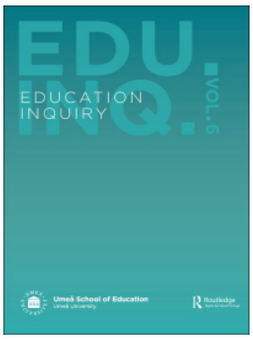

Routledge
Taylor & Francis Group

Education Inquiry

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/zedu20

Promises and breakages of automated grading systems: a qualitative study in computer science education

Clàudia Figueras, Alexandra Farazouli, Teresa Cerratto Pargman, Cormac McGrath & Chiara Rossitto

To cite this article: Clàudia Figueras, Alexandra Farazouli, Teresa Cerratto Pargman, Cormac McGrath & Chiara Rossitto (13 Feb 2025): Promises and breakages of automated grading systems: a qualitative study in computer science education, Education Inquiry, DOI: 10.1080/20004508.2025.2464996

To link to this article: https://doi.org/10.1080/20004508.2025.2464996

© 2025 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.

Published online: 13 Feb 2025.

Submit your article to this journal

Article views: 520

View related articles

View Crossmark data

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=zedu20

EDUCATION INQUIRY
https://doi.org/10.1080/20004508.2025.2464996

Routledge
Taylor & Francis Group

ORIGINAL ARTICLE

OPEN ACCESS Check for updates

Promises and breakages of automated grading systems: a qualitative study in computer science education

Clàudia Figueras [a], Alexandra Farazouli [b], Teresa Cerratto Pargman [a], Cormac McGrath [b] and Chiara Rossitto [a]

[a]Department of Computer and Systems Sciences, Stockholm University, Stockholm Sweden; [b]Department of Education, Stockholm University, Stockholm Sweden

ABSTRACT

Automated grading systems (AGSs) have gained attention for their potential to streamline assessment in higher education. However, their integration into university assessment practice poses challenges, particularly for teachers in computer science seeking to balance their workload while ensuring an adequate and fair assessment of students' programming skills and knowledge. The present study focuses on individuals with expertise in developing, using, and researching AGSs in higher education, whom we refer to as "AGS experts". Through semi-structured interviews, we examine how the AGSs they engage with impact their work and assessment practices in computer science education. Drawing on the concept of *breakages*, we argue that while AGS experts invest time and effort in developing these systems, enticed by the promises of more efficient workload management and improved assessment practices, the actual use may introduce tensions leading to breakages disrupting assessment practices. Our findings illustrate the complexities and the potential impact the deployment of AGS brings to assessment practices within a public university setting and discuss the implications for future research.

KEYWORDS

Automated Grading Systems; autograding; higher education; breakages; assessment

1. Introduction

Automated grading systems (AGSs) are becoming increasingly popular as they provide opportunities to fully, or partly, automate the grading of large numbers of assignments and examinations (Swiecki et al., 2022) of various designs (e.g. multiple-choice, essay questions) and subjects (e.g. computer science programming, mathematics, language learning). Despite the name suggesting full automation, AGSs are typically utilised as decision-support systems, with teachers retaining the ultimate – moral and legal – responsibility for grading. As in other areas, implementing AI and algorithmic decision-making systems in Higher Education (HE) is often associated with promises of increased efficiency, reduced workload, objectivity, and cost savings (Cerratto Pargman & McGrath, 2021). While AGSs may be introduced as a solution to the issues arising from the digitalisation of educational practices in HE (Bearman et al., 2020),

CONTACT Clàudia Figueras claudia@dsv.su.se Department of Computer and Systems Sciences, Stockholm University, Kista, Stockholm SE-164 07, Sweden

© 2025 The Author(s). Published by Informa UK Limited, trading as Taylor & Francis Group.
This is an Open Access article distributed under the terms of the Creative Commons Attribution-NonCommercial License (http://creativecommons.org/licenses/by-nc/4.0/), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited. The terms on which this article has been published allow the posting of the Accepted Manuscript in a repository by the author(s) or with their consent.

deploying AGSs in educational practice is complex. Such complexity is, among other things, linked to the ethical and social implications that such systems can bring to everyday assessment practices (Farazouli, 2024; Selwyn et al., 2022). Nevertheless, empirical studies exploring the impact that AGSs have on university teachers' assessment and how their use manifests in practice are scarce (Farazouli, 2024). We address this empirical gap through a study of "AGS experts", individuals with expertise in developing, researching and/or use AGSs in their assessment practices. We interviewed various AGS experts at a computer science department to investigate how they experience the use of AGSs for assessment in a Computer Science Higher Education context.

By examining AGS experts' experiences, we aim to contribute to a more nuanced understanding of how technological innovations such as AGSs are perceived and valued in everyday assessment practices – including the potential for their implementation to widen the gap between performance-oriented assessment practices and genuine student learning and engagement (Vattøy et al., 2022) at a time of accelerated digital transformation (Holmes & Tuomi, 2022; Nicklin et al., 2022). We challenge naive views of the relations between education and technology currently reflected in technological determinism and instrumentalist (Jandrić & Knox, 2022) beliefs regarding the deployment of AGSs by focusing on their actual rather than intended use. To this end, the following research questions were posed:

RQ1: *What do Automated Grading System (AGS) experts value about the use of AGSs in assessment practices?*

RQ2: *What breakages do Automated Grading System (AGS) experts perceive when using AGSs in assessment practices?*

By "assessment practices", we mean grading course assignments and exams in computer science education. Research question one (RQ1) seeks to understand the broader qualities of AGSs considered valuable by AGS experts, such as the specific characteristics of the AGSs and the benefits they confer on assessment practices. Research question two (RQ2) examines "breakages" (Jackson, 2014; Pink et al., 2018), understood as potential constraints, disruptions, and unintended consequences associated with the integration of AGSs in assessment practices, which we further explain under section 3. More specifically, RQ2 seeks to provide an account of the breakages that AGS experts recount from their experience in the design, development and deployment of AGSs in assessment practices.

Our study makes several contributions to the education and educational technology literature. First, we offer a critical understanding of AGS experts' experience of developing and using AGSs in university computer science education in their assessment practices, enriching the current polarised discourses on adopting AGSs (Hsu et al., 2021; Sherman et al., 2013; Baniassad et al., 2021; Haldeman et al., 2021). Secondly, we provide empirical evidence of the opportunities, detailing what works with AGSs in assessment practice, along with the breakages, which include constraints and unintended consequences that AGSs can introduce in assessment. Thirdly, our work contributes to the broader debate on the influence of digital technologies and AI, such as machine learning techniques, algorithmic decision-making and predictive analytics on teaching and learning practices.

2. AI and Automation in assessment

Automated grading systems, as decision-making and support systems, have been under development for more than 50 years (Burrows et al., 2015; Page, 1966), with technology-enhanced assessment and feedback systems being introduced as tools for supporting teachers' grading and students' learning (Burrows et al., 2015; Webb et al., 2013; Webb & Ifenthaler, 2018). AGSs have garnered significant attention in recent years (Burrows et al., 2015) as the growing challenge of teaching at scale has prompted an increased reliance on the use of automated systems such as AGSs (Haldeman et al., 2018). Such systems have been portrayed as high-quality and sustainable solutions for managing the rapid growth in student numbers, especially in computer science and programming courses, and coping with the demands of assessing larger groups of students (Rao, 2019) . However, the deployment of AGSs, particularly, has sparked a debate concerning accuracy and precision (Hsu et al., 2021).

