AI Can Help Instructors Help Students: An LLM-Supported Approach to Generating Customized Student Reflection Responses

Sandra Wiktor
College of Computing and Informatics
University of North Carolina at Charlotte
Charlotte, USA
swiktor@charlotte.edu

Mohsen Dorodchi
College of Computing and Informatics
University of North Carolina at Charlotte
Charlotte, USA
mohsen.dorodchi@charlotte.edu

Nicole Wiktor
College of Computing and Informatics
University of North Carolina at Charlotte
Charlotte, USA
nwiktor@charlotte.edu

Abstract*—This innovative practice paper presents an LLM-supported technique to help instructors respond effectively to periodic students' reflections. Efficient communication between instructors and students is integral to supporting a productive learning environment. Recognizing the significance of understanding students' perceptions and challenges, we present the initial implementation of a system to help instructors analyze and respond to students' feedback promptly and effectively. This research is inspired by and extends prior works where instructors sent progress check emails to students, with some works finding that such communication increased students' motivation. To collect feedback, we administer regular student reflections throughout the semester that capture how students feel about the course and uncover the challenges they face. This regular feedback-gathering approach allows instructors to better track their students' progress and respond to comments throughout the semester to provide guidance. However, reading and responding to each reflection manually in the context of their overall learning experience can be time consuming. To address this challenge, we introduce an LLM-based automated approach that generates tailored, performance-contextualized responses to student reflections that can be used to guide first-contact interventions. The generated reflection responses (GRRs) address issues discussed in student reflections and provide advice, support, course information, and follow-up questions to the students. Additionally, they provide feedback to students based on their accomplishments and behavioral data within the learning management system (LMS), such as submission patterns. In this work, we discuss our method of generating responses based on students' reflections and their LMS behavior. We also present example scenarios of the proposed approach. Preliminary results indicate that this approach can help instructors facilitate positive educational interactions with students and that the participating students view the interventions favorably, fostering a constructive learning environment. This work provides an initial presentation of our large language model-based response generation method to motivate further investigation into AI-assisted student support mechanisms

***Index Terms*—student experience, reflection**

I. Introduction

Facilitating communication between students and the instructor is crucial for the smooth administration of a course. Feedback collected at the end of the semester, mid-semester [1], or intermittently [2], can provide instructors with useful insights into students' experiences in a course. While these feedback responses are sometimes used for course or instructor evaluation [3], they can also contextualize and convey students' experiences throughout the semester to supplement traditional performance indicators such as grades. Mid-semester feedback in particular enables instructors to make timely adjustments to the course during the semester, improving the learning environment and end-of-course evaluations [4]. This information can help instructors make decisions about how they should manage the course, whether by modifying or enhancing some course components that a group of students reported not receiving well, or dealing with individual students' issues. *Emotional* feedback in particular has been of interest to the community, and has been used to predict student outcomes [5] and inform real-time interventions [6]. Instructors can stage individual interventions through one-on-one interactions or personalized emails. Prior studies have demonstrated benefits of personal communication via email. Situational awareness alerts, for example, can improve early submission rates and decrease late submissions [7]. Another study suggests that providing personal messages to students could increase their motivation [8].

To develop effective interventions, it is essential to consider various aspects of students' experiences, as their struggles often stem from different stressors including factors outside of the course, such as a lack of confidence or personal obligations [9]. By responding to students' feedback in a way that addresses their unique circumstances, we can trigger a positive student-instructor interaction. However, parsing through each student's written feedback and metrics of participation or performance to write a custom response for everyone can be time-consuming [10]. To assist this process, we propose enlisting the help of large language models (LLMs) to organize information about students in a course and develop a cohesive initial response.

As LLMs have demonstrated creative writing abilities [11], we can employ them to generate personalized responses,

integrating both traditional performance metrics (like grades) and free-written input (like reflections). In high-enrollment courses, such generated responses can reduce the instructor's burden in creating just-in-time interventions by combining students' information from multiple sources and suggesting ways to address the students' concern.

