Automated Assignment Scoring Via Azure OpenAI ChatGPT

By: Callum Fry

Project Unit: Your Project Unit Code Here!
Supervisor:

Submission date:
7 May 2024

1 Introduction

1.1 Context of Research

Machine Learning (ML) is an evolving branch of Artificial Intelligence (AI) that uses computational algorithms aimed to imitate human intelligence by using data from its environment to learn and improve (El Naqa et al., 2015). In more recent years, machine learning has become more widely used; this has been driven by the development of new theories and practical applications. As well as the rapid increase in accessible online data and affordable computing power (M., 2015).

In this landscape, cloud platforms like Microsoft Azure have emerged as powerful tools for deploying AI models. Specifically, Microsoft Azure is a public cloud computing platform developed by Microsoft (Borra, 2024). This product provides a comprehensive range of services but the most relevant for this project is Azure AI Studio, this provides developers with the ability to create an AI model and deploy it in a secure environment.

In the United Kingdom, most universities currently mark dissertations by selecting two members of staff and allowing them to mark the assignment without discussing it. However, this method may be outdated, and replacing one marker for an AI could be possible, or adding a third marker in the form of AI.

Aiding lectures with this process is likely to have numerous benefits, because markers experience stress, anxiety, and other mental health challenges during periods of heavy assignment marking (Henderson-Brooks, n.d.). The benefits include saving time and resources, improved feedback, more efficient marking, and removal of human bias.

These benefits allow markers more time to improve the quality of their teaching while allowing students to review feedback to further enhance their skills for future work.

1.2 Project Aim

This paper aims to enhance both the quality and speed of assignment marking by implementing automation tools for grading assignments. By implementing AI and machine learning models using cloud solutions. Reducing the time and effort required for manual grading, the proposed solution seeks to relieve the workload on lecturers while improving feedback for students.

1.3 Project Objectives

The objectives of this project are:

1. Gain an understanding of the current research in this area, and existing tools to produce a literature review.

2. Meeting staff to gather requirements, therefore determining what the project needs to achieve. Interviewing lecturers to analyse the requirements to be able to prioritise which features are necessary and which are not.

3. Developing architecture design and sequence diagrams to be able to visualise a design that can be implemented.

4. Implement an accurate and efficient ML model that can mark assignments and produce grades. Based on an input of a marking scheme and student assignments.

5. Evaluate the effectiveness of the model by comparing the results produced to already marked assignments.

2 Literature Review

2.1 Research Background, Context, and Definitions

To fully understand this chapter, some meanings are defined here:

Artificial Intelligence (AI): computer systems that are capable of completing tasks that would require human intelligence (McCarthy, 2004).

Machine Learning (ML): a subset of AI that focuses on developing algorithms to enable computers to learn from their environment and datasets (El Naqa et al., 2015).

Dataset: a collection of data that can be either structured or unstructured, used for training or testing ML models. These datasets can be imported into Azure ML studio and used through the implementation of the model.

Hyperparameters: these are configurable parameters to be able to guide the learning of a model. An example of this could be the learning rate of a neural network, allowing for better optimization (Arnold et al., n.d.).

This literature review aims to research existing knowledge, solutions, and developments in this area. Critical analysis and investigations of already existing solutions to similar tasks will be completed to understand the most appropriate approach.

This area of research is important at this time because the trends of machine learning are starting to evolve in the education sector, a main sector of this is the grading system. By leveraging machine learning, the grading system can be reframed to encourage a more efficient and accurate system (Jalil et al., 2019).

Looking deeper into saving time, some initial research was completed on comparing an automated system against an instructor. Grading a select number of students took the instructor four hours and eleven minutes, averaging four minutes and ten seconds per student. In comparison, the automated system took fourteen seconds to mark the same number of students (Bian et al., 2020). Therefore, the educational system is starting to show promising applications for machine learning, which has increased the trend of research into this area.

There are three main sections of research for this field being, machine learning, artificial intelligence and applications for machine learning in the education sector. Research and popularity into artificial intelligence has increased significantly, since 2013 the share of research papers with titles or abstracts that mention AI or ML has increased from 10% to 27% (Van Noorden & Perkel, 2023).
Some initial research was completed into the application of using a ChatGPT model through Azure to be able to grade students' assignments. This preliminary solution was found to mark 70 students' long essay assignments in ten minutes, whereas previously it took a few hours to manually mark.