Recent studies on AI developments, such as AGSs, have highlighted its potential to enhance established assessment practices by being more adaptive and dynamic than manual grading, primarily due to its ability to provide immediate and meaningful feedback (Haldeman et al., 2018; Sherman et al., 2013; Swiecki et al., 2022). However, resonating with concerns on the use of AI in the public sector more broadly, scholars have also recognised that deploying AI in assessment can bring challenges related to accountability and the risk of limiting the pedagogical role of assessment and learning (Holmes & Tuomi, 2022; Swiecki et al., 2022; Vattøy et al., 2022). Despite this, there is a lack of empirical evidence that informs how AGSs work in practice (Farazouli, 2024). Our study contributes to critical discourses regarding AI and automation in higher education. It aims to open new empirical terrains exploring the actual – and not only the intended – use of AGSs in the context of computer science education with a focus on assessment.

3. Breakages as points of departure

Seeking to critically understand how AGSs are experienced in practice, we turned to Jackson's concept of "broken world thinking", which provides a unique perspective for studying digital systems (Jackson, 2014). Broken world thinking "asks what happens when we take erosion, breakdown, and decay rather than novelty, growth, and progress, as our starting points in thinking through the nature, use, and effects of information technology and new media" (Jackson, 2014, p. 221). Jackson's concept encourages us to view breakdowns, and other moments when things do not work as expected as integral to the lifecycle of technology; the perspective that extends to environmental, technological, and societal systems, which are inherently fragile and prone to break, reflecting the inherent changing nature of our world (Jackson, 2014). Following Jackson, we focus specifically on "breakages" in this paper, understanding them as integral to the lifecycle of technology, which also includes "breakdowns" and "repairs".

In science and technology studies (STS), breakages and breakdowns are viewed as more than mere failures; they often act as critical components of success and reflection (Rosner & Fox, 2016). Several scholars have applied these concepts to scrutinise broader relations between society and technology (Jackson, 2014; Pink et al., 2018; Rosner &

Fox, 2016; Tanweer et al., 2016; Dye et al., 2019; Nemer & Chirumamilla, 2019; Rosner & Ames, 2014). Pink et al. (2018) probed the creation and breakage of personal data in self-tracking devices and data science practices to understand how meaning is attributed to data. Notably, these moments of breakage spotlight the intricacies, vulnerabilities, and the broader socio-cultural context of technology that seldom surface in analyses seeking to understand how digital technology enhances education.

The present study focuses on the breakages that AGSs might bring to assessment practices in order to grasp the complexity of integrating technology into higher educational practice. Following Felt et al. (Felt et al., 2016), we view the development of AGSs as inseparable from the social contexts and institutional structures in which they are developed and used. Such a perspective acknowledges that technological advancements not only influence societal norms and behaviours but are also shaped by them.

Inspired by Pink et al. (2018) and Jackson (2014), we approach AGSs with an interest in understanding how such systems are made, broken, and repaired as we argue such an analytical lens can reveal socio-technical entanglements, which most often are overlooked in studies about technologies in education (Edwards & Fenwick, 2016). To this end, we examined breakages in assessment practices within and resulting from AGS development and use in an education setting. Unlike breakdowns, which are total system failures, breakages are minor malfunctions or disruptions that can occur for any number of reasons; for example, in AGSs, breakages can be caused by human error, software bugs, or improper configurations. Breakages do not necessarily render an AGS wholly inoperative but may require users to find alternative solutions or workarounds. It is also worth noting that a breakage within an AGS does not inherently suggest a coding flaw – the system may remain operational.

Breakages arise in everyday educational practices, and their identification and categorisation are negotiated and shaped by material, infrastructural, gendered, political, and socioeconomic factors (Rosner & Ames, 2014). Recognising that breakages are not solely failures but opportunities for learning and adaptation may foster a culture of resilience around technologies developed and used in educational settings.

4. Methods

Most often, empirical studies on AGSs interview or survey students (Hsu et al., 2021; Rao, 2019) or teachers (Rao, 2019). So, we conducted an interview study with AGS experts, that is, university staff in computer science education who develop and/or use AGSs in their assessment practices. The study participants guided us through their developed AGSs, how they initiated and developed AGSs and the challenges they encountered when using them. By engaging with AGS experts who are both developers and educators, we aimed to shed light on the interplay between the different systems' technical qualities and assessment practices rather than viewing them as separate entities.

Conducting in-depth, semi-structured interviews allowed the participants to reflect and discuss their experiences with the systems they have developed. The participants' expertise and engagement with such technologies added richness to our data, providing details about the development and use of AGSs.

4.1. *Sampling: Participants and AGSs*

4.1.1. *Participants*

We employed a purposive sampling strategy to select study participants involved in the development and/or use of one or more AGS in their institution. All participants in this study were employed within the same Computer Science department of a public Swedish university. The sample consisted of seven participants of different ages and experience levels who were involved in teaching activities in various bachelor's and master's courses (e.g. Introduction to Data Science and Java Programming), summarised in Table 1.

The participants had a high level of expertise in computer science and, therefore, brought detailed and rich insights that contributed to an in-depth understanding of how AGSs are designed and used. We interviewed each participant individually, focusing on their AGS development process, implementation challenges and ethical considerations surrounding the use of AGSs in assessment. Because they were all developers engaged in developing, using, or researching AGSs, we were able to explore the relationship between development and the pedagogical act of student assessment and identify the tensions arising therefrom.

Given that AGSs are still in a pilot phase and are not widely and officially used by public universities internationally and in Sweden, identifying participants that have been involved in the design and the deployment of these systems in practice was a challenging task, and therefore our sample was limited into seven AGS experts. However, although we acknowledge that this is a relatively small sample ($n = 7$), we believe that as experts in both AGS development and teaching, the participants provided and discussed important and in-depth knowledge about the inner workings of AGS and their use when assessing assignments within computer science.

Furthermore, although our study was conducted at a Swedish Computer Science department, and subject-specific findings about such assignments cannot be broadly generalised to all disciplines, the broader concerns identified – such as efficiency, fairness, and workload management – can help inform other departments considering the implementation of AGSs for assignments and examinations.

4.1.2. *AGS*

The participants of the study were involved in the design and development of six AGSs that were designed to assess coding, short-text and long-text student responses. Each system was distinguished by its architecture, functionality, and areas of

Table 1. Summary of participants' work experience, position and AGS role.