This work proposes an intervention system that generates responses to reflections with the use of LLMs, referred to in this text as *generated reflection responses* (GRRs). The GRRs are guided by student reflections and course performance to support *first-contact interventions*. In this context, we define *first-contact interventions* as the initial communication initiated by an instructor to assist a student. Rather than serve as the final and only intervention, these first-contact interventions can help lead the instructor and student into productive dialogues.

Specifically, this work contributes a method for designing course interventions using specific course attributes that can be inputted into an LLM. To contextualize this work, we address the following main research question:

- How can LLMs help instructors develop first-contact interventions that can foster positive student-instructor interactions?

In the following sections, we present related work, provide the overall overview of the GRR workflow, and present an implementation of this system in one introductory computer science course we will refer to as CS-A in this text. We refer to the assignment of one set of reflection questions for students to complete as a reflection cycle. We use the symbols R1, R2, ..., R5 to represent the students' reflections for each reflection cycle (i.e, R1 represents the reflections from the first reflection cycle, R2 represents reflections from the second, and so forth). As this system is modular with interchangeable components, the examples provided only serve as a demonstration of how the model can be utilized. The method can be adjusted to accommodate the needs of any instructor and course.

A. Ethical Considerations

This study was approved under IRBIS-21-041. Student names and personally identifiable information are never shared with any large language models or AI tools.

II. Literature

A. Student Feedback and Student Reflections

Literature shows that many critical insights can be found in student reflections and other forms of feedback. Some work suggests that, when paired with professional development, student feedback has the potential to positively impacting teaching practice [12]. One study found that university teachers generally perceive student feedback positively, and student feedback can positively impact teaching and course development [13]. They also consider positive feedback very important to the course, and a majority believe they would not introduce unjustified changes to the course [13]. In one study, researchers used student reflections to evaluate students' level of understanding and analytical thinking in a service-learning course [14]. The students' reflections in this course included those asking students to provide explanations of course concepts and asking what the course revealed about their individual traits like their strengths and weaknesses. They also served to identify and analyze students' objectives. Using the reflections and the model the researchers created, they determined the depth of learning and level of critical thinking [14]. In another study conducted in a computer science course, feature vectors were extracted from reflections to quickly detect students at-risk of receiving a D, F, or withdrawing using text analysis by Linguistic Inquiry and Word Count (LIWC), leading to an accuracy of 90% in detecting at-risk of DFW students within the first few beginning weeks of the course [2]. Another study utilized reflections to understand and find factors that contribute to student engagement in a massive open online course (MOOC) [15]. In general, the research community values student feedback delivered in any form, as it can lead to course improvements.

B. Impacts of Emotions in Learning

Students' emotions can reveal important insights about their learning experience. Emotions play a significant role in learning and other cognitive tasks like attention and reasoning [16] as well as motivation [17]. Thus, understanding and considering the role emotions play in learning is vital. The education research sphere is proliferated with work on emotion detection of students and interventions based on certain emotional expressions, as through affective tutoring systems [18], [19], analyzing the sentiment of MOOC comments [20], and adjusting the learning rate of students working in programming learning environments based on emotional expression [21], [22]. The research community has determined student emotions as relevant to a student's learning experience, and thus, how students feel should be taken into account when developing interventions.

C. Personalized Interventions through Emails

Maintaining ongoing communication between the professor and students is one way to promote students' success in the educational realm [23]. One study suggests that email communication between instructor and student has a direct effect on teaching evaluation and also encourages interpersonal relationships between instructor and student, resulting in a positive influence on teaching evaluation [24]. Education research reveals an interest in different types of email communications between student and professor, such as the study of personal, motivational emails on students [8]. In one study, researchers conducted a study in a archaeology course with undergraduate students and sent them motivational emails relating to how this course could benefit the students, i.e students were told if they followed the strategies stated, their grades would improve. This study found that personal messages to students increased their levels of motivation. Therefore, research shows promising results in the use of email-based interventions.

