

Toward AI evaluation of student essays

P. Rantanen*, M. Saari*, U-T. Virta*, P. Abrahamsson*

* Tampere University / Faculty of Information Technology and Communication Sciences, Tampere, Finland
mika.saari@tuni.fi

Abstract—The integration of artificial intelligence (AI) into teaching practices has expanded rapidly, and there is increasing discussion about computer-assisted assessment. This study investigates the application of AI for evaluating student essays, focusing on a Software Testing course at Tampere University. Traditionally, on this course essays are assessed manually by teaching assistants using predefined evaluation criteria. This research explores the feasibility of automating the evaluation process by comparing assessments generated by AI models—OpenAI’s large language models (LLM) API—with human evaluations.

The study focuses on the research question: Can automated assessment replace human evaluation? The key stages in this study include selecting a suitable AI model, its implementation and configuration, and designing an automated assessment workflow that aligns with human evaluation. The results indicate that, while AI models can evaluate at a level comparable to humans, current language models have limitations, such as consistency issues and an inability to interpret context. The findings of this study suggest that AI can be used as complementary tools rather than fully replacing humans.

This work highlights the potential of AI in educational assessment while acknowledging the need for further refinement in model performance. Future directions include exploring open-source models and addressing the technical challenges related to resource requirements.

Keywords—AI, automatic evaluation, student essays

I. INTRODUCTION

The use of artificial intelligence (AI) has grown significantly in recent years, especially following the public release of models like ChatGPT. This rise in AI adoption has enabled the exploration of new research domains, such as the application of AI in various fields [1]. In education, AI can be used in multiple ways, and it is essential to guide students toward the responsible and effective use of these tools. Particularly in programming courses, AI can be an invaluable asset for enhancing student learning and problem-solving skills [2]. However, increasing ethical risks have raised concerns regarding several aspects, such as personal data privacy, learner autonomy, and the potential misuse of AI technologies in education [3].

Computer-Assisted Assessment (CAA) has been the focus of research even prior to the development of artificial intelligence [4]. AI support in education, while promising, is still in the early stages of adoption. This prompted us to design an experimental study to investigate the use of AI in the automatic assessment of student essays. In this paper, we describe a Software Engineering course where students complete a project in the form of an essay, which has traditionally been assessed manually using predefined

evaluation criteria. Given recent advancements in AI, we wanted to explore whether AI could be used to evaluate student essays based on the same criteria as those used by human evaluators.

Our research question is as follows:

RQ: Can automated assessment replace human evaluation?

To address the question, we developed a system using two different AI models: OpenAI’s API and an open-source LLM such as the LLaMA model.

In this study, we present the model used for the automatic assessment of student essays, detailing the process and tools involved in its implementation. Furthermore, we analyze the results of the automated assessments and compare them with human evaluations. In the discussion section, we highlight key findings and considerations for the potential adoption of automated assessment systems in educational contexts.

II. BACKGROUND

AI has been integrated into our everyday work ever since ChatGPT was released to the general public. While AI assistants and tools were initially widely used within the IT sector, AI tools have now been adopted across other fields as well. A good example of this is higher education, where AI has been introduced as a teaching assistant [5]. In higher education—and particularly in programming courses—AI is used as a support tool, and feedback has also been gathered from students on their experiences of using AI tools [2].

AI tools are also emerging to assist teachers in the assessment of practice exercises. In a study by Kortemeyer [6], learning and assessment in physics through artificial intelligence are discussed. In solving physics problems, not only the final results but also their derivations are essential. This research investigated an AI-assisted workflow in which MathPix and GPT-4 tools were used to evaluate handwritten physics derivations. In the final assessment, the study found that the developed method could be best utilized as a support tool for human evaluators.

In a study by Weegar and Idestam-Almqvist [7], the use of machine learning for grading was critically examined. Their findings indicate that, despite advancements in classifier accuracy, fully replacing manual grading with machine learning-based methods is not yet feasible. Automated grading outputs still require human review to ensure fair-

ness and accuracy, particularly in high-stakes examinations where precise assessment is essential.