Another added benefit found was they had more time to improve the quality of their teaching instead of spending this time on marking the assignments. They spent more time planning, learning and developing their skills for their upcoming lessons.

2.2 Similar Works

- Automation of checking student assignments in IT-related subjects based on AI systems (Sharyhin & Klochko, 2024)
- Fine-tuning ChatGPT for automatic scoring - ScienceDirect
- Automated assignment grading using Azure & ChatGPT - Azure OpenAI, Chat GPT, Automated Assignment Scoring

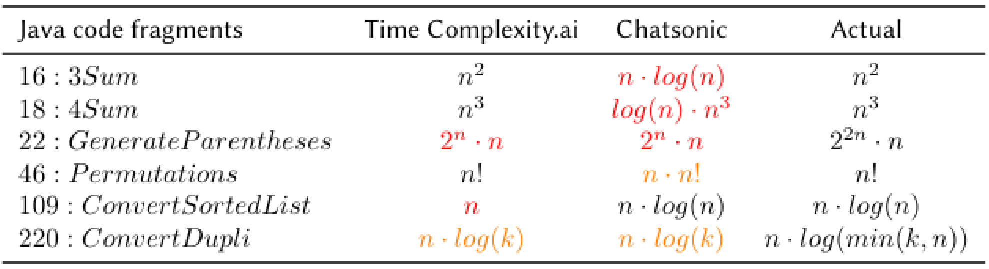
2.2.1 Automation of checking students' assignments in IT

(Sharyhin & Klochko, 2024) uses AI systems to analyse the time complexity of code segments submitted by students. It discusses methods for submissions, reviewing and feedback to support teachers with large amounts of grading.

The strengths of their project are efficiency, immediate feedback, scalability, and adaptability. The automation can efficiently check solutions, providing relevant feedback to students on if they need to improve their code or if it is correct. The models used also make it scalable and adaptable, with abilities of understanding multiple programming languages and customizable by the teachers to meet certain requirements.

However, the downsides are initial setup complexity, overreliance, consistency, and correctness. To initially configure the layers and models takes a large amount of time and investment, this may put some users off due to the time commitment. Another issue that occurs with AI is overreliance, they become so focused on system-specific optimization that they become very strict and forget about real-world problem-solving. The consistency and correctness of the results are not fully accurate as shown in Figure 1 the models occasionally fail to produce the correct results which will result in correct submissions being rejected.

Results of time complexity assessment of selected Java code fragments for different AI systems.



Java code fragments	Time Complexity.ai	Chatsonic	Actual
$16 : 3Sum$	n^2	$n \cdot log(n)$	n^2
$18 : 4Sum$	n^3	$log(n) \cdot n^3$	n^3
$22 : GenerateParentheses$	$2^n \cdot n$	$2^n \cdot n$	$2^{2n} \cdot n$
$46 : Permutations$	$n!$	$n \cdot n!$	$n!$
$109 : ConvertSortedList$	n	$n \cdot log(n)$	$n \cdot log(n)$
$220 : ConvertDupli$	$n \cdot log(k)$	$n \cdot log(k)$	$n \cdot log(min(k, n))$

Figure 1. Time Complexity of Java code: Showing two models compared to the actual answer (Sharyhin & Klochko, 2024).

2.2.1.1 Features

In an automated evaluation, the project determines the time complexity of a code segment, if this meets the requirements it allows for the student to submit. If the code does not meet the optimal solution, then the student may try again until they provide the correct solution.

In feedback generation, the AI provides constructive feedback, especially if the optimal solution is not met. It provides messages when calculating the time complexity, showing students where the complexity is derived from. The AI could also provide suggestions to improve efficiency.

Customizability, the project can understand different programming languages, which allows many assignments to be checked. The learning objectives can be adapted depending on the time complexity that needs to be met for different coding segments.

2.2.1.2 Architecture

The project contains four main layers of architecture, inputs, analysis, feedback, and customizability. These four layers represent different parts of the project which can be adjusted and improved for different situations.

The input layer is where students submit their code segments, this interface simply passes the input to the analysis layer. The analysis layer uses AI models such as ChatGPT or Time Complexity.ai to evaluate the solution submitted. The feedback layer takes the results from the AI models, based on the optimal solution and detected errors, and produces responses to the student whether positive or negative. The customizability layer is a separate layer in which teachers can provide criteria and parameters for the students' programs to meet.