D. LLM in Education

LLMs have permeated many disciplines, but of particular interest is the impact of these advancements in AI in education. While technologies like ChatGPT have raised ethical considerations about plagiarism, such as when paper writing is assisted or even replaced with the use of ChatGPT [25], such technologies can provide transformative opportunities and advancements in education [26]. LLMs have been used as interactive study guides, assistants for academic writing, and a classroom facilitator, among many other use cases, including LLM classroom chatbots [25]. For example, in one work, researchers tested the application of a GPT model as a virtual teaching assistant [27]. When students were permitted to use ChatGPT via private discussion groups in Teams, they used the AI chatbots for tasks such as tutoring, improving their understanding, and debugging code [28]. One work used LLMs to respond to open-ended questions for students to test their knowledge and provide suggestions for improvement [29]. Another work evaluates how LLM can help instructors grade short textual answers, and determines that it can be a useful tool for this purpose, but necessitates human oversight [30]. For instructors, these models can generate coding tutorials as well as questions for homework and exams [25] [31]. In computer science education, these LLMs provide unique use cases through their ability to generate code. They can help students write code, edit/simplify their existing code, providing clear explanations for what functions the code performs in addition to providing code comments; for instructors, these models can generate coding tutorials as well as questions for homework and exams [25]. These questions can also be customized to an instructor's specific use case. For example, one work used an LLM to generate programming exercises with *context* [31]. The use of large language models to provide feedback to students and support instructors' course development efforts has been well documented in literature, and this work serves as an extension of this research trend.

III. Method

A. Reflections

To collect feedback that provides students with the opportunity to both share their unique experiences in the course and provide an evaluation of the course, we employ the use of free-written student reflections. These student reflections are designed to capture their emotions and the challenges they face.

Reflection is an important part of education, as it can help students deepen their learning process [32]. They can also serve as an avenue of communication between students and instructors [33]. The reflection prompts provide students not only with an opportunity to reflect on their progress so far, preparing them for future material, but also to provide feedback about the course and describe any situation that may be influencing their ability to complete the course. The reflections utilized in this work are centered around two major concepts: how the student is feeling towards the course (*emotion extracting questions*), and what challenges the student experienced during the course (*challenge centered questions*).

Emotion extracting questions (EEQs) consist of two questions, one multiple-choice (EEQ1) and one free-response question (EEQ2). The multiple-choice question asks students how they felt during the course so far and, to answer, provides them a list of emotions from which to choose to illustrate their experience in the course. The emotions students could select in the reflection questions were *excited*, *satisfied*, *neutral*, *confused*, *anxious*, and *frustrated*. Students were also permitted to select multiple emotions to describe their experience. We selected these emotions based on which could potentially provide the most academically relevant information. Research has found that frustration, anxiety, and confusion are among the emotions considered relevant in the academic setting, which informed my choice to select these as the emotions representing negative sentiment [34]. Satisfaction, one of the two positive options, has also been acknowledged as an academic emotion in one study exploring the emotions that freshman college students face [35]. We also included excitement to present two levels of intensity to the positive emotions, which is roughly equivalent to the *joy* or *enthusiasm* included in the preceding study as well. EEQ2 asks students to explain why they felt that way ("Explain why you selected the above choice(s).") EEQ2 provides an avenue for students to discuss any concerns or positive experiences they had during the course.

The *challenge centered questions* (CCQs) consist of three parts:

1) CCQ1: What was your biggest challenge(s) for these past modules?
2) CCQ2: How did you overcome this challenge(s)? Or what steps did you start taking towards overcoming it?
3) CCQ3: Do you have any current challenges in the course so far? If so, what are they?

These questions were used from in similar studies analyzing student reflections [10], upon which this current work builds.

B. GRR System Overview

Figure 1 presents a general overview of the GRR system's workflow, repeated for each student s in the course. First, a student's raw data is inputted into the system, represented by $\mathbf{v}^s_{r,b}$, where r indicates the student's reflections, and b indicates grade-book from the learning management system (LMS) Canvas for each student s. The *Pre-Processing Module* converts the raw data into distinct vectors of student data. The produced vectors are as follows for each student s: $\mathbf{v}^s_r$ (reflections), g^s (current grade), $\mathbf{v}^s_k$ (grades in each syllabus category, explained in Section III-B3), and $\mathbf{v}^s_i$ (participation indicators, explained in Section III-B3). We use these features to build custom *prompts* for each student that are used to guide the LLM's response. By building custom prompts tailored to students' unique needs rather relying on a single-prompt one-size-fits-all approach, we can develop relevant, personalized responses. The final prompt, θ_p is composed of three sub-prompts: a *general sub-prompt* represented by θ^g_p,