III. USE CASE: GRADING SOFTWARE TESTING PLANS

The "Software Testing" course [8] has been held at the Tampere University for many years. The course structure consists of a (written) exam, practical (weekly) exercises, and a larger two-part group assignment, with approximately half of the course grade being awarded from the group assignment. In the assignment students form groups of two students (pair work) to design a software testing plan for an application, and in the second part they implement the tests according to their plan. In this case, we chose to investigate how well AI was able to grade the first part (design plan) of the students' work.

There were a couple of reasons for choosing the first part. First, the task description (and grading instructions) of part one has remained largely the same, and even though in this case we only chose student work from a single academic year (2023-2024), it might be possible to compare a larger set of work in the future. Unfortunately for this case, our university regulations prohibit the storing of student work unless the work is specifically collected for research purposes with the consent of the students, and for this reason we do not have access to work from previous years.

Second, the part one design document is generally text only. The students are required to draw a diagram to illustrate their testing setup, which can be challenging to analyze using basic LLM setups, but the testing process also has to be described in the text, so missing the diagram should not be a major issue. Grading part two would require the AI to access student code repositories, run the test suites, and validate the results, which would be a more complicated task. There is also no requirement for the students to preserve their repositories after the course has ended and, thus, some of the repositories could already be missing.

A. Group Assignment

The group assignment instructions (*assignment.pdf*) can be found in the GitLab repository provided in the references [9]. The same repository also includes the grading instructions (*instructions.md*), which are given both to the humans (teachers) who are grading the work and also to the AI. It is worth noting that in this case we are not assessing the quality of the group assignment or the grading criteria—the assumption is that they are applicable for the case in question. Instead, we are interested in whether AI can generate grading results comparable to those made by humans, while using the same instructions provided for humans. The reasoning is that whilst a teacher for a software testing course might be able to create customized prompts for an AI, this may not be the case for all teachers. Also, for the grading solution to be universally applicable, the AI should understand instructions written

in natural language without requiring the use of overly complicated prompting magic.

B. Grading Criteria

The full grading instructions can be found in the *instructions.md* file [9]. The grading instructions consist of five separate criteria, each having three or four sub-criteria. An example of Criterion 3 is provided in Figure 1.

```
### Criteria 3: Testing environment and tools (1 point)
0.5 At least 5 to 10 tools and/or testing environments have been included in the plan.
There is a brief documentation of each tool or environment introducing the tools.
0.25 The plan lists tools, but the documentation for the tools is lacking.
-1 No tools or environments have been tested, and the document contains no descriptions
of testing tools.

0 The students have tested the tools they have chosen, and these tests are documented in
the plan.
-0.25 Based on the documentation it is unclear if and how the students have tested the
chosen tools.
```

Fig. 1. Grading criteria example.

A human grader would read the submitted student work and go through the list of criteria, picking the lines from each sub-criterion that best describe the content in the work. In other words, a teacher could pick either the 0 or -0.25 point line from the lower sub-criteria in the example figure, depending on which of the lines better matches the students' work. The AI (LLM), on the other hand, does not understand the content of the work as such, but the assumption is that it would pick the matching line based on its similarity to the content of the students' works—basically, more or less as a human would do.

All points from each criterion (and sub-criteria) are added together. In the final grading the points are rounded to the nearest integer, but in this case we are only comparing the unrounded total points (sums). An example of final grading for two pieces of student work (*1.pdf* and *2.pdf*) is shown in Table I. It should also be noted that, even though the instructions do not explicitly allow it, as can be seen in the example, human graders have in some cases given other point scores than those defined in the instructions. These are generally for cases where the grader has been uncertain of whether to give, for example, 0 or 0.25 points, and has chosen to give 0.125.

In the case of generative AI, the output will vary slightly on each run and, in practice, multiple (grading) attempts are required to find out exactly how much the grading varies. In our case we chose to calculate the mean and standard deviation for AI grading to illustrate how much the results vary. Using the mean also causes the points not to match the exact point criteria listed in the *instruction.md* file, but as mentioned above, human graders do not always award points exactly as defined in the instructions either, so these minor variations should more or less even out. The AI grading process is explained in more detail in Section IV.