2.2.2 Automated assignment grading using Azure & ChatGPT

Their project researches how AI can streamline the grading process for lectures using the Azure OpenAI environment and ChatGPT. Specifically grading essays submitted by students in Microsoft Word and Adobe PDF, the models provide a grade, feedback, marks and potential plagiarism from copying or generative AI.

The strengths of their project are efficiency, scalability, customisation, and feedback. The model can grade 70 students' essays in under ten minutes, costing only $0.5 for Azure’s cloud computing services. The project is scalable because it can review multiple essays simultaneously while being customisable, so the lectures can adjust the marking scheme on which the essays are graded.

The data privacy concerns are raised because there is not enough cleaning of the data, this will be learnt to improve the upcoming projects. Due to assignments containing private information such as full names, student numbers and possibly more, they are not necessarily needed for the AI to grade, so such should be cleaned. This takes teachers time to identify and remove these unnecessary pieces of data.

2.2.2.1 Features

This AI model produces very clear and informative feedback, using a different technique in comparison to other projects. The feedback is created by using prompt engineering techniques, which tell the AI to create a message by acting. For example, proving the prompt “Take on the role of a teacher by assigning marks and providing constructive comments for a writing assignment” to the AI. Many of these prompts are inputted to provide stricter outputs with a higher probability of success.

A main feature is using Azure's cloud services, which provide many advantages over a locally based system, such as security, integration, reliability, and customisation of models. Azure's security and reliability mean that uptime and robust support will be optimized, while the infrastructure of Azure provides more security to the data being processed.

Integrating projects with Azure is a good idea because it allows for feature improvements to be made with other Azure products that help with storage, monitoring, or analytics. The analytics would prove how effective the system is to be able to upgrade the model or for proof of product.

2.2.2.2 Architecture

The architecture flow of their project is: collecting the inputs (assignments and marking scheme), preprocessing data, model review, feedback creation, and output delivery.

The input collection is completed manually, which is a weakness of their project, as spoken about before. The data is then preprocessed, this is to ensure that the essays are in the correct file type and in the correct folder structure to be processed by the model. This will help to remove errors since many students submit essays in different ways.

The model then reviews the essays submitted, produces feedback from them and creates an output for the lecture to be checked by a human and sent to students. Azure AI Studio provides a playground for users to check the performance of a model before deploying it. Using different evaluation tools to review the model and apply a rubric-based scoring.

The feedback is then produced by using prompt engineering to provide personalised constructive feedback. Just like natural human feedback, the tailored responses are designed to help students see where they did well and areas of improvement, to highlight gaps of knowledge. The evaluation grade and feedback are then to be reviewed by a human before sending back out to the class.

2.2.3 Fine-tuning ChatGPT for automatic scoring

(Latif & Zhai, 2024) explores the potential of fine-tuning ChatGPT-3.5 for automated assignment grading and compares its results with BERT (Bidirectional Encoder Representations from Transformers). This is to better understand how automated grading systems can be implemented and which models may be superior. The study highlights the potential benefits while digging into some concerns.

The strengths and weaknesses of using ChatGPT for automated grading have been highlighted, offering valuable insights into its potential applications and limitations. The strength their project shows is how ChatGPT outperforms BERT, which is very important knowledge for selecting the correct models to base the upcoming project on.

ChatGPT shows an average of accuracy 9.1% higher than BERT, this is a significant difference and can be further shown in Figure 2. A range of different questions were given to the models, represented by the names on the X axis of Figure 2. In some questions, such as Gas-filled balloons, no difference was noted, which showed the models were equally capable of identifying this question.

