented Generation (RAG) to enhance the automated marking process.

Considering the QWK and Accuracy values generated by assessing students answers for both Professional issues in IT (Table 4.6) and Information Security (Table 4.11), it can be observed that the performance of GPT models can be improved by providing external knowledge. The given scores by GPT models have a higher agreement with the scores given by lecturers when model answer, course content or both are given.

6.2.3 Objective 3

Investigating the possibility of designing an improved prompt for all question categories

Within this research RAG approach were used to modify the basic prompt. This approach showcased that the marking process can be improved by providing relevant course content to the LLMs. Apart from that there were two types of questions available within the dataset as questions with precise answers and questions with open ended answers. Table 4.7 shows that, average QWK for both precise answers and open ended answers have achieved higher QWK values when providing course content. Apart from that when providing the model answer, both of them also achieved higher QWK values. However , when the prompts were modified using prompt engineering techniques, all engineered prompts fail to achieve higher QWK values for open ended answers even though they mark precise answers with higher QWK values.

6.2.4 Objective 4

Examine how well GPT models can handle different subjects and types of questions in the marking context.

Table 4.6 and Table 4.11 show that average QWK values for both Professional Issues in IT and Information Security subjects when providing GPT with course content, model answer or both, the agreement between GPT granted scores and lecturer granted scores is high. It shows that GPT performs the marking process at a similar level for both subjects following different approaches.

6.3 Limitations

Our research was limited to assessing the short answers for theoretical subjects utilizing the RAG approach and various prompt engineering techniques. Assessment of Essay questions and questions with diagrams wasnt involved.

Further, the dataset that was scanned only contained Questions, Student answers, model answer and scores given for student answers. A grading rubric was not available in the dataset. Therefore, a simple grading rubric format was developed by analyzing the marking patterns of lecturers.

6.4 Recommendation

According to the findings of this research, the lowest agreement with the human evaluator score is shown by the marking done using questions and answers. A slight improvement is shown when additionally providing the grading rubrics. The other three scenarios of providing model answers, course content, and providing both have shown higher levels of agreement with the scores provided by human evaluators. Not only that but they also have a similar level of agreement when comparison is done among themselves. Therefore, it shows that only providing the question and answer doesnt achieve high performance. Providing the grading rubric improves the marking process by guiding the GPT model and explaining how the scores should be granted. Providing any or both of the model answers and course content improves the marking process vastly. The following recommendations can be made considering the finding and their implications.

- Instead of creating a model answer specifically for a question, only the course content can be provided along with the grading rubric to mark student answers.

- Providing a grading rubric is necessary because it defines the score range that can be granted for a particular question along with possible scenarios of student answers.

- Providing both model answers and course content is more suitable than providing only one of those components.

6.5 Future Work

In future endeavors, expanding the dataset to include a greater number of data points per subject could significantly increase the depth and robustness of our analysis. While our current research relied on a dataset comprising 1000 data points, exploring larger datasets, particularly those available in the public domain, presents an opportunity to discover deeper insights and validate the generalizability of our findings. Transitioning towards public datasets would not only enhance the reproducibility of our results but also facilitate collaboration and knowledge sharing within the wider research community. By leveraging publicly available datasets, we can ensure greater transparency in our research methodology.

Furthermore, future investigations could benefit from a comparative analysis with baseline models and alternative language model architectures. Although the GPT LLMs was used in this study, investigating alternative LLM versions could provide significant knowledge about the relative effectiveness and suitability of various LLMs in student naswer assessment.

Additionally, incorporating automated marking techniques, particularly utilizing the RAG approach for coding questions, holds promise for streamlining evaluation processes and enhancing efficiency in educational assessments. By integrating external information into the marking process, we can augment the accuracy and consistency of evaluations while accommodating the complexities inherent in coding-based assessments.

References

[1] H. B. Ajay *et al.*, "Analysis of essays by computer (aec-ii). final report.," 1973.

[2] J. Li, L. Gui, Y. Zhou, D. West, C. Aloisi, and Y. He, "Distilling chatgpt for explainable automated student answer assessment," *arXiv preprint arXiv:2305.12962*, 2023.

[3] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "Ragas: Automated evaluation of retrieval augmented generation," *arXiv preprint arXiv:2309.15217*, 2023.

[4] A. Mizumoto and M. Eguchi, "Exploring the potential of using an ai language model for automated essay scoring," *Research Methods in Applied Linguistics*, vol. 2, no. 2, p. 100050, 2023.

[5] K. Ragupathi and A. Lee, "Beyond fairness and consistency in grading: The role of rubrics in higher education," *Diversity and inclusion in global higher education: Lessons from across Asia*, pp. 73–95, 2020.

[6] C. Xiao, W. Ma, S. X. Xu, K. Zhang, Y. Wang, and Q. Fu, "From automation to augmentation: Large language models elevating essay scoring landscape," *arXiv preprint arXiv:2401.06431*, 2024.