IV. AI GRADING PROCESS

A high-level diagram illustrating the AI grading process is shown in Figure 2. As mentioned in earlier sections,

TABLE I
EXAMPLE OF FINAL GRADING BY A HUMAN GRADER AND AI (GPT4o).

pdf	p_c1		p_c2		p_c3		p_c4		p_c5		p_sum	
1.pdf	0.375		2.125		0.75		1.75		0.125		5.125	
2.pdf	0.5		2.5		0.75		2.25		0.25		6.25	
	ai_p_c1_mean	ai_p_c1_std	ai_p_c2_mean	ai_p_c2_std	ai_p_c3_mean	ai_p_c3_std	ai_p_c4_mean	ai_p_c4_std	ai_p_c5_mean	ai_p_c5_std	ai_p_sum_mean	ai_p_sum_std
1.pdf	0.5	0.0	2.0	0.0	0.65	0.21	2.0	0.0	0.47	0.42	5.62	0.57
2.pdf	0.5	0.0	2.5	0.0	0.94	0.12	2.19	0.26	1.44	0.12	7.56	0.42

the full *assignment.pdf* and *instructions.md* can be found in the code repository [9]. A rough example of *points.csv* is shown in Table I. In the example shown in the table the data is split into separate lines for typographic reasons (clarity), but in the actual file the human and AI points and sums are on the same row (one row per student work, i.e., PDF file), and in the case of a single run, there would only be single point values instead of means and standard deviations.

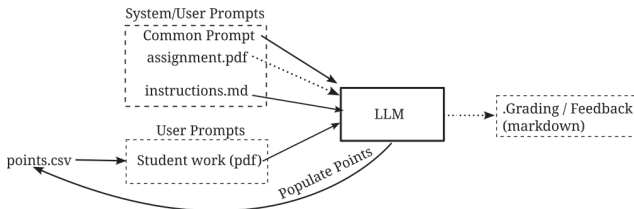


Fig. 2. High-level diagram of the grading process.

All the students' work (PDF files) is converted to text. In our case there are no image-only PDFs, so only the raw text is extracted. Each piece of work is passed to the LLM one-by-one as its own user prompt, containing no additional information or extra prompt instructions, only text.

In addition to the user prompt, two or three system prompts are given to the LLM. The basic instructions, called *Common Prompt* in this paper, are always given as a system prompt for each grading task. The assignment instructions can be given as a system prompt—whether to give an assignment instruction prompt or not is discussed more in Section V. In every case, the grading instructions prompt is given as a system prompt. The actual prompts (*Common Prompt*, *Assignment Prompt*, and *Instruction Prompt*) are shown in Figure 3. Also, in October 2024, o1 models provided in Azure did not support the "system role", and for o1 mini the system prompts are also given as user prompts.

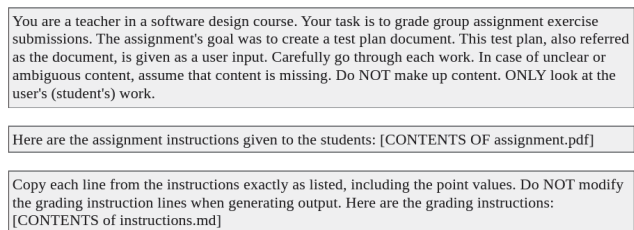


Fig. 3. System prompts, from top to bottom: Common Prompt, Assignment Prompt, and Instruction Prompt.

Based on these prompts the LLM will generate grading information and brief feedback (summary) about the processed student work. In practice, the teachers are also required to write some brief feedback for students about their work, and the AI's summary could be used for this purpose, but in the context of this paper we do not analyze the quality or usefulness of the textual summary produced by the LLM, and we are only interested in the numeric scores. In Figure 2, *Grading/Feedback (markdown)* is marked with a dotted arrow line and box to illustrate the omission of feedback and the inclusion of point scores. *Populate Points* is marked with a solid arrow line.

Finally, the *points.csv* file is updated with the generated AI grading, and after multiple runs with each file, a combined dataset including means and standard deviations can be created, as illustrated in Table I.