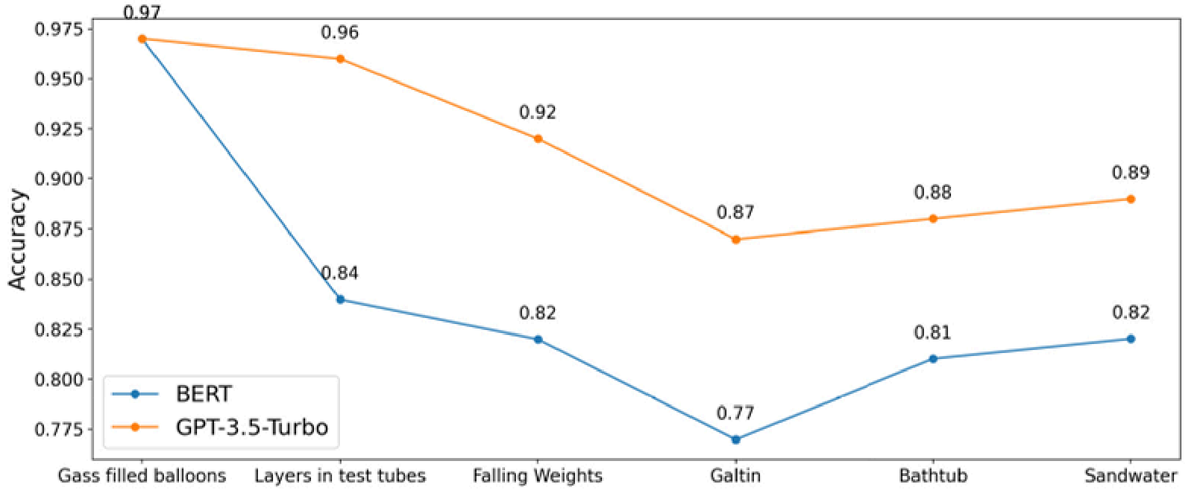


Figure 2. Accuracy comparison of fine-tuned ChatGPT and BERT for different assessment tasks (Latif & Zhai, 2024).

However, their project does not account for bias, fairness, and ethical implications. While also only comparing ChatGPT to BERT when some newer AI models such as Google's Gemini or Bard could be used, these models could provide better accuracy for automated grading.

2.2.3.1 Features

The features their project presents are data collection and preprocessing, fine-tuning and validations. These three main features are to ensure that the models are ready for grading assessment questions correctly.

The data is cleaned during preprocessing, which removes any irrelevant data or personally identifiable information. This is a necessity which should be carried out in the upcoming project to ensure student privacy. The data is then tokenized, so the AI model understands the data in a suitable format. However, the volume of data has not been taken into account which could cause issues when tokenizing large amounts of data.

The fine-tuning adjusts the learning rate, epochs, and batch size, which are all equally important in optimizing the accuracy of the AI model. A loss function is used, which measures how well the model is learning, quantifying this statistic allows for further fine-tuning of the model. Once fine-tuned, the model is validated using a separate evaluation set to access the performance. These are some interesting processes which will highly advance the performance of the AI model.

2.2.3.2 Architecture

The architecture of their project is collecting the data, preprocessing, model initialization, fine-tuning procedures, evaluation and validation, and baselines. An influential architectural step which has been included in their project is the fine-tuning procedures. This phase takes an already intelligent model in ChatGPT and turns it into a more functional model that can more accurately grade assessments.

2.3 Summary

As part of the reviews conducted in this chapter, several points arose that we would address in our requirements.

An automated assignment grading system needs to be backed by a large cloud provider such as Azure, as discussed in 2.2.2. The reliability that Azure provides means that the project is more stable, it also allows for many extra improvements and services to be added such as monitoring methods.

Another issue identified in this literature review was the need to clean/preprocess assignments to remove unnecessary data such as student ID's. Removing such data is a necessity due to it having no impact on the AI's result but could carry some heavy privacy and security issues.

In 2.2.2 we will now be implementing prompt engineering into the project to allow the feedback layer to be more accurate. This will help to provide better feedback to students that feels more realistic and gives insights into mistakes.

The design of the project has been improved since reviewing these papers due to having two new layers brought to my attention. These are the fine-tuning layer and validation layer, the fine-tuning layer is very important to improve the accuracy of the AI as shown in 2.2.3. The validation layer is critical because it allows us to check that the fine-tuning is making the model more accurate, when trying to improve a model some issues may occur such as overfitting. These can be resolved easily if the problem is identified by a validation layer, this layer can also help to prove the effectiveness of the AI, making it more appealing to users.

2.3.1 Comparison

All of these sources reviewed provided different learnings, with distinctive strengths and weaknesses. Some significant insights gained were for privacy, architecture, models, and feedback.

They all specialise in different areas of automation, however, they all highlight the need for data privacy. When collecting assignments from students, the data must be cleaned. This is because assignments can contain data such as student numbers, full names, and sensitive data. Properly cleaning the data will remove these points, since they are unnecessary for the AI to learn and grade papers.