[7] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[8] D. Ramesh and S. K. Sanampudi, "An automated essay scoring systems: a systematic literature review," *Artificial Intelligence Review*, vol. 55, no. 3, pp. 2495–2527, 2022.

[9] P. W. Foltz, D. Laham, and T. K. Landauer, "The intelligent essay assessor: Applications to educational technology," *Interactive Multimedia Electronic Journal of Computer-Enhanced Learning*, vol. 1, no. 2, pp. 939–944, 1999.

[10] L. M. Rudner and T. Liang, "Automated essay scoring using bayes' theorem," *The Journal of Technology, Learning and Assessment*, vol. 1, no. 2, 2002.

[11] M. A. Sultan, C. Salazar, and T. Sumner, "Fast and easy short answer grading with high accuracy," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1070–1075, 2016.

[12] S. Vij, D. Tayal, and A. Jain, "A machine learning approach for automated evaluation of short answers using text similarity based on wordnet graphs," *Wireless Personal Communications*, vol. 111, pp. 1271–1282, 2020.

[13] N. Süzen, A. N. Gorban, J. Levesley, and E. M. Mirkes, "Automatic short answer grading and feedback using text mining methods," *Procedia Computer Science*, vol. 169, pp. 726–743, 2020.

[14] K. Taghipour and H. T. Ng, "A neural approach to automated essay scoring," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 1882–1891, 2016.

[15] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[16] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *et al.*, "Improving language understanding by generative pre-training," 2018.

[17] R. Yang, J. Cao, Z. Wen, Y. Wu, and X. He, "Enhancing automated essay scoring performance via fine-tuning pre-trained language models with combination of regression and ranking," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1560–1569, 2020.

[18] C. Sung, T. I. Dhamecha, and N. Mukhi, "Improving short answer grading using transformer-based pre-training," in *Artificial Intelligence in Education: 20th International Conference, AIED 2019, Chicago, IL, USA, June 25-29, 2019, Proceedings, Part I 20*, pp. 469–481, Springer, 2019.

[19] M. O. Dzikovska, R. D. Nielsen, and C. Leacock, "The joint student response analysis and recognizing textual entailment challenge: making sense of student responses in educational applications," *Language Resources and Evaluation*, vol. 50, pp. 67–93, 2016.

[20] D. Mhlanga, "The value of open ai and chat gpt for the current learning environments and the potential future uses," *Available at SSRN 4439267*, 2023.

[21] K. P. Yancey, G. Laflair, A. Verardi, and J. Burstein, “Rating short l2 essays on the cefr scale with gpt-4,” in *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*, pp. 576–584, 2023.

[22] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, and D. C. Schmidt, “A prompt pattern catalog to enhance prompt engineering with chatgpt,” *arXiv preprint arXiv:2302.11382*, 2023.

[23] J. Shin, C. Tang, T. Mohati, M. Nayebi, S. Wang, and H. Hemmati, “Prompt engineering or fine tuning: An empirical assessment of large language models in automated software engineering tasks,” *arXiv preprint arXiv:2310.10508*, 2023.

[24] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, *et al.*, “Chain-of-thought prompting elicits reasoning in large language models,” *Advances in neural information processing systems*, vol. 35, pp. 24824–24837, 2022.

[25] O. Ovadia, M. Brief, M. Mishaeli, and O. Elisha, “Fine-tuning or retrieval? comparing knowledge injection in llms,” *arXiv preprint arXiv:2312.05934*, 2023.

[26] O. Ram, Y. Levine, I. Dalmedigos, D. Muhlgay, A. Shashua, K. Leyton-Brown, and Y. Shoham, “In-context retrieval-augmented language models,” *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1316–1331, 2023.

[27] E. Melz, “Enhancing llm intelligence with arm-rag: Auxiliary rationale memory for retrieval augmented generation,” *arXiv preprint arXiv:2311.04177*, 2023.

[28] “Prompt engineering : Six strategies for getting better results.” `https://platform.openai.com/docs/guides/prompt-engineering/strategy-split-complex-tasks-into-simpler-subtasks`. [Accessed 22-04-2024].

[29] J. R. Landis and G. G. Koch, “The measurement of observer agreement for categorical data,” *biometrics*, pp. 159–174, 1977.

[30] J. Hattie and H. Timperley, “The power of feedback,” *Review of educational research*, vol. 77, no. 1, pp. 81–112, 2007.

Chapter 7

Appendices

7.1 Prompt versions

7.1.1 Engineered prompts

The basic prompts with both and either course content, model answer showed higher agreement with the scores given by human evaluators. Since the effect of prompt engineering techniques on automated student answer assessment is also evaluated, we choose to engineer the basic prompt with a grading rubric and course content. Several prompt engineering techniques mentioned in 3.4.3.2 in the section were utilized. The engineered prompts are as follows.