V. TESTS AND RESULTS

Our dataset consisted of a total of 68 pieces of student work. Each piece of work ranged from 10 to 30 pages, having approximately 10 pages of text on average, plus 2-3 pages of references and appendices. The works were manually checked to ensure they did not contain any image-only PDF files, and "clearly incorrect" work was removed. This included empty paper submissions or work that contained only a single brief chapter or was of very low quality—essentially work that would be directly rejected by a human grader. These were considered "clearly incorrect" because, in this study, we did not aim to evaluate how AI would handle empty papers, incorrect file formats, or other irrelevant content. After running the tests and validating the results, a single human grading (*49.pdf*) was found to be erroneous and was removed.

The results can be seen in Table II. The original results also contained the scores for individual criteria, as explained in Section III-B, but for simplicity they are excluded from the table, and only the total sums are given.

A. The Tests

We executed the tests using Microsoft Azure's variant of the OpenAI service [10], and more specifically, using the (Chat) Completions API. The tests were done using GPT-4o (version 2024-08-06), GPT-4o mini (version 2024-07-18) and o1 mini (preview version 2024-09-12) models, in October 2024 (17th-31st). Each piece of student work was run 10 times on each model setup, totaling 3400 file analyses for the whole dataset.

Generative AI output will always vary slightly and, on each run, some of the files would fail to produce

TABLE II

RESULTS OF RUNNING THE GRADING ON GPT-4o, GPT-4o mini, AND o1 mini, WITH 10 RUNS PER PDF FILE (STUDENT WORK). FAILED FILES ARE MARKED IN RED. The *noinst* variants were run WITHOUT ASSIGNMENT INSTRUCTIONS (*assignment.pdf*). THE MAXIMUM SCORE IS EIGHT POINTS.