Code ID	Years in academia	Position	AGS role (i.e. Researcher/Developer/Teacher)
I-1	25	Lecturer	D-T
I-2	10	Assistant professor	R-D-T
I-3	8	Research and teaching assistant	R-D-T
I-4	35	Associate professor	D-T
I-5	1	Research assistant	R-D
I-6	1	Research assistant	R-D-T
I-7	15	Professor	R-D-T

application. Four of the systems were operational (i.e. they have already been deployed), while two were still in the development stage as research projects at the time of data collection.[1] System A employs natural language processing (NLP) in a first-cycle introductory computer science course. It clusters open-ended short-text students' answers based on similarity, suggests grades for each cluster, and allows teachers to simultaneously assess all responses to each question. System B is used in a first-cycle programming course, verifying whether coding assignments meet functional requirements. System C serves a similar purpose in a second-cycle programming course. System D is designed to automatically grade open-ended short-text assignments and exams in a first-cycle programming fundamentals course. Systems E and F, although not yet deployed, aim to provide recommendations for passing or failing grades in a data science course with open-ended long-text answers and assign quality scores to code snippets.

4.2. *Data collection*

Participants were recruited through an email invitation containing information on the study's principal objective, its affiliation with a larger research project, and its approval by the Swedish Ethical Review Board (Dnr 2022–0105–01). Once participants agreed to take part, they received a confirmation email containing an informed consent form, which they were asked to read and sign before their interview.

The interviews were conducted one-to-one, in English, at the participants' workplace or via Zoom. The first author conducted the interviews, which lasted between 45 and 90 minutes each, from mid-November 2022 to mid-January 2023. All the interviews were recorded, transcribed verbatim, and anonymised to ensure confidentiality. The research team strictly adhered to the procedures outlined by the European Union General Data Protection Regulation (GDPR) and the Swedish Authorities regarding data collection, storage, handling, and de-identification. We prepared a semi-structured interview guide (see Appendix) that covered the three primary themes: the participants' motivation in developing AGSs, their perception of subsequent changes in assessment practices, and the ethical aspects of AGS deployment. The interview guide was used as a support document in our semi-structured interviews to explore emerging themes and adapt them to what participants feel comfortable discussing. The second author's 2 years of experience in researching and developing an AGS within a public institution has contributed to our understanding of AGS experts' concerns and shaped the design of the interview guide.

The interviews comprised three sections. First, each participant was asked a series of introductory questions about their background and current position. Secondly, the participant was asked about the AGSs they had developed or used: a description of the system, its aim and purpose, the participant's contribution to the development process, its intended use, and its functionality. Thirdly, the participant was asked about their work with the AGSs, ethical issues, and the AGSs' impact on student-teacher relationships. We discussed topics such as obtaining student consent for data collection and deploying AGSs for assessment purposes, technical and ethical challenges encountered during development and implementation, fairness considerations, and the potential misuse of the tool.

4.3. *Data analysis*

We analysed the data using Braun and Clarke's (Braun & Clarke, 2022) reflexive thematic analysis (RTA) approach. RTA encourages researchers to be transparent about their theoretical departures (in our case, the STS literature on breakages and broken world thinking) when formulating questions and interpreting the data. Through active engagement with the data, we generated themes inductively (Braun & Clarke, 2021; Byrne, 2022), aligning with our qualitative aim to emphasise interpretation rather than content summarisation (Braun & Clarke, 2021). RTA's flexibility and depth also allowed for a richer analysis, although it requires reflexivity and can be time-consuming.

Our analysis was iterative, involving multiple cycles of theme generation and refinement. The first and second authors led the process, alternating between in-depth analytical sessions and presenting and discussing their interpretations with the other authors. They first adopted an exploratory approach, familiarising themselves with the empirical material by listening to the audio recordings separately, reading the transcripts closely, and making notes of potentially interesting points. These notes included the authors' interpretations of various aspects, including participants' views on the advantages and disadvantages of AGSs, contradictions, and changes in assessment practices due to grading automation. For example, they noted down the contradiction that even though AGSs may save grading time, it still takes a long time to create questions that will be automatically marked.

Open coding was performed separately by the first two authors, who formed initial themes. They first looked at the rationales and motivations that the participants had for deciding to develop or use AGSs, which connected to the promises that AGSs offer; in other words, the characteristics the participants value in these systems. Examples of such codes included "saving time" and "fewer work spikes". They noticed that unexpected constraints sometimes disrupted those promises, which they identified as breakages. For example, some of these breakage codes included "taking time to create questions" and "reluctance to change exam content".

The first two authors compared their coding and discussed how these might align with the research questions. These discussions continued until a consensus was reached, resulting in the final themes for each research question. The data were analysed to identify what participants valued in AGSs and to uncover instances of complete or partial breakages – subtle points of tension, constraints, or unintended consequences within the development or use of AGSs that contradicted the promises or values that participants had when deciding to develop or use AGSs. Through an iterative process, codes were refined by merging similar codes and re-evaluating their relevance. For example, "saving time" and "increasing efficiency" were merged into the sub-theme of "speeding up the assessment process" within the broader theme of "facilitation of workload management". Consolidated themes were then presented to the research team for further analysis and refinement so that they captured participants' experiences and perspectives.

Finally, the findings of our study aimed to reflect the collective insights of the participants and do not account for the influence of individual demographic variables such as age or professional experience. Although there were differences in how

participants spoke about AGSs – some focusing more on technical qualities (e.g. algorithms) and others emphasising social aspects (e.g. impacts on teaching practices) – we did not attempt to establish a link between these perspectives and the participants' age or professional background. Our goal was to identify shared experiences and patterns giving a broader understanding of the use of AGSs and AGS experts' perceptions.

5. Findings

We present our findings in two main sections, addressing the research questions individually. We begin by reporting the findings relating to RQ1 ("*What do Automated Grading System experts value about the use of AGSs in assessment practices?*") in subsection 5.1, followed by RQ2 ("*What breakages do Automated Grading System experts perceive when using AGSs in assessment practices?")* in subsection 5.2. In both sections, we include quotes from the participants.

5.1. *What do automated grading system experts value about the use of AGSs in assessment practices?*

To address this question, we explored the AGS experts' goals and needs underpinning the development and use of AGSs, including pedagogical, work-related, and professional needs addressed by these systems. Our findings highlighted two primary themes: the facilitation of teachers' workload management and the improvement of the assessment practice. The subsequent subsections provide detailed insights into both.

5.1.1. *Facilitation of workload management*

The study participants identified workload management as a primary reason for integrating AGSs into HE assessment practices. They expressed the desire for AGSs to help save time, particularly when they were stressed over managing large courses and meeting tight deadlines.

Several factors drive the need for efficiency in teaching and assessment. First, several participants said they had hundreds of students and a limited number of teaching assistants. Moreover, in the context of our study, participants described that teachers often have only three weeks to mark exams and assign final grades. This urgency amplified the appeal of AGSs and, as an extra benefit, students received their grades much sooner, as explained by the quote below:

> One motivation is also, of course, to be productive, to be able to do as much of the job as possible in as short time as possible. So, why not use tools to help you to be more productive? Moreover, for students, instead of having to wait three weeks, maybe they can get (their grades and feedback) in one week. (I-4)

Secondly, as participants' assessment practices often include evaluating coding tasks, manual assessment requires a meticulous analysis of each line of code, testing the output, and ensuring that the code's logic aligns with the examination requirements. One participant stated that AGSs radically reduce the assessment time needed to run

the compiler (the computer program that tests the functionality of the code/program) and check for technical and syntactical issues in the code.