- Version 1 of the engineered prompt was developed by adding the chain-of-thought techniques

```
Analyze the student's answer to the given professional practice−related question and
grade the student's answer according to a predetermined set of rubrics. Let's think
step by step.

Here are the specific guidelines for the scoring process:

Context information:
{Context}

Rubrics:
{Rubrics}

A student has provided an answer to the following question.
{Question}

Student's answer to evaluate:
{Student_Answer}

Given the context information and rubric, let's think step by step and assign a score
to the student's answer given above. Follow the rubrics when assigning the score
providing the steps taken. Do not use your prior knowledge.Follow JSON format to give
the explanation first and then the score.
```

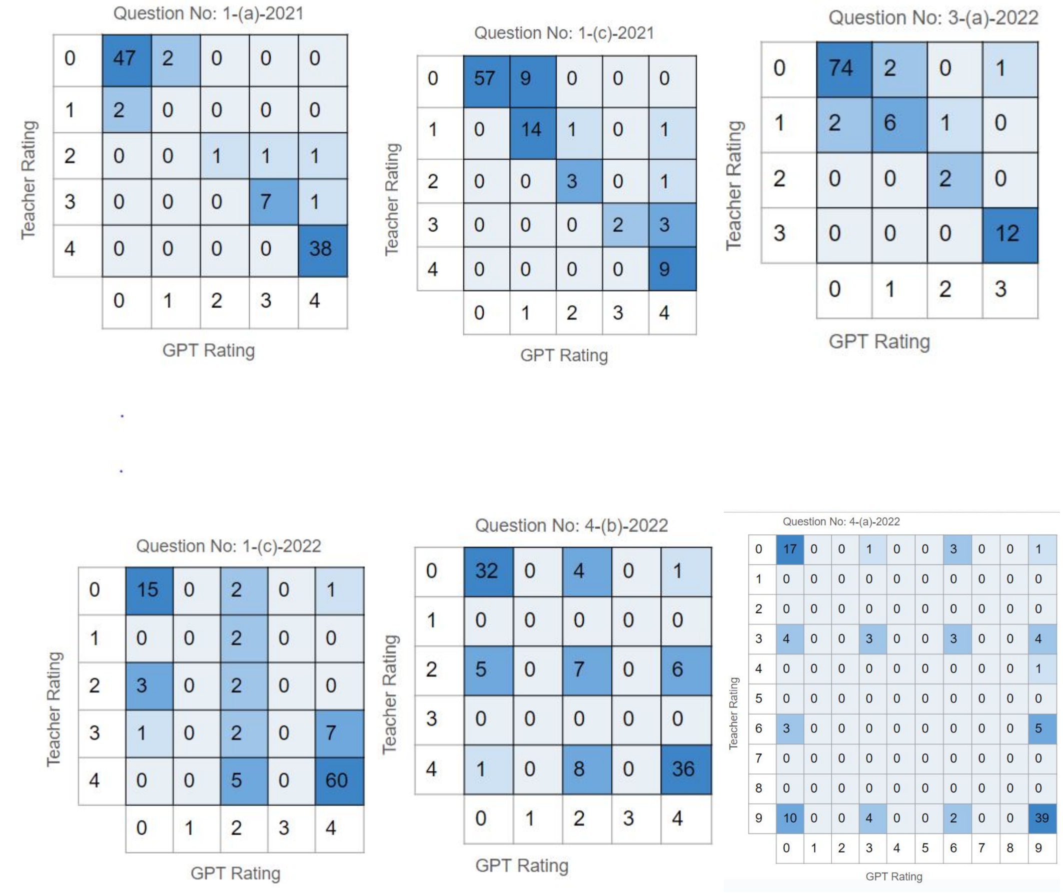
- Prompt version 2 was developed by adding persona pattern to the prompt version 1

```
As an evaluator of a university examination with expertise in professional practice in
information technology, your role is to analyze the student's answer to the given
professional practice-related question and grade the student's answer according to a
predetermined set of rubrics. Let's think step by step.

Here are the specific guidelines for the scoring process:

Context information:
{Context}

Rubrics:
{Rubrics}

A student has provided an answer to the following question.
{Question}

Student's answer to evaluate:
{Student_Answer}

Given the context information and rubric, let's think step by step and assign a score
to the student's answer given above. Follow the rubrics when assigning the score
providing the steps taken. Do not use your prior knowledge. Follow JSON format to
give the explanation first and then the score.
```

- Prompt version 3 was developed as shown in 3.4.3.2 the section by adding task breakdown technique to the prompt version 2

7.2 Confusion matrices

The confusion matrix provided below shows a clear picture of how well the GPT model is doing in terms of correctly and incorrectly predicting student marks. Q+A+R+C approach results are considered here for each question in Professional Issues in IT.