pdf	Human				GPT4o				GPT4o-noinst				GPT4o-mini-noinst				GPTo1-mini				GPTo1-mini-noinst							
	p	sum	mean	std	p	sum	mean	std	p	sum	mean	std	p	sum	mean	std	p	sum	mean	std	p	sum	mean	std	p	sum	mean	std
1.pdf	5.125	5.62	0.57	0.495	5.78	0.29	0.655	5.85	0.21	0.725	6.05	0.96	0.925	6.17	0.31	1.045												
2.pdf	6.25	7.56	0.94	1.31	6.95	0.16	0.7	7.2	0.26	0.95	6.07	1.19	-0.18	6.1	0.12	-0.15												
3.pdf	5.25	5.88	0.84	0.63	5.95	0.28	0.7	6.4	0.21	1.15	6.67	0.6	1.42	6.3	0.72	1.05												
4.pdf	6.5	7.06	0.24	0.56	7.39	0.16	0.89	7.95	0.16	1.45	7.29	0.73	0.79	7.0	1.25	0.5												
5.pdf	5.0	5.05	0.34	0.049	6.3	0.21	1.3	6.55	0.91	1.55	5.03	1.43	0.03	5.43	0.62	0.43												
6.pdf	7.25	7.68	0.24	0.43	7.43	0.32	0.070	7.9	0.0	0.65	7.38	0.5	0.13	7.43	0.55	0.18												
7.pdf	7.25	7.97	0.08	0.72	6.95	0.16	-0.3	7.22	0.3	-0.030	6.89	1.05	-0.36	7.53	0.28	0.28												
8.pdf	6.4	5.96	0.45	-0.44	5.4	0.46	-1	7.05	0.31	0.65	4.81	1.03	-1.59	5.51	1.5	-0.89												
9.pdf	7.0	8.0	0.0	1	6.7	0.26	-0.3	7.67	0.58	0.67	6.35	0.47	-0.65	6.22	0.91	-0.78												
10.pdf	6.275	5.07	0.71	-1.205	4.95	0.32	-1.325	6.75	0.24	0.475	4.18	0.67	-2.095	3.46	1.28	-2.815												
11.pdf	6.15	5.73	0.41	-0.42	6.05	0.24	-0.100	7.05	0.54	0.9	5.66	0.93	-0.49	6.23	0.49	0.08												
12.pdf	6.4	7.55	0.52	1.15	6.75	0.29	0.35	8.0	0.0	1.6	6.09	0.58	-0.31	5.99	0.69	-0.41												
13.pdf	5.4	5.29	0.57	-0.11	5.53	0.51	0.13	6.83	0.18	1.43	3.29	0.97	-2.11	4.9	0.94	-0.5												
14.pdf	4.25	5.72	0.67	1.47	5.8	0.31	1.55	7.01	0.23	2.76	5.85	0.95	1.6	5.43	1.34	1.18												
15.pdf	2.05	0.28	0.25	-1.77	3.85	0.32	1.9	6.56	0.15	4.51	3.19	1.13	1.14	4.12	2.92	0.87												
16.pdf	5.375	4.22	0.36	-1.155	5.0	0.24	0.24	6.97	0.08	1.595	5.0	0.61	-0.375	4.69	0.74	-0.685												
17.pdf	4.25	5.88	0.76	1.63	6.03	0.32	1.78	6.9	0.0	2.65	5.5	0.75	1.25	6.14	0.35	1.89												
18.pdf	6.9	7.84	0.12	0.94	7.47	0.12	0.569	7.36	0.24	0.46	6.98	1.08	0.08	7.47	0.55	0.569												
19.pdf	6.0	5.39	0.61	0.501	5.37	0.36	1.095	5.88	0.82	0.36	5.88	0.82	-0.12	5.5	0.57	-0.5												
20.pdf	6.625	6.64	0.25	0.014	6.67	0.36	0.044	7.22	0.53	0.595	7.12	0.98	0.495	7.22	0.31	0.595												
21.pdf	6.4	5.53	0.38	-0.87	5.38	0.37	-1.02	6.88	0.36	0.48	4.15	1.83	-2.25	5.5	0.85	-0.9												
22.pdf	6.0	5.95	0.64	-0.049	3.3	0.38	-2.7	6.83	0.24	0.83	4.75	0.68	-1.25	4.88	0.85	-1.12												
23.pdf	6.25	6.44	0.19	0.19	6.25	0.15	-0.19	7.19	0.61	2.065	5.9	0.83	-0.35	6.4	0.66	0.15												
24.pdf	5.125	5.65	0.2	0.525	4.58	0.53	-0.545	6.95	0.07	-0.3	6.21	0.55	-1.04	5.95	0.64	-1.3												
25.pdf	7.25	6.5	0.0	-0.75	4.95	0.45	-2.3	6.95	0.07	-0.3	6.21	0.55	-1.04	5.95	0.64	-1.3												
26.pdf	7.0	7.61	0.4	0.61	6.6	0.21	-0.4	8.04	0.39	1.04	6.42	0.66	-0.58	6.17	0.61	-0.83												
27.pdf	6.75	7.42	0.17	0.67	5.75	1.0	0	6.79	0.32	0.04	6.03	0.96	-0.72	6.81	0.46	0.059												