These sources also express the need for customisation in different cases. The AI models should be able to learn different programming languages for example, this is needed because different assignments may contain different programming languages. The customisation should also allow markers to input different mark schemes, different assignments will need different marking schemes for the AI to process.

5 Design

5.1 Architecture design/Component diagram

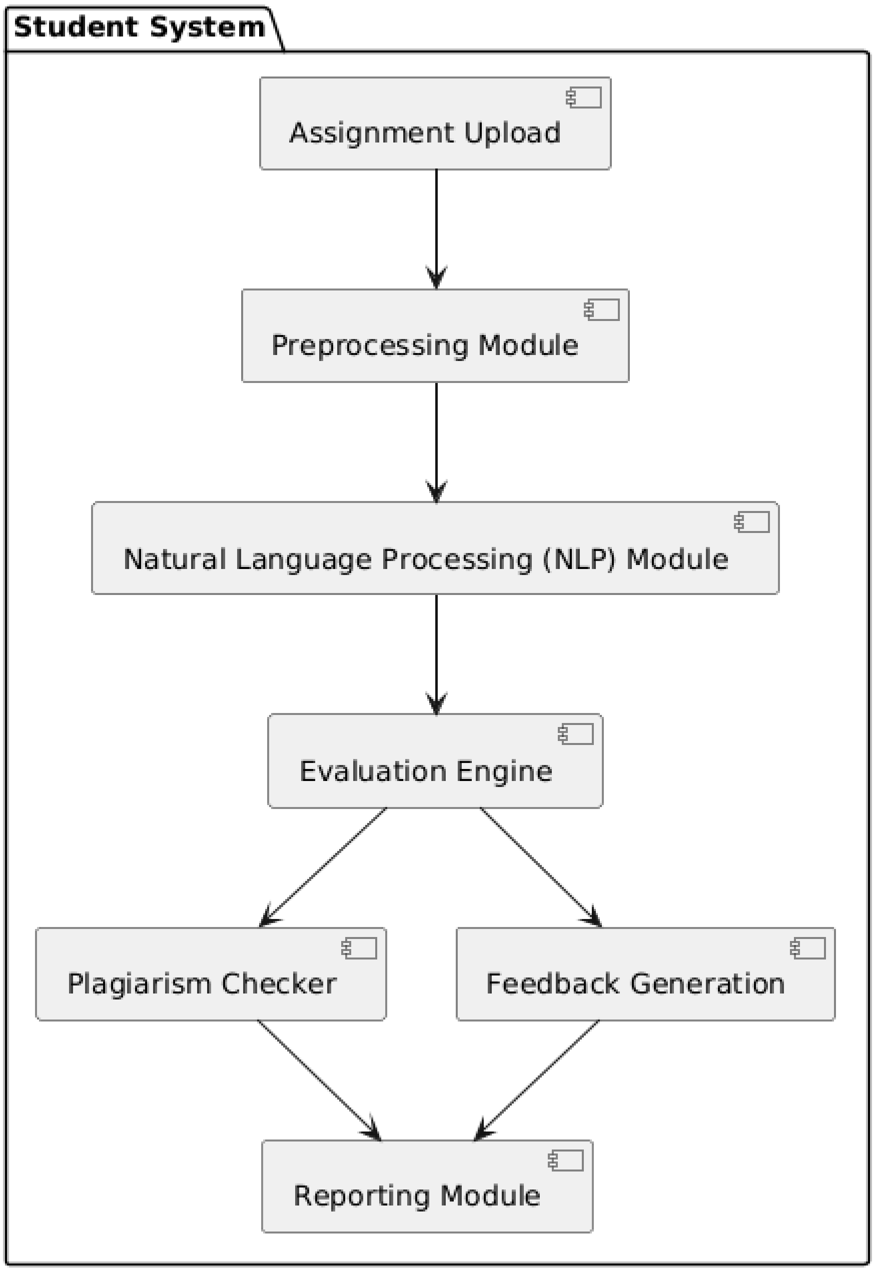


Figure x - Component diagram of the the system

5.2 Sequence Diagram

Assignment upload -> Criteria upload -> Preprocessing module (text extraction) -> NLP module (semantic analysis, grammar and syntax) -> Evaluation (grades the assignments based on criteria) -> Plagiarism checker (checks for plagiarism and maybe AI generation?)

-> feedback generation (feedback is produced to help the students improve) -> reporting module