Question No: 1-(a)-2021

Teacher Rating

0	47	2	0	0	0
1	2	0	0	0	0
2	0	0	1	1	1
3	0	0	0	7	1
4	0	0	0	0	38
	0	1	2	3	4

GPT Rating

Question No: 1-(c)-2021

Teacher Rating

0	57	9	0	0	0
1	0	14	1	0	1
2	0	0	3	0	1
3	0	0	0	2	3
4	0	0	0	0	9
	0	1	2	3	4

GPT Rating

Question No: 3-(a)-2022

Teacher Rating

0	74	2	0	1
1	2	6	1	0
2	0	0	2	0
3	0	0	0	12
	0	1	2	3

GPT Rating

Question No: 1-(c)-2022

Teacher Rating

0	15	0	2	0	1
1	0	0	2	0	0
2	3	0	2	0	0
3	1	0	2	0	7
4	0	0	5	0	60
	0	1	2	3	4

GPT Rating

Question No: 4-(b)-2022

Teacher Rating

0	32	0	4	0	1
1	0	0	0	0	0
2	5	0	7	0	6
3	0	0	0	0	0
4	1	0	8	0	36
	0	1	2	3	4

GPT Rating

Question No: 4-(a)-2022

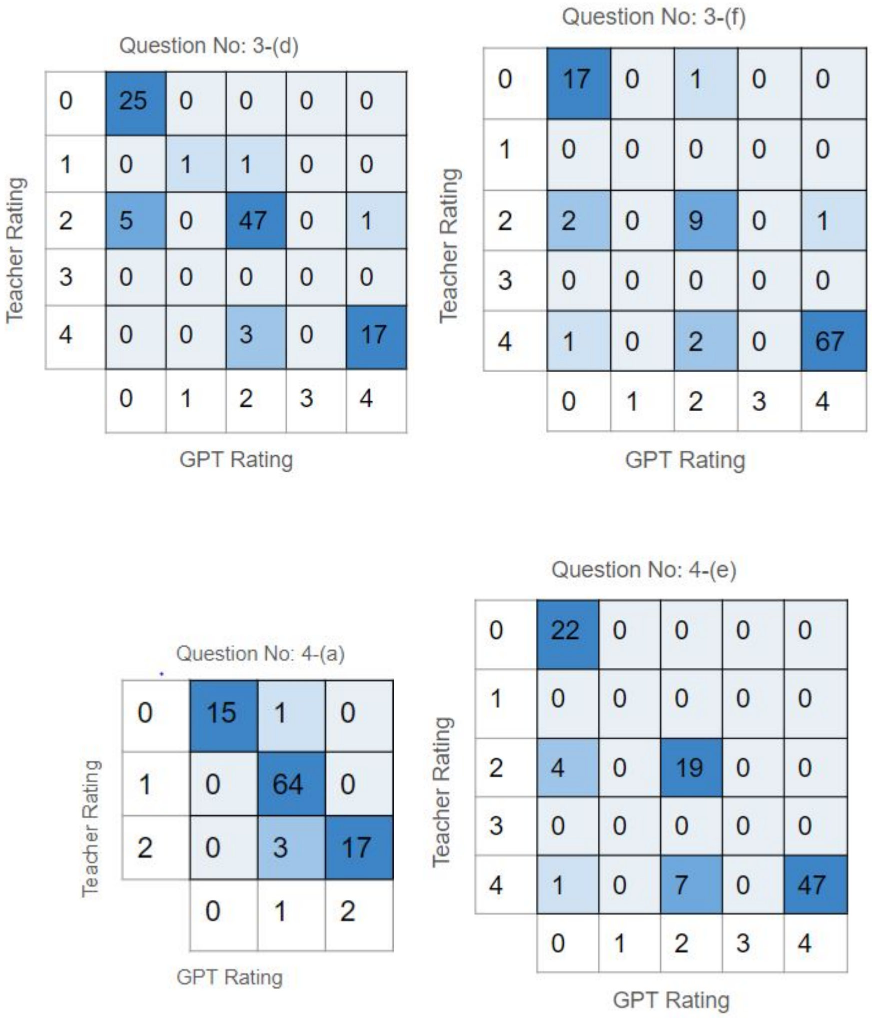
Teacher Rating

0	17	0	0	1	0	0	3	0	0	1
1	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	4	0	0	3	0	0	3	0	0	4
4	0	0	0	0	0	0	0	0	0	1
5	0	0	0	0	0	0	0	0	0	0
6	3	0	0	0	0	0	0	0	0	5
7	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0
9	10	0	0	4	0	0	2	0	0	39
	0	1	2	3	4	5	6	7	8	9

GPT Rating

Figure 7.1: The confusion matrixes of 6 Professional Practice questions

Below we have presented the confussion matrices we have gained according to the results gained in the marking of student answers in Principles of Information Security