28.pdf	5.75	6.05	0.11	0.3	6.0	0.0	0.25	7.88	0.36	0.83	6.69	0.75	-0.31	6.1	0.58	-0.9												
29.pdf	7.0	7.88	0.18	0.88	6.55	0.16	-0.45	7.83	0.36	0.83	6.69	0.75	-0.31	6.1	0.58	-0.9												
30.pdf	6.75	7.94	0.17	1.19	6.8	0.26	0.049	7.3	0.27	0.55	7.35	0.45	0.6	7.04	0.78	0.29												
31.pdf	5.9	6.17	2.11	0.27	5.1	0.38	-0.800	6.61	0.18	0.36	5.8	0.4	-0.10	5.94	0.51	0.04												
32.pdf	6.25	6.47	0.22	0.22	6.08	0.29	-0.17	6.58	0.48	0.39	5.42	0.44	0.795	5.97	0.87	1.345												
33.pdf	4.625	6.71	0.32	2.085	6.47	0.12	1.845	7.1	0.39	2.475	5.42	0.44	0.795	5.97	0.87	1.345												
34.pdf	5.4	6.26	0.44	0.859	5.31	0.49	-0.090	6.22	0.45	0.819	4.61	1.62	-0.79	4.9	0.88	-0.5												
35.pdf	6.0	5.75	0.52	-0.25	5.25	0.35	-0.75	6.8	0.16	0.8	5.72	1.15	-0.28	5.55	0.92	-0.45												
36.pdf	4.875	5.4	0.6	0.525	4.085	0.36	1.095	5.75	0.36	0.875	5.03	0.57	0.107	5.1	0.225	0.26												
37.pdf	5.875	7.04	0.3	1.165	5.45	0.26	-0.425	7.12	0.21	1.245	6.39	0.64	0.515	6.17	0.62	0.295												
38.pdf	7.5	8.0	0.0	0.5	6.08	0.17	-1.42	6.7	0.35	-0.8	7.33	0.41	-0.17	6.65	1.45	-0.85												
39.pdf	5.75	8.0	0.0	2.25	7.75	0.0	2	8.0	0.0	2.25																		
40.pdf	5.875	7.55	0.48	1.875	7.12	0.13	1.245	6.95	0.51	1.075	6.71	0.44	0.835	7.15	0.39	1.275												
41.pdf	6.125	7.95	0.16	0.825	7.1	0.0	-0.125	7.25	0.26	0.125	7.56	0.39	1.185	7.06	0.78	0.685												
42.pdf	6.375	7.97	0.08	1.595	6.65	0.24	0.275	7.0	0.5	0.625	7.56	0.39	1.185	7.06	0.78	0.685												
43.pdf	7.5	8.0	0.0	0.5	6.65	0.32	-0.85	8.0	0.0	0.5	7.75	0.27	0.25	7.66	0.4	0.16												
44.pdf	5.025	4.85	0.28	-0.175	6.85	0.16	1.825	6.97	0.08	1.945	4.3	0.72	-0.725	4.76	0.64	-0.295												
45.pdf	6.4	7.53	0.16	1.13	6.8	0.27	0.399	7.92	0.14	1.52	5.33	1.45	-1.07	5.84	0.6	-0.57												
46.pdf	7.25	7.95	0.16	0.7	7.0	0.41	-0.25	7.05	0.16	-0.2	6.47	0.51	-0.78	6.53	0.42	-0.72												
47.pdf	6.125	7.25	0.2	1.125	6.18	0.53	0.054	7.5	0.0	1.375	7.57	0.19	1.445	7.56	0.69	1.435												
48.pdf	4.65	5.1	0.27	0.449	5.78	0.78	1.13	6.7	0.39	2.05	5.34	0.81	0.69	5.53	1.03	0.88												
49.pdf	5.0	6.0	0.0	1.0	5.0	0.0	0	6.45	0.16	1.08	5.31	1.08	1.39	5.47	1.39	0.25												
50.pdf	5.125	7.1	0.17	1.975	5.8	0.51	0.675	7.3	0.16	2.175	6.61	0.61	1.485	6.15	0.95	1.025												
51.pdf	5.25	5.9	0.57	0.65	6.24	0.36	0.99	6.75	0.35	1.5	5.22	0.95	-0.03	4.72	0.96	-0.53												
52.pdf	7.25	7.5	0.22	0.25	6.65	0.24	-0.6	7.25	0.26	0	7.19	0.32	-0.059	7.14	0.28	-0.11												
53.pdf	6.5	6.6	0.13	0.099	6.22	0.28	0.74	7.4	0.21	0.9	7.0	0.6	0.75	6.75	0.63	0.25												
54.pdf	4.4	4.71	0.37	-0.69	4.7	0.78	-0.7	5.72	0.34	0.319	4.65	0.78	-0.75	4.59	0.76	-0.81												
55.pdf	5.125	5.65	0.26	0.525	4.69	0.5	-0.435	7.3	0.6	2.175	5.67	0.95	0.545	5.71	0.89	0.585												
56.pdf																												