According to the participants, using AGSs reduces the time needed for assessment and frees up time for other teaching responsibilities, such as preparing lectures, revising syllabi, conducting course evaluations, and developing plans for subsequent course improvements. Thus, while AGSs were not seen as a replacement for performing assessments, they significantly reduced the time spent on grading tasks:

> It also reduces the workload immensely when it comes to (code) building, compiling all of these steps with the code. I think it slices half of the time required for grading the assignment. I think maybe before, it took several days; maybe now it takes two days. So, I think that's a big win. (I-2)

Thirdly, grading hundreds of exams consecutively was described as demanding and laborious, often leading to errors and inconsistencies. By breaking down assessments into smaller parts, the participants mentioned that AGSs could offer the opportunity to avoid work spikes caused by receiving numerous and lengthy student assignments within short periods. Moreover, the participants mentioned that grading would become more consistent with AGSs. One participant described how collating all the exams alone took a full day (since they had around 450 exams), while the use of AGS allowed them to practice a more efficient way of organising the exams which drastically reduced the time and effort needed.

5.1.2. *Improving assessment practice*

As we discuss in this section, the participants viewed AGSs as valuable tools for improving student assessment and, therefore, benefiting students' learning. First, according to the participants, AGSs are used to analyse students' progress, for instance, by identifying patterns of failure in students' assignments or examination answers. System A, for example, helps identify common mistakes and patterns in incorrect open-ended short-text answers, enabling participants to refine their teaching methods and address difficult-to-understand concepts more effectively.

Secondly, participants stressed the importance of fairness, associating AGSs with objective, consistent, non-discriminatory and less vulnerable to human factors (such as tiredness). Participants expressed that AGSs could assist them in reducing potential biases. More specifically, participants discussing human errors in grading associated with factors such as fatigue mentioned that AGSs provide objective, consistent, non-discriminatory, and less biased results (e.g. because they do not get tired or experience emotion). This was recognised as a significant advantage, as stated in the following quote:

> The main point is that we also, as teachers, as graders, make mistakes when we grade. (…) We consider it as safe because it's a human there who does it. So, in this case, I would rather have an AI system that is not as lazy, that would make fewer erroneous judgements. (I-7)

Thirdly, another key benefit discussed by the participants is the support AGSs provide in applying consistent grading criteria across all student submissions. Some AGSs (e.g., system A) offer an overview of students' answers to a specific question simultaneously,

facilitating a more efficient grading process. The feature of some systems clustering similar student responses was considered particularly valuable because it ensures that comparable answers receive consistent grades. This approach also helps mitigate challenges associated with remembering previous grading decisions, especially when dealing with a substantial number of exams:

> "(. . .) one advantage is that we get to see all the answers to the same question at the same time. So, it's maybe easier to be fair also. So, if the answers are very similar also when you cluster them together, it's easier to give them the same grade. You don't have to remember, 'Oh, this person mentioned these three things but forgot the fourth, got this score in the last one'. You can see it directly. So, I think that can make it a bit fairer or at least supports you in making fairer decisions maybe". (I-3)

Lastly, several participants explained that AGSs could help improve the learning experience by emulating genuine programming environments. This training function allows students to experiment and test their solutions before final submission, fostering increased autonomy and confidence. Moreover, the participants noted that students seem to appreciate the opportunity to engage with an environment that closely mirrors the tools and processes employed by "real-world" programmers.

5.2. *What breakages do automated grading system experts perceive when using AGSs in assessment practices?*

In this section, we present the findings related to RQ2, examining how participants experienced breakages introduced by AGSs to their assessment practices. Drawing on participants' experiences deploying AGS and their impact on their assessment practices, several *breakages*, such as setbacks, disruptions and unintended consequences, were identified in the analysis. Such breakages signified moments where the participants' aspirations and expectations regarding the development of AGSs were challenged. The identified breakages, presented below, were associated with assessment design, transferability, bias perpetuation and fairness issues, the limited capacity of AGSs to assess student work, a loss of communication between students and teachers, and the impact of AGSs on the quality of submissions.

5.2.1. *Assessment design issues*

Contrary to the initial expectation that AGSs would streamline assessment and reduce grading time, several participants faced a significant breakage; namely, designing questions suitable for automated marking proved time-consuming and effort-intensive. According to the participants, the inflexible nature of AGS made it hard to modify exam content from year to year – despite recognising the need for improvement. Participant I-2 elaborates on this, highlighting the reluctance to modify assignments due to the considerable effort involved in making them more AGS-tailored:

> Changing assignments between years is a massive undertaking. It also takes a lot of effort to change these tests that have been implemented. So, we are reluctant to change the assignment, even though we know that there are some throwbacks with it. (I-2)

5.2.2. *Transferability issues*

For the participants, building AGSs sufficiently versatile to be used in different assignments and other courses was desirable. However, AGSs are not easily adjustable because they are ordinarily tailored to specific courses. Additionally, they can include bias embedded in historical student data and preferences associated with a particular course instructor during the system's training phase. The participants averred that adjusting the AGSs so they could be transferred to other contexts was more labour-intensive than manually grading, which defeated the object:

> But then, we changed the assignment, and then suddenly the model didn't do very good anymore. (. . .) It could replicate my quality assessment quite well, but yeah, we don't know how well it generalises to other assignments. I mean, if I have to manually assess 2,000 functions per course, then I think that takes more time than just simply grading them manually. (I-2)

5.2.3. *Bias perpetuation and fairness issues*

Several participants expected that AGSs would make assessments more objective, consistent, non-discriminatory, and less biased. However, concerns about fairness were pronounced, stemming from biases in training processes and the limited diversity among data annotators. While the participants associated AGSs with fairness and objectivity, they acknowledged that they could sometimes inadvertently perpetuate or introduce new biases.

Participants anticipated potential bias issues associated with AGS using machine learning or similar AI systems for grading. They mentioned that bias in the training data, along with the training strategies, could result in unfair grading, and there would be a higher risk of perpetuating historical bias (or outdated knowledge) if the model is not routinely updated. What intensified this concern among the participants was the speed at which AGSs can amplify and spread such biases compared with manual grading.

The participants were aware that human bias influenced grading. Still, they were concerned that an AGS, if trained on a specific grading style, might inadvertently magnify these biases (and render assessment less fair). One participant noted they were the only person grading the course, so the system was replicating their individual style. The participants suggested that more people should be brought in to grade to ensure fairness, allowing for more diverse perspectives and reducing individual biases:

> And then, at the same time, though, bear in mind that this project, the ground truth, was given by one person, me, who graded the course. So, the ground truth that the model will learn really depends on me. So, it's not really learning how to grade the course. It's learning how to replace me in grading the course. (I-7)

In sum, crafting questions for automated marking was time-consuming, leading to reluctance to change the exam design and adapt it to changing assessment needs. Participants highlighted the difficulty of making AGSs transferable to other contexts due to their context-specific design, extensive retraining needed, and embedded training biases.

5.2.4. *Limited capacity of AGSS to assess student work*

Several participants believed that AGS could not assess code quality, such as non-functional requirements, plagiarism detection, and other qualitative aspects of coding.