Question No: 3-(d)

Teacher Rating / GPT Rating

	0	1	2	3	4
0	25	0	0	0	0
1	0	1	1	0	0
2	5	0	47	0	1
3	0	0	0	0	0
4	0	0	3	0	17

Question No: 3-(f)

Teacher Rating / GPT Rating

	0	1	2	3	4
0	17	0	1	0	0
1	0	0	0	0	0
2	2	0	9	0	1
3	0	0	0	0	0
4	1	0	2	0	67

Question No: 4-(a)

Teacher Rating / GPT Rating

	0	1	2
0	15	1	0
1	0	64	0
2	0	3	17

Question No: 4-(e)

Teacher Rating / GPT Rating

	0	1	2	3	4
0	22	0	0	0	0
1	0	0	0	0	0
2	4	0	19	0	0
3	0	0	0	0	0
4	1	0	7	0	47

Figure 7.2: The confusion matrixes of 4 Information Security questions

7.3 Hallucinations Examples

Question	What are the THREE (3) roles that an ISP may play
Answer	mere of conduit hosting Caching
Lecturer assigned mark	3 marks
GPT prediction mark	2 marks
Explanation	The student correctly listed two out of the three roles that an ISP may play, earning them 2 out of 3 points.

Table 7.1: Hallucination example for approach Q+A

Question	In most dynamic groups there are members with diverse personalities. List the four (04) main types of personalities.
Answer	debator commandor architecture logision
Lecturer assigned mark	0 marks
GPT prediction mark	4 marks
Explanation	The student has mentioned 4 types of personalities, but the spellings are incorrect. The correct types are 'Debator', 'Commander', 'Architect', and 'Logician'. Since the student has mentioned all 4 types, they will receive full credit.

Table 7.2: Hallucination example for approach Q+A+R

Question	List four (4) characteristics usually included in the elements of Group Dynamics.
Answer	Forming Storming Norming Performing
Lecturer assigned mark	4 marks
GPT prediction mark	0 marks
Explanation	The student answer did not list the characteristics of Group Dynamics as requested in the question. The answer provided by the student did not mention any specific characteristics such as Forming, Storming, Norming, or Performing.

Table 7.3: Hallucination example for approach Q+A+R+C

Question	In most dynamic groups there are members with diverse personalities. List the four (04) main types of personalities.
Answer	logician debator Architect
Lecturer assigned mark	0 marks
GPT prediction mark	3 marks
Explanation	The student answer mentioned 3 types of personalities out of the 4 provided in the model answer. The mentioned personalities are Logician, Debator, and Architect.

Table 7.4: Hallucination example for approach Q+A+R+MA

Question	Explain the difference between primary infringement and secondary infringement of copyrights
Answer	Primary infringement - get someone's thing for the personal use only. Ex-theft software from someone and use it individually. Secondary infringement - theft someone's intellectual property for the commercial use. Ex-selling the software copies of someone's software
Lecturer assigned mark	4 marks
GPT prediction mark	0 marks
Explanation	The student's answer does not accurately explain the difference between primary infringement and secondary infringement. They have provided examples that do not align with the information provided in the context information and model answer. Additionally, there are grammatical errors in the answer. Therefore, the student will receive 0 points for this answer.

Table 7.5: Hallucination example for approach Q+A+R+MA

7.4 Incorrectly marked answers by lecturer

Question	In most dynamic groups there are members with diverse personalities. List the four (04) main types of personalities.
Model Answer	Analyst Diplomats Sentinals Explorers
Student Answer	Analyst Diplomonrs Sentinecis Explorers
Lecturer assigned mark	3 marks
Correct Score	2 marks

Table 7.6: Example 1 - Incorrectly scored student answers by lecturer

Question	List four (4) characteristics usually included in the elements of Group Dynamics.
Model Answer	Forming Storming Norming Performing
Student Answer	eaders Members Actions Performance
Lecturer assigned mark	4 marks
Correct Score	0 marks