Turbo, GPT-4, and GPT-4 Turbo), omitting the assignment instructions made no measurable difference, i.e., the results were still unusable.

B. Conclusions

Considering our research question *Can automated assessment replace human evaluation?*, we need to have some baseline to compare against. In our case, part one of the group assignment has been graded for many years, and we have some historical data on how much the grading varies between graders. Looking at the results of human graders, we can say that there is on average a 0.5-1.0 point variance between the grades in our use case, which has 8 points maximum. The same work is never graded by multiple graders, but there have been discussions on problem cases, as well as grading practice in cases where the averages between different graders seem to vary a lot. Nevertheless, this estimate must be viewed with some uncertainty. However, with several years' experience and a few hundred pieces of graded work, there is some baseline to compare against. The dataset chosen for this use case was grading given by two human graders, and the average difference between the grades was approximately 0.5 points, which is lower than in previous years.

In Table II, p_sum_diff shows the difference between *Human p_sum* and each model's *ai p_sum_mean* ($ai_p_sum_mean - Human_p_sum = p_sum_diff$), with positive values indicating that AI scores are higher than human scores. If we consider a difference of approximately one point an acceptable variance, then with a few exceptions, the points graded by the AI are close enough to the points awarded by human graders, with GPT-4o models performing best overall.

We can also see that the AI, in general, gives slightly higher grades than humans. One possible reason for this is that the LLMs (or the API) we used does not directly understand diagrams. It seems that AI will give the maximum points as long as any diagram is present. The portion of points gained via diagrams is not overly high in the assignment, and approximately 0.5 points could be the point decrease given by a human grader, assuming the diagram is not completely incorrect. Similarly, the AI does not validate the references students have used, but one could also argue that human graders also do not always look at the references in too much detail.

In this case we did not further analyze the individual criteria. In practice, these are very case-specific, and should be validated carefully within the context of the course or work to be graded. Examining where the AI differs in individual criteria could also help to determine whether the criteria could be modified or even checked manually by a human. In this case, checking criteria related to figures included in students' work is one example case where human verification of the criteria might be needed.

Assuming that human graders are also not infallible, and when checking dozens of pieces of work totaling hundreds of pages, the human graders are almost guaranteed to

make some mistakes, we could also accept the fact that AI may also make mistakes. With this assumption and the acceptable low difference between human grades and AI grades, we can say that the AI produces results comparable to those of human grades, and thus, could replace human graders, at least in our use case. Furthermore, even if some human intervention, such as checking individual criteria were required, this would still save hours and hours of somewhat tedious and error-prone work of reading through long textual documents.

VI. DISCUSSION

Even though the statistical data shows that AI could replace human graders, some open issues remain. First and foremost, even though our tests show that our approach could be applicable in the software testing domain, it does not guarantee that is universally applicable in all domains. The training data for commercial models is not open as such, so it is challenging to ascertain their capabilities without testing. Thus, to truly validate the approach, more testing in other domains and use cases is required.

Using commercial models also poses additional limitation on the applicability of the solution. Even though AI service providers such as OpenAI does offer the older models for use, the pricing policy is aggressive. For example, model 3.5 received updates from OpenAI in 2023 and GPT-4o was released in May 2024 (only five months before our paper was written), with GPT-4 Turbo being the default model until then. Using the current prices of 2.5 U.S. dollars per million input tokens for GPT-4o and 0.15 U.S. dollars per million input tokens for GPT-4o mini [11] as a reference point, the current prices for GPT-4 Turbo are now four times the price of GPT-4o and 67 times the price of GPT-4o mini. For GPT-3.5 Turbo the price is not quite that high, but it is still over three times the price of GPT-4o mini. This makes it somewhat financially risky to rely on using older models, and based on our tests, as the same prompts do not work with the older models, it is also possible that they will not work on newer models, forcing an update of the implementation when the models (and model pricing) are updated.

Naturally, one should also update the task instructions occasionally, and could update the prompts at the same time, but as testing the prompts takes a certain amount of time, it may not always be worth setting up an automated solution as opposed to manual human grading. Alternatively, if the required GPU resources are available, it might also be possible to run open source models, which would make it easier to "freeze" the implementation in a more technically and financially stable version. How well various open-source models would work in this use case is one possible future direction for research.

In this case we also used only the baseline models available from OpenAI. It would also be possible to fine-tune the models, for example, with our existing dataset of student work. Fine-tuning might help to decrease the amount of failed AI grading or eliminate the few prob-

lem cases, even though the grading is within acceptable tolerances in our current implementation.

Finally, in this paper we have not addressed any ethical or legal limitations that using AI grading might pose. The ethical discussion is a larger topic in itself, and whether local laws, regulations, or school policies allow or should allow the use of fully AI-powered graders to grade students is another debate altogether.