Two systems (B and C) focus on assessing functional requirements, leaving the evaluation of code quality to be carried out manually. Furthermore, the participants indicated that AGSs struggle with free-text questions, particularly those that are essay-like or open-ended. The inherent complexity of the former (because of structure, coherence, and depth of analysis) makes automated grading particularly difficult. Participants often invested significant time determining which questions were suitable for automated grading, limiting their creative freedom when designing assessments. As was mentioned in the previous subsection, 5.2.1, the impact of AGSs on assignment design is such that it often results in close-ended questions and memorisation tasks rather than open-ended questions and complex problem-solving tasks. Hence, student answers are usually homogeneous and similar, limiting creativity and innovative thinking. One participant reflecting on AGS disadvantages or "what could go wrong with AGS" discussed:

> Most of the assignments would look very similar to each other; there is no creative freedom. (…) So, it's limiting, perhaps, for those who wish to go above and beyond the base requirements. (…) [AGSs] fit everybody into a small box. All the students need to be inside a tiny little box, and there is no room for anyone to shine. (I-2)

5.2.5. *Student–teacher communication loss*

Despite a shared initial enthusiasm for AGSs among AGS experts, several participants expressed concerns about the potential loss of their personal connection with students. Some felt they had gained insights from manually reviewing exams that enabled them to adapt their teaching methods. For instance, one participant reflected on their potential impact on teacher-student relationships, worrying that they may diminish the very essence of teaching and the intimate feedback loops between teachers and students:

> If the assessment of the course is automated…, then I'm really losing the most essential part of my being a teacher, which is the connection with the students. When you mark exams, you're really looking at each and every answer. Handwritten is even better because you really see and feel. It's closer. And when I read an exam paper, I feel like I'm interacting. (…) And it's the feedback from me as well. I gain insights into my own evaluation criteria. And if it's automated, there is no way to figure that out. (I-7)

5.2.6. *AGS impact on submission quality*

The participants raised several concerns regarding learning processes and how they are framed when AGSs are used, particularly the fact that they may encourage students to aim to satisfy the minimal functional requirements (such as compiling, running, and producing the expected output), possibly at the expense of not cultivating their programming skills. Consequently, students may rush through assignments without paying attention to detail. Although AGSs emulate real programming environments, they may not assess students' programming skills adequately and holistically. For instance, system D uses unit tests to verify each unit's functionality. However, some students seem to rely on these unit tests in ways not anticipated during the system design.

> "[The programming] environment is more similar in that [the students] actually make something that works, but it doesn't necessarily test if they can program because certain students tend to use these sorts of systems as crutches.[2] (…) The important thing to test is

> 'Do you understand programming?' Getting it through the compiler is another issue, really, and that could just as well be tested on the assignment". (I-1)

According to the participants, AGSs may not actively encourage student reflection and a genuine understanding of programming (including problem-solving and design skills) in a comprehensive manner. While AGSs often provide feedback on code assignments (e.g. Systems B, C, and D), there is a concern that students may perceive this feedback merely as a series of error messages requiring immediate correction rather than viewing it as an opportunity to understand and learn from their mistakes through meaningful assessments.

In sum, participants mentioned concerns that AGSs could stifle student creativity by not encouraging creative and complex problem-solving. They were also concerned about potential damage to teacher-student relationships, as AGSs could reduce the personal interactions between teachers and students and nuanced insights of manual marking. In addition, AGSs were also perceived as potentially not promoting student reflection or a comprehensive understanding of programming, as feedback might be perceived as error correction rather than a learning opportunity.

6. Discussion

In this study, we conducted semi-structured interviews with AGS experts to examine what they value about AGSs and discern their perceptions of the breakages introduced by AGSs to assessment practices in computer science education. Our findings revealed that the study participants value AGS for its ability to help manage their substantial workload, offering a means to achieve more consistent and stable grading outcomes and save time. Additionally, they generally perceive AGSs as valuable tools for enhancing assessment practices, thereby potentially enriching students' learning. We also identified *breakages* AGSs pose in participants' assessment practices. Breakages were studied as snapshots of the subtle yet impactful constraints identified in AGS implementation within a computer science department at a public HE institution in Sweden. While breakages might not manifest overtly as complete breakdowns, they can still significantly impact a technology's use if not properly repaired. Notably, unlike major breakdowns, breakages can be repaired without requiring a complete system overhaul.

Contrary to the notion that teachers adopt new technologies when they perceive them as helpful and easy to use (Aldunate & Nussbaum, 2013), our findings suggest that even when teachers develop custom-built artefacts, challenges can arise when integrated into assessment practices. Although the development of AGSs is driven by aspirations for efficiency and improving assessment practices, breakages surface as critical points of tension, and anticipated benefits clash with reality.

The AGSs herein were developed to manage teachers' workload. Yet, as the participants themselves further explained, the time-intensive creation of automated questions and their specificity to particular courses led to breakages, hindering transferability. Our findings indicate that AGSs often constrain the type of assessment and task that can be assessed. In particular, AGSs were limited in terms of their capacity to evaluate the qualitative facets of student assignments, such as free-text responses and code quality,

leading us to the conclusion that the more complex the assessment task was, the more difficult it was for AGSs to be reliably integrated to participants' assessment practices. These tensions underscore that while the promise of efficiency is appealing, realising that promise in the context of HE practices is not always straightforward. This finding is consistent with previous studies on AGSs in large cohorts on upper-level computing courses (Hooper et al., 2024), in which teachers and students found their use burdensome. For instance, teachers have to invest considerable effort in programming unit tests, providing reference items, and writing custom test files to automate assessment (Hooper et al., 2024). Teachers, therefore, need to balance the benefits of automation against the initial and ongoing costs of refining and maintaining these systems.

The participants in the present study were driven by the appeal of improving assessment practices with AGS. The promise of more objective, consistent, non-discriminatory, and less biased assessments resonated strongly with them. This idea stems from the perception that being inherently subjective, human graders may benefit from the "mechanical objectivity" (Daston & Galison, 1992) of technological systems. Such a view resonates with research illustrating that algorithms are often considered neutral, objective entities (Green & Viljoen, 2020). Nonetheless, breakages in assessment practice, such as limited numbers of annotators and the perpetuation of bias through using AGSs, did not align with the pursuit of fairness. These concerns accord with previous studies showing that some AGSs may exhibit bias against minority students as a result of differences in vocabulary and sentence structure (Miao et al., 2021). AGSs still rely on teachers' individual grading styles, and their judgement and decision-making continue to play a role as participants interpret assessment criteria and make evaluative decisions. Furthermore, scholars have addressed the desocialising effect of automation and its impact on teacher-student interactions (Farazouli, 2024; Selwyn et al., 2023); indeed, several participants in the present study argued that manually grading exams provided valuable insights into their teaching methods and students' comprehension of the material. This self-reflective dimension could be compromised with the automation of the grading process. Despite the time-consuming and tedious nature of grading, it gives teachers a greater understanding of their students' competencies (Miao et al., 2021).