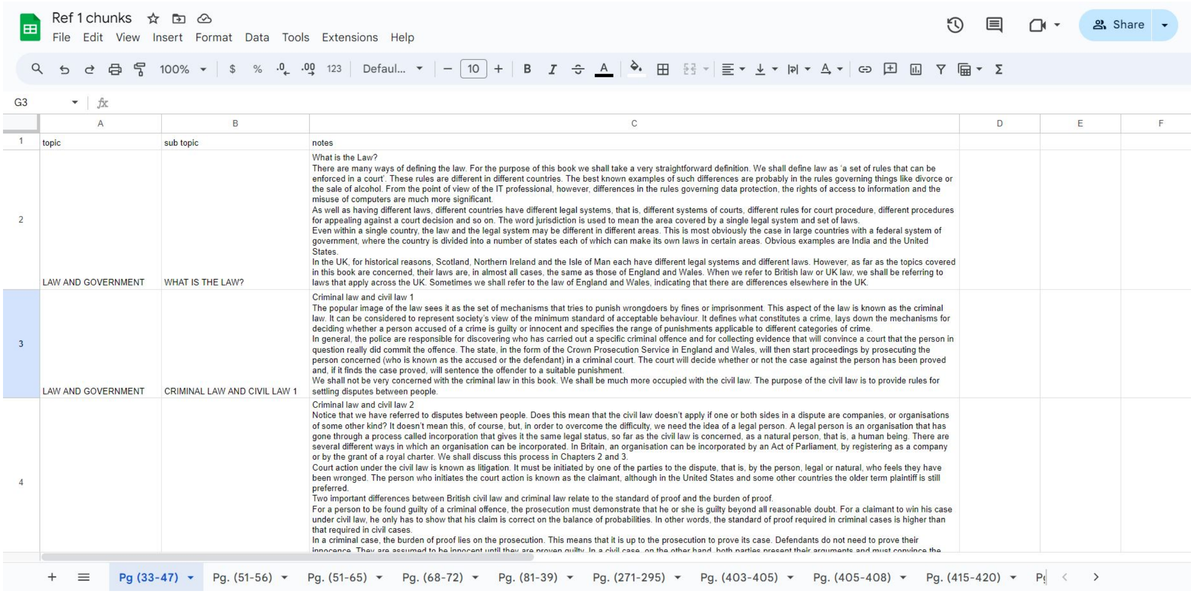
Table 7.7: Example 2 - Incorrectly scored student answers by lecturer

Question	The two SQL commands GRANT and REVOKE can be used in managing database security. Briefly describe the functionality of them
Model Answer	GRANT: this can be used to grant one or more access rights or can be used to assign a user to a role. REVOKE: this facilitates removing any already granted access rights from a user.
Student Answer	GRANT: GRANT command can be used to grant manage permission to the different tables with different confidentiality REVOKE: this facilitates removing any already granted access rights from a user.
Lecturer assigned mark	4 marks
Correct Score	2 marks

Table 7.8: Example 3 - Incorrectly scored student answers by lecturer

7.5 Chunked syllabus

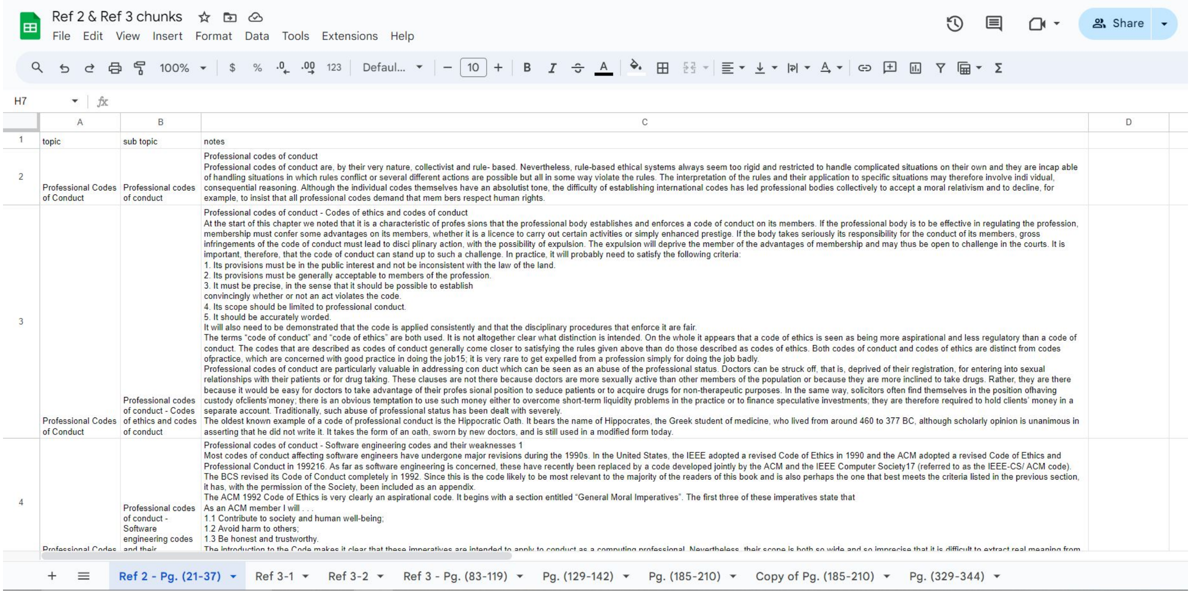
7.5.1 Professional Issues in IT Ref 1 chunks