VII. SUMMARY

This paper discussed whether AI-powered solutions could be used to grade long textual (essay-like) student work, with our use case being the grading of test plans created as student group work in a software testing course. Thus, we presented the following research question:

RQ: Can automated assessment replace human evaluation?

Based on the data collected from testing three different models (GPT-4o, GPT-4o mini and o1 mini), and comparison with grading made by human graders (teachers), the conclusion is that, **yes, the AI-generated grading results are similar enough to human grading, that at least in our case, AI could be used to replace humans.** Still, certain issues remain with, for example, AI analysis of figures in student work or in validation of references. In practice, some parts of the work may have to be checked manually by humans, meaning AI would perhaps work slightly better as a supplementary tool. Even with the minor issues and the possibility to require some human intervention, based on our studies the use of AI can save a considerable amount of tedious, time-consuming, and error-prone work in grading long textual student work.

ACKNOWLEDGMENTS

This work was funded by the European Regional Development Fund and the Regional Council of Satakunta.

REFERENCES

- [1] A. Nguyen-Duc, B. Cabrero-Daniel, A. Przybylek, C. Arora, D. Khanna, T. Herda, U. Rafiq, J. Melegati, E. Guerra, K.-K. Kemell, M. Saari, Z. Zhang, H. Le, T. Quan, and P. Abrahamsson, "Generative artificial intelligence for software engineering – a research agenda," 2023. [Online]. Available: <https://arxiv.org/abs/2310.18648>
- [2] M. Saari, P. Rantanen, M. Nurminen, T. Kilamo, K. Systä, and P. Abrahamsson, *Toward Guiding Students: Exploring Effective Approaches for Utilizing AI Tools in Programming Courses*. Cham: Springer Nature Switzerland, 2024, pp. 331–346. [Online]. Available: https://doi.org/10.1007/978-3-031-55642-5_16
- [3] A. Nguyen, H. N. Ngo, Y. Hong, B. Dang, and B.-P. T. Nguyen, "Ethical principles for artificial intelligence in education," *Education and Information Technologies*, vol. 28, no. 4, pp. 4221–4241, 2023.
- [4] D. Pérez-Marín, I. Pascual-Nieto, and P. Rodríguez, "Computer-assisted assessment of free-text answers," *The Knowledge Engineering Review*, vol. 24, no. 4, pp. 353–374, 12 2009, copyright - Copyright © Cambridge University Press 2009; Last updated - 2023-11-29.
- [5] J. Kim, K. Merrill, K. Xu, and D. D. Sellnow, "My teacher is a machine: Understanding students' perceptions of ai teaching assistants in online education," *International Journal of Human-Computer Interaction*, vol. 36, no. 20, p. 1902 – 1911, 2020, cited by: 118. [Online]. Available: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-85089451219&doi=10.1080%2F10447318.2020.1801227&partnerID=40&md5=a5a1acc27c87d5ea0123a31370cf4260>
- [6] G. Kortemeyer, "Toward ai grading of student problem solutions in introductory physics: A feasibility study," *Physical Review Physics Education Research*, vol. 19, p. 020163, 11 2023. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevPhysEducRes.19.020163>
- [7] R. Weegar and P. Idestam-Almqvist, "Reducing workload in short answer grading using machine learning," *International Journal of Artificial Intelligence in Education*, vol. 34, pp. 247–273, 6 2024. [Online]. Available: <https://link.springer.com/10.1007/s40593-022-00322-1>
- [8] Tampere University, "Comp.se.200 software design course brochure," <https://sisu.tuni.fi/student/courseunit/otm-6fd3d772-a15e-4653-9d5a-607700440588/brochure>, accessed: 2024-12-11.
- [9] GPT-Laboratory, "Ai grader," <https://github.com/GPT-Laboratory/ai-grader>, accessed: 2024-12-11.
- [10] Microsoft, "Azure openai service," <https://azure.microsoft.com/en-us/products/ai-services/openai-service>, accessed: 2024-12-11.
- [11] OpenAI, "Openai api pricing," <https://openai.com/api/pricing/>, accessed: 2024-12-11.