Another emergent breakage was that AGSs promote more instrumental learning incentives for students, discouraging reflection and thinking outside the box. When the participants used AGSs to provide feedback, they felt that students focused too much on getting their code to simply compile and run instead of grasping core programming concepts. This breakage resonates with studies (Baniassad et al., 2021) where teachers observed that some students substituted AGSs for personal reflection. A literature review which examined 121 research papers between 2017 and 2021 on AGS tools for programming education (Messer et al., 2024) concluded that AGSs often only provide feedback on whether unit tests have passed or failed, the difference between expected and actual output, and how the output differs from the reference solution. Such tools rarely provide qualitative feedback on things such as readability or maintainability (Messer et al., 2024), which can negatively impact students' critical thinking and understanding of programming concepts. A possible solution to this problem is to improve the feedback given by AGSs. Haldeman et al. (2021) proposed using a knowledge map that allows teachers to ensure cohesion between the challenges posed by programming assignments and the material taught in the classroom, thereby integrating formative feedback. This enhancement led to a threefold increase in

student error correction (Haldeman et al., 2021). Notwithstanding, such breakages raise issues about AGSs' ability to assess students' knowledge and skills. For instance, considering the description by Hsu et al. of AGSs as "imperfect autograders" (Hsu et al., 2021), one might wonder whether these systems are merely facilitating students to game the system or encouraging students to become more critical users.

In line with previous research such as Hsu et al. (2021) discussing the imperfect nature of AGS as systems, our study also highlighted AGS imperfections in the design and deployment of these systems in practice. More specifically, our study findings pinpointed that such imperfections, or breakages, were found as contradicting tensions between expectations and implications and were visible through the duality of the participants' roles who were found in places to negotiate and balance their technological enthusiasm with their pedagogical goals in assessment practices. For instance, while participants valued AGSs' efficiency for testing basic code functionality, they still had to manually assess non-functional requirements and code quality to maintain pedagogical standards. Furthermore, as Hsu et al. (2021) also discuss, assessment is contextual and relational and therefore acknowledging AGS' imperfections requires a critical consideration of the implications for educators who rely on these systems in their teaching and assessment practices, not least because they can affect the dynamics of the teaching-learning process. Adopting imperfect AGSs requires careful consideration of how these systems affect the role of educators in the classroom. It is imperative to recognise that these systems are tools that should complement, rather than replace, the expertise and judgement of educators. Our participants' dual roles as teachers and developers revealed the importance of engaging teachers in the design, development, and evaluation of AGSs – not only to ensure these systems meet specific educational needs but also to safeguard student learning outcomes, echoing previous calls urging teachers to become "agents of change" (Pantić & Florian, 2015). Collaboration between educators and AGS developers may accelerate the repair and refinement of these systems to meet teachers' practical needs and expectations. Including students in several stages of AGS design, development and deployment processes could also inform AGS research.

Our study makes several key contributions to understanding the role and impact of AI automation in educational assessment, particularly through the case of AGSs. First, our study makes an empirical contribution by examining rich qualitative data on the experiences of teachers on the use and development of AGSs. This dual perspective offers unique understandings into the challenges of implementing AI automation in educational settings. The interview data reveal how AGS experts navigate the tensions between technological promises and assessment practices, contributing to our understanding of technology adoption in education. Second, through our analysis of "breakages" within automation in assessment we contribute to identify the disconnect between idealised technical visions and the complex nature of assessment practices. Rather than viewing breakages as obstacles or flaws, they can be re-contextualised as valuable feedback that acts as practical pointers for AGS developers. This perspective aligns with Bødker's research on breakdowns (Bødker, 1989; Bødker & Grønbæk, 1991), which suggests that disruptions in technological use make users more aware of the tools they use, revealing limitations that might otherwise go unnoticed. Similarly, the breakages in AGSs prompt AGS experts to critically engage with these tools, adapting and negotiating their use in ways that reflect both pedagogical needs and

system constraints. Third, our work advances the STS perspective in educational technology by examining how AI automation affects social interactions in assessment practices. We illustrate how AGSs are not neutral tools but active agents that have histories and values embedded (Fenwick & Edwards, 2016) which reshape the relationship between teachers, students and assessment processes. This connects to broader STS discussions about how educational technologies are "deeply bound in the politics of what it becomes permissible to know and to do" (Fenwick & Edwards, 2016, p. 13). This contribution is particularly relevant as educational institutions increasingly embrace AI automation, raising important questions about agency and the changing role of human judgement in assessment as AGSs actively reshape the socio-material relations through which education happens.

7. Conclusions

Drawing on the concept of breakages, we explored the challenges that AGS might introduce to assessment in computer science education. The participants expected AGSs to help them manage their workload and improve their assessment practices. However, as they engaged in the deployment and use of these systems, tensions emerged, leading to what we termed "breakages". These highlighted the limitations of technologies, provided a window into broader social and cultural contexts, and illustrated how the values and expectations that guide assessment practices were critically challenged.

Our research illustrates that while AI automation through AGSs offers promising solutions for assessment challenges in computer science education, its implementation requires careful consideration of both technical and pedagogical dimensions. The breakages we identified reveal technical limitations as well as fundamental questions about the role of automation in education.

Automated grading systems are a sensitive topic, particularly in the context of applying predictive AI and machine learning techniques to formal assessment situations. We hope to have furthered the understanding of the complexities surrounding their implementation by stressing the importance of carefully weighing the practical implications of deploying AGSs in computer science and programming courses.

Understanding and addressing the implications of AGSs will facilitate the development of more equitable and effective assessment practices in HE. Future researchers might explore strategies for repairing the breakages caused by AGSs and investigate the long-term effects of AGSs use on students' learning experiences and outcomes. This is critical for shaping informed practices that balance technology's benefits while maintaining educational values.

Finally, it is worth reiterating that because our study was limited to computer science education, its findings are not intended to be generalisable to other contexts but rather to inspire future empirical research to explore the impact of AGSs in other contexts as well. Future empirical research that examines the impact of AGSs in other contexts (e.g. different courses, departments, faculties, and institutions) and the types of questions and knowledge that AGSs are designed to assess is needed. This would provide a broader understanding of the impact of AGSs on teaching practice and student learning.

Funding

This work was supported by the Wallenberg AI, Autonomous Systems and Software Program–Humanity and Society (WASP-HS) under Grant MMW 2020-0138.

Notes

1. Please note that the AGSs included in our study have been designed and developed for use within a HE institution and are not publicly available; open data and source code access permissions have not been granted.
2. Our intention here is not to generalise the framing of AGSs as crutches – tools that we know are essential to the mobility of many people. We are simply highlighting the fact that, according to this participant, some students use AGSs to submit iterative versions of an assignment with minimal changes (e.g. altering a single character at a time) rather than opting for more substantial modifications in fewer attempts.

Notes on contributors

Clàudia Figueras is a PhD candidate at the Department of Computer and Systems Sciences at Stockholm University, Sweden. Her research examines how various stakeholders perceive and address ethical issues in the development and use of AI-based systems in public organisations.

Alexandra Farazouli is a doctoral candidate in the Department of Education at Stockholm University, Sweden. Her research examines the ethical implications of emerging AI-driven technologies on assessment practices in higher education. Farazouli's work focuses on how AI and automation shape experiences, influence attitudes, and provoke critical questions within learning environments.