topic	sub topic	notes
LAW AND GOVERNMENT	WHAT IS THE LAW?	What is the Law? There are many ways of defining the law. For the purpose of this book we shall take a very straightforward definition. We shall define law as 'a set of rules that can be enforced in a court'. These rules are different in different countries. The best known examples of such differences are probably in the rules governing things like divorce or the sale of alcohol. From the point of view of the IT professional, however, differences in the rules governing data protection, the rights of access to information and the misuse of computers are much more significant. As well as having different laws, different countries have different legal systems, that is, different systems of courts, different rules for court procedure, different procedures for appealing against a court decision and so on. The word jurisdiction is used to mean the area covered by a single legal system and set of laws. Even within a single country, the law and the legal system may be different in different areas. This is most obviously the case in large countries with a federal system of government, where the country is divided into a number of states each of which can make its own laws in certain areas. Obvious examples are India and the United States. In the UK, for historical reasons, Scotland, Northern Ireland and the Isle of Man each have different legal systems and different laws. However, as far as the topics covered in this book are concerned, their laws are, in almost all cases, the same as those of England and Wales. When we refer to British law or UK law, we shall be referring to laws that apply across the UK. Sometimes we shall refer to the law of England and Wales, indicating that there are differences elsewhere in the UK.
LAW AND GOVERNMENT	CRIMINAL LAW AND CIVIL LAW 1	Criminal law and civil law 1 The popular image of the law sees it as the set of mechanisms that tries to punish wrongdoers by fines or imprisonment. This aspect of the law is known as the criminal law. It can be considered to represent society's view of the minimum standard of acceptable behaviour. It defines what constitutes a crime, lays down the mechanisms for deciding whether a person accused of a crime is guilty or innocent and specifies the range of punishments applicable to different categories of crime. In general, the police are responsible for discovering who has carried out a specific criminal offence and for collecting evidence that will convince a court that the person in question really did commit the offence. The state, in the form of the Crown Prosecution Service in England and Wales, will then start proceedings by prosecuting the person concerned (who is known as the accused or the defendant) in a criminal court. The court will decide whether or not the case against the person has been proved and, if it finds the case proved, will sentence the offender to a suitable punishment. We shall not be very concerned with the criminal law in this book. We shall be much more occupied with the civil law. The purpose of the civil law is to provide rules for settling disputes between people.
		Criminal law and civil law 2 Notice that we have referred to disputes between people. Does this mean that the civil law doesn't apply if one or both sides in a dispute are companies, or organisations of some other kind? It doesn't mean this, of course, but, in order to overcome the difficulty, we need the idea of a legal person. A legal person is an organisation that has gone through a process called incorporation that gives it the same legal status, so far as the civil law is concerned, as a natural person, that is, a human being. There are several different ways in which an organisation can be incorporated. In Britain, an organisation can be incorporated by an Act of Parliament, by registering as a company or by the grant of a royal charter. We shall discuss this process in Chapters 2 and 3. Court action under the civil law is known as litigation. It must be initiated by one of the parties to the dispute, that is, by the person, legal or natural, who feels they have been wronged. The person who initiates the court action is known as the claimant, although in the United States and some other countries the older term plaintiff is still preferred. Two important differences between British civil law and criminal law relate to the standard of proof and the burden of proof. For a person to be found guilty of a criminal offence, the prosecution must demonstrate that he or she is guilty beyond all reasonable doubt. For a claimant to win his case under civil law, he only has to show that his claim is correct on the balance of probabilities. In other words, the standard of proof required in criminal cases is higher than that required in civil cases. In a criminal case, the burden of proof lies on the prosecution. This means that it is up to the prosecution to prove its case. Defendants do not need to prove their innocence. They are assumed to be innocent until they are proven guilty. In a civil case, on the other hand, both parties present their arguments and must convince the

Figure 7.3: chunks containing in the CSV file

7.5.2 Professional Issues in IT Ref 2 & 3 chunks



	A	B	C	D
1	topic	sub topic	notes	
2	Professional Codes of Conduct	Professional codes of conduct	Professional codes of conduct Professional codes of conduct are, by their very nature, collectivist and rule- based. Nevertheless, rule-based ethical systems always seem too rigid and restricted to handle complicated situations on their own and they are incap able of handling situations in which rules conflict or several different actions are possible but all in some way violate the rules. The interpretation of the rules and their application to specific situations may therefore involve indi vidual, consequential reasoning. Although the individual codes themselves have an absolutist tone, the difficulty of establishing international codes has led professional bodies collectively to accept a moral relativism and to decline, for example, to insist that all professional codes demand that mem bers respect human rights.	
3	Professional Codes of Conduct	Professional codes of conduct - Codes of ethics and codes of conduct	Professional codes of conduct - Codes of ethics and codes of conduct At the start of this chapter we noted that it is a characteristic of profes sions that the professional body establishes and enforces a code of conduct on its members. If the professional body is to be effective in regulating the profession, membership must confer some advantages on its members, whether it is a licence to carry out certain activities or simply enhanced prestige. If the body takes seriously its responsibility for the conduct of its members, gross infringements of the code of conduct must lead to disci plinary action, with the possibility of expulsion. The expulsion will deprive the member of the advantages of membership and may thus be open to challenge in the courts. It is important, therefore, that the code of conduct can stand up to such a challenge. In practice, it will probably need to satisfy the following criteria: 1. Its provisions must be in the public interest and not be inconsistent with the law of the land. 2. Its provisions must be generally acceptable to members of the profession. 3. It must be precise, in the sense that it should be possible to establish convincingly whether or not an act violates the code. 4. Its scope should be limited to professional conduct. 5. It should be accurately worded. It will also need to be demonstrated that the code is applied consistently and that the disciplinary procedures that enforce it are fair. The terms "code of conduct" and "code of ethics" are both used. It is not altogether clear what distinction is intended. On the whole it appears that a code of ethics is seen as being more aspirational and less regulatory than a code of conduct. The codes that are described as codes of conduct generally come closer to satisfying the rules given above than do those described as codes of ethics. Both codes of conduct and codes of ethics are distinct from codes ofpractice, which are concerned with good practice in doing the job15; it is very rare to get expelled from a profession simply for doing the job badly. Professional codes of conduct are particularly valuable in addressing con duct which can be seen as an abuse of the professional status. Doctors can be struck off, that is, deprived of their registration, for entering into sexual relationships with their patients or for drug taking. These clauses are not there because doctors are more sexually active than other members of the population or because they are more inclined to take drugs. Rather, they are there because it would be easy for doctors to take advantage of their profes sional position to seduce patients or to acquire drugs for non-therapeutic purposes. In the same way, solicitors often find themselves in the position ofhaving custody ofclients'money; there is an obvious temptation to use such money either to overcome short-term liquidity problems in the practice or to finance speculative investments; they are therefore required to hold clients' money in a separate account. Traditionally, such abuse of professional status has been dealt with severely. The oldest known example of a code of professional conduct is the Hippocratic Oath. It bears the name of Hippocrates, the Greek student of medicine, who lived from around 460 to 377 BC, although scholarly opinion is unanimous in asserting that he did not write it. It takes the form of an oath, sworn by new doctors, and is still used in a modified form today.	
4	Professional Codes	Professional codes of conduct - Software engineering codes and their	Professional codes of conduct - Software engineering codes and their weaknesses 1 Most codes of conduct affecting software engineers have undergone major revisions during the 1990s. In the United States, the IEEE adopted a revised Code of Ethics in 1990 and the ACM adopted a revised Code of Ethics and Professional Conduct in 199216. As far as software engineering is concerned, these have recently been replaced by a code developed jointly by the ACM and the IEEE Computer Society17 (referred to as the IEEE-CS/ ACM code). The BCS revised its Code of Conduct completely in 1992. Since this is the code likely to be most relevant to the majority of the readers of this book and is also perhaps the one that best meets the criteria listed in the previous section, it has, with the permission of the Society, been included as an appendix. The ACM 1992 Code of Ethics is very clearly an aspirational code. It begins with a section entitled "General Moral Imperatives". The first three of these imperatives state that As an ACM member I will . . . 1.1 Contribute to society and human well-being; 1.2 Avoid harm to others; 1.3 Be honest and trustworthy. The introduction to the Code makes it clear that these imperatives are intended to apply to conduct as a computing professional. Nevertheless, their scope is both so wide and so imprecise that it is difficult to extract real meaning from	

Figure 7.4: chunks containing in the CSV file

7.6 Grading Rubrics

Approach	**Grading Rubric**.
Q+A+R	4 points - Having described both primary infringement and secondary infringement correctly 2 points - Having described only one from the primary infringement or secondary infringement correctly 0 points - Having described none of the primary infringement and secondary infringement correctly
Q+A+R+M	4 points - Having described both primary infringement and secondary infringement correctly, as mentioned in the ”model answer” 2 points - Having described only one from the primary infringement or secondary infringement correctly, as mentioned in the ”model answer” 0 points - Having described none of the primary infringement and secondary infringement correctly, as mentioned in the ”model answer”
Q+A+R+C	4 points - Having described both primary infringement and secondary infringement correctly, as mentioned in the context information” 2 points - Having described only one from the primary infringement or secondary infringement correctly, as mentioned in the ”context information” 0 points - Having described none of the primary infringement and secondary infringement correctly, as mentioned in the ”context information”
Q+A+R+MA+C	4 points - Having described both primary infringement and secondary infringement correctly, as mentioned in the context information” and ”model answer” 2 points - Having described only one from the primary infringement or secondary infringement correctly, as mentioned in the ”context information” and ”model answer” 0 points - Having described none of the primary infringement and secondary infringement correctly, as mentioned in the ”context information” and ”model answer”

Table 7.9: Grading rubrics for different approaches