Teresa Cerratto Pargman is a Professor of human-computer interaction at the Department of Computer and Systems Sciences at Stockholm University, Sweden. She is also an associate director for outreach at Digital Futures in Sweden. Currently, she is studying how non-technical university programs become AI-ready. She works from critical perspectives on digital technology.

Cormac McGrath is an Associate professor of Education at the Department of Education at Stockholm University, Sweden. Cormac's interests relate to the intersection between technology and educational practice.

Chiara Rossitto is an Associate Professor of human-computer interaction at the Department of Computer and Systems Sciences at Stockholm University, Sweden. Her work explores questions related to more sustainable futures, social change, and the politics of designing.

Acknowledgments

We thank all the participants for their time and generously sharing their experiences working with AGSs. We also extend our sincere gratitude to the members of the WASP-HS project 'Ethical and Legal Challenges in Relationship to AI-Driven Practices in Higher Education' for their invaluable feedback.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Authors' contributions

The first and second authors contributed equally to the planning, execution and writing of the paper. The remaining authors contributed equally by providing feedback and guidance throughout the research and writing process.

ORCID

Clàudia Figueras http://orcid.org/0000-0003-3812-1492
Alexandra Farazouli http://orcid.org/0000-0001-7601-3850
Teresa Cerratto Pargman http://orcid.org/0000-0001-6389-0467
Cormac McGrath http://orcid.org/0000-0002-8215-3646
Chiara Rossitto http://orcid.org/0000-0001-5819-8302

References

Aldunate, R., & Nussbaum, M. (2013). Teacher adoption of technology. *Computers in Human Behavior*, *29*(3), 519–524. https://www.sciencedirect.com/science/article/pii/S074756321200297X

Baniassad, E., Zamprogno, L., Hall, B., & Holmes, R. , (2021). Stop the (autograder) insanity: Regression penalties to deter autograder overreliance. Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (pp. 1062–1068). Association for Computing Machinery, New York, NY. https://doi.org/10.1145/3408877.3432430

Bearman, M., Boud, D., & Ajjawi, R. (2020). New Directions for Assessment in a Digital World. In M. Bearman, P. Dawson, R. Ajjawi, J. Tai, & D. Boud, (Eds.), *Re-imagining University Assessment in a Digital World* (pp. 7–18 978-3-030-41956-1). Springer International Publishing. https://doi.org/10.1007/978-3-030-41956-12

Bødker, S. (1989). A human activity approach to user interfaces. *Human–Computer Interaction*, *4*(3), 171–195. https://doi.org/10.1207/s15327051hci0403_1

Bødker, S., & Grønbæk, K. (1991). Cooperative prototyping: Users and designers in mutual activity. *International Journal of Man-Machine Studies*, *34*(3), 453–478. https://doi.org/10.1016/0020-7373(91)90030-B

Braun, V., & Clarke, V. (2021, March). Should I use TA? Should I not use TA? Comparing reflexive thematic analysis and other pattern-based qualitative analytic approaches. *Counselling and Psychotherapy Research*, *21*(1), 37–47.

Braun, V., & Clarke, V. (2022). *Thematic analysis: A practical guide*. SAGE.

Burrows, S., Gurevych, I., & Stein, B. (2015). The eras and trends of automatic short answer grading. *International Journal of Artificial Intelligence in Education*, *25*(1), 60–117. https://doi.org/10.1007/s40593-014-0026-8

Byrne, D. (2022). A worked example of braun and clarke's approach to reflexive thematic analysis. *Quality & Quantity*, *56*(3), 1391–1412. doi: https://doi.org/10.1007/s11135-021-01182-y

Cerratto Pargman, T., & McGrath, C. (2021). Be careful what you wish for! Learning analytics and the emergence of data-driven practices in higher education. In S. Petersson (Ed.), *Digital human sciences: New objects - new approaches. Chapter 9* (pp. 203–226). Stockholm University Press. https://doi.org/10.16993/bbk.i

Daston, L., & Galison, P. (1992). The image of objectivity. *Representations*, *40*, 81–128. https://doi.org/10.2307/2928741

Dye, M., Nemer, D., Kumar, N., & Bruckman, A.(2019, November). If it rains, ask grandma to disconnect the nano: Maintenance & care in Havana's StreetNet. Proceedings of the ACM on Human-Computer Interaction (Vol. 3(CSCW). https://doi.org/10.1145/3359289

Edwards, R., & Fenwick, T. (2016). Digital analytics in professional work and learning. *Studies in Continuing Education*, *38*(2), 213–227. https://doi.org/10.1080/0158037X.2015.1074894

Farazouli, A. (2024. Automation and assessment: exploring ethical issues of automated grading systems from a relational ethics approach. In A. Buch, Y. Lindberg, & T. Cerratto Pargman (Eds.), *Framing futures in postdigital education. postdigital science and education*. Springer. https://doi.org/10.1007/978-3-031-58622-4_12

Felt, U., Fouche, R., Miller, C. A., & Smith-Doerr, L., (Eds.) (2016). *The handbook of science and technology studies* (4th ed.). MIT Press; The MIT Press.

Fenwick, T., & Edwards, R. (2016). How sociomaterial approaches could support higher education as a critical practice. In B. Leibowitz, V. Bozalek, P. Kahn (Eds.), *Theorising learning to teach in higher education* (pp. 1–15). Routledge.

Green, B., & Viljoen, S. (2020). Algorithmic realism: Expanding the boundaries of algorithmic thought. Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency; New York, NY. Association for Computing Machinery (pp. 19–31). https://doi.org/10.1145/3351095.3372840;.FAT* '20

Haldeman, G., Babe¸s-Vroman, M., Tjang, A., & Nguyen, T. D. (2021, May). CSF: Formative feedback in auto-grading. *ACM Transactions on Computing Education*, *21*(3). 1–30. https://doi.org/10.1145/3445983

Haldeman, G., Tjang, A., Babe¸s-Vroman, M., Bartos, S., Shah, J., Yucht, D., & Nguyen, T. D. (2018). Providing meaningful feedback for autograding of programming assignments. Proceedings of the 49th ACM Technical Symposium on Computer Science Education (pp. 278–283). Association for Computing Machinery, New York, NY. https://doi.org/10.1145/3159450.3159502

Holmes, W., & Tuomi, I. (2022). State of the art and practice in AI in education. *European Journal of Education*, *57*(4), 542–570. https://doi.org/10.1111/ejed.12533

Hooper, S., Wünsche, BC., Luxton-Reilly, A., Denny, P., & Feng, T. H. (2024). Advancing automated assessment tools - opportunities for innovations in upper-level computing courses: A position paper. In: *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, Portland, OR, USA (pp. 519–525). Association for Computing Machinery. https://doi.org/10.1145/3626252.3630772

Hsu, S., Li, T. W., Zhang, Z., Fowler, M., Zilles, C., & Karahalios, K. (2021). Attitudes surrounding an imperfect AI autograder. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1–15). Association for Computing Machinery, New York, NY. https://doi.org/10.1145/3411764.3445424

Jackson, S. J. (2014). Rethinking Repair. In T. Gillespie, P. Boczkowski, & K. Foot (Eds.), *Media technologies: essays on communication, materiality, and society* (pp. 221–239). The MIT Press.

Jandrić, P., & Knox, J. (2022). The postdigital turn: Philosophy, education, research. *Policy Futures in Education*, *20*(7), 780–795. https://doi.org/10.1177/14782103211062713

Messer, M., Brown, N. C. C., Kölling, M., & Shi, M. (2024, February). Automated grading and feedback tools for programming education: A systematic review. *ACM Transactions on Computing Education*, *24*(1). 1–43. https://doi.org/10.1145/3636515

Miao, F., Holmes, W., Huang, R., & Zhang, H. (2021). *AI and education: guidance for policy-makers* (UNESCO). ISBN: 978-92-3-100447-6. https://doi.org/10.54675/PCSP7350; https://unesdoc.unesco.org/ark:/48223/pf0000376709?locale=en

Nemer, D., & Chirumamilla, P. (2019). Living in the broken city: Infrastructural inequity, uncertainty, and the materiality of the digital in Brazil. In J. Vertesi & D. Ribes (Eds.), *digitalSTS: A field guide for science & technology studies* (pp. 221–239). Princeton University Press. https://www.degruyter.com/document/doi/10.1515/9780691190600/html?lang=en

Nicklin, L. L., Wilsdon, L., Chadwick, D., Rhoden, L., Ormerod, D., Allen, D., Witton, G., & Lloyd, J. (2022). Accelerated HE digitalisation: Exploring staff and student experiences of the COVID-19 rapid online-learning transfer. *Education and Information Technologies*, *27*(6), 7653–7678. https://doi.org/10.1007/s10639-022-10899-8

Page, E. B. (1966). The imminence of … grading essays by computer. *Phi Delta Kappan.47*(5), 238–243. https://www.jstor.org/stable/20371545

Pantić, N., & Florian, L. (2015, January). Developing teachers as agents of inclusion and social justice. *Education Inquiry*, *6*(3), 27311. https://doi.org/10.3402/edui.v6.27311

Pink, S., Ruckenstein, M., Willim, R., & Duque, M. (2018, January). Broken data: Conceptualising data in an emerging world. *Big Data & Society*, *5*(1), 205395171775322. http://journals.sagepub.com/doi/10.1177/2053951717753228

Rao, D. M. (2019). Experiences with auto-grading in a systems course. 2019 IEEE Frontiers in Education Conference (pp. 1–8). IEEE, (FIE); Covington, KY.

Rosner, D. K., & Ames, M. (2014). Designing for repair? Infrastructures and materialities of breakdown. Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing (pp. 319–331); CSCW Association for Computing Machinery, New York, NY. https://doi.org/10.1145/2531602.2531692 '14.

Rosner, D. K., & Fox, S. E. (2016, April). Legacies of craft and the centrality of failure in a mother-operated hackerspace. *New Media and Society*, *18*(4), 558–580. https://doi.org/10.1177/1461444816629468

Selwyn, N. (2022). Less work for teacher? The ironies of automated decision-making in schools. In S. Pink, M. Berg, & D. Lupton, M. Ruckenstein(Eds.), *Everyday automation: Experiencing and anticipating emerging technologies* (1st ed., pp. 73–86. Routledge. https://www.taylorfrancis.com/books/9781003170884

Selwyn, N., Hillman, T., Bergviken Rensfeldt, A., & Perrotta, C. (2023, January). Digital technologies and the automation of education—key questions and concerns. *Postdigital Science & Education*, *5*(1), 15–24. https://doi.org/10.1007/s42438-021-00263-3

Sherman, M., Bassil, S., Lipman, D., Tuck, N., & Martin, F. (2013, June). Impact of auto-grading on an introductory computing course. *J Comput Sci Coll*, *28*(6), 69–75.

Swiecki, Z., Khosravi, H., Chen, G., Martinez-Maldonado, R., Lodge, J. M., Milligan, S., Selwyn, N., & Gašević, D. (2022). Assessment in the age of artificial intelligence. *Computers and Education: Artificial Intelligence*, *3*, 100075. https://www.sciencedirect.com/science/article/pii/S2666920X22000303

Tanweer, A., Fiore-Gartland, B., & Aragon, C. (2016, June). Impediment to insight to innovation: Understanding data assemblages through the breakdown-repair process. *Information Communication & Society*, *19*(6), 736–752. https://doi.org/10.1080/1369118X.2016.1153125

Vattøy, K. D., Gamlem, S. M., Kobberstad, L. R., & Rogne, W. (2022). Students' experiences of assessment and feedback engagement in digital contexts: A mixed-methods case study in upper secondary school. *Education Inquiry*, *0*, 1–22. https://doi.org/10.1080/20004508.2022.2122202

Webb, M., Gibson, D., & Forkosh-Baruch, A. (2013). Challenges for information technology supporting educational assessment. *Journal of Computer Assisted Learning*, *29*(5), 451–462. https://onlinelibrary.wiley.com/doi/abs/10.1111/jcal.12033

Webb, M., & Ifenthaler, D. (2018). Using information technology for assessment: Issues and opportunities. In *Second handbook of information technology in primary and secondary education* (pp. 577–580). Springer International Publishing. https://doi.org/10.1007/978-3-319-71054-91

Appendix: Semi-Structured Interview Guide

Interview Opening

We provide relevant information about our project, the aim of the study, details about the informed consent form, and clarifications on the data collection and storage procedure.

Introduction - Participant's Background

- Could you please share some details about your professional background?
- What is your educational background (field, department)?
- What is your current position and role in the department (%teaching, %research, % administration)?

On AGSs: The Tool and the Need

- Can you tell me what systems or tools you use to automate assessment? What do you call them?
- What is the aim of the system(s)? What does it/do they do?
- In what way are you involved in the system's development? Could you describe that involvement?
- Who else is involved in the system's development? Why did you develop the system?
- How did the development process unfold? Who created it? When? How was the idea for the system initiated? What was it designed to improve or solve?
- Who is/was funding it?
- How would you describe the system's role in the context of a course? Can you give me an example?
- When is the system used?

How AGSs Have Changed the Participant's Work Practice

- Could you please tell me how you use the system or how it is used? For example, is it used for assignments or exams or something else?
- Compared with manual grading (or how you graded before using this system), how does the system change how you assess students' work?
- What would you say are the advantages and disadvantages of grading with this tool?
- Have you experienced any benefits in using this AGS in the context of your grading practice?
- Have you experienced any problems using this AGS in the context of your grading practice?
- Overall, what do you think are the benefits for students?
- Overall, what do you think are the potential risks for students?

Ethical Questions Relating to AGSs

- How do you think using the system affects how you provide student feedback?
- How do you think using the system affects how you communicate with the students? Are the students aware that such a system is in place?
- Do you have any examples of technical challenges that occurred when developing or implementing the system? How did you handle such challenges?

- Do you have any examples of ethical challenges that arose when developing or implementing the system? How did you handle such challenges?
- What do you consider it means to "be fair" in the context of developing an AGS? What motivates you to be fair? Why is that important?
- Do you think students could try to trick the system? How successful do you think they are?

Interview Closing

Participants share any potential additional thoughts, we thank them for taking part, and we give them our contact details.