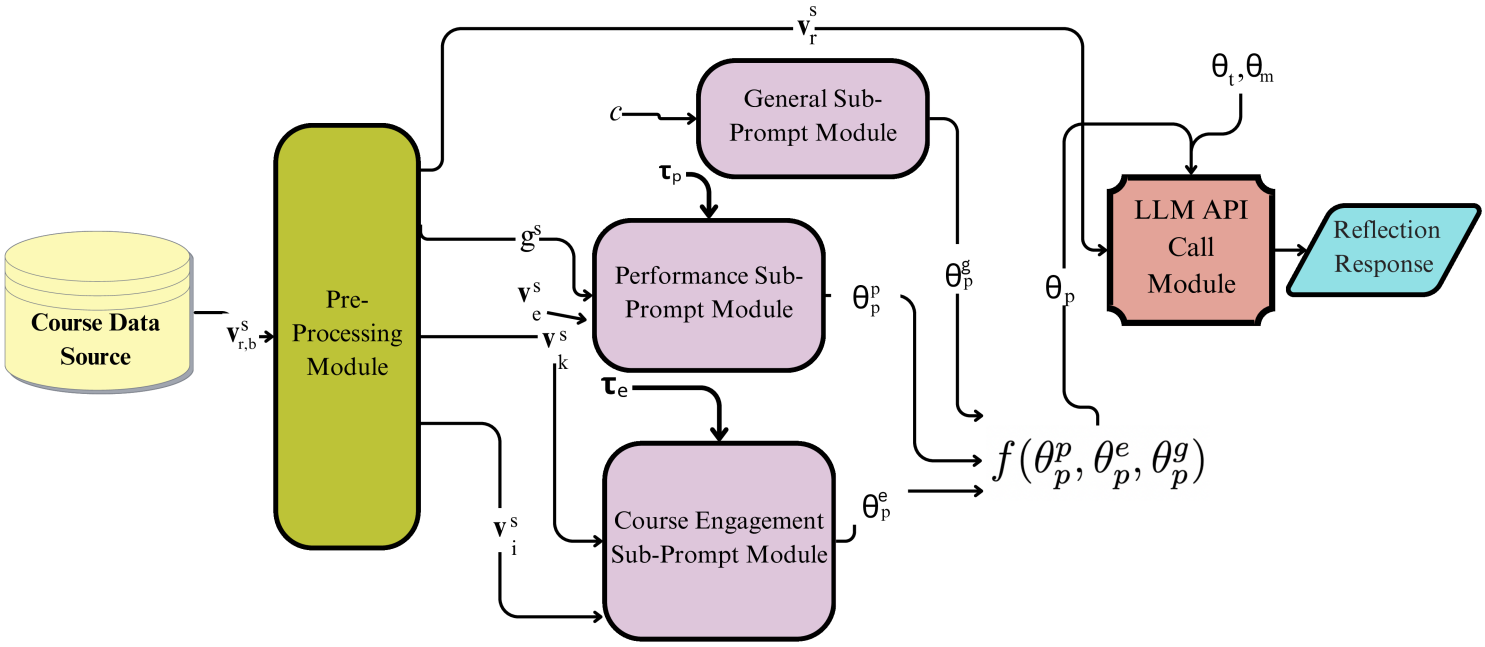

Fig. 1. The basic overview of the GRR Workflow for one student.

a *performance sub-prompt* represented by θ_p^p, and a *course engagement sub-prompt* represented by θ_p^e. These sub-prompts are built in their coinciding modules for each student. While the general sub-prompt remains consistent for all students to define the general task, the latter two are selected based on specific attributes of each student. The final prompt is defined as $\theta_p = f(\theta_p^p, \theta_p^e, \theta_p^g)$ where *f(x,y,z)* concatenates all prompt sub-components in consecutive order. θ_m parameter represents the large language model (LLM) selected, and θ_t parameter represents the temperature. In CS-A (the course in which we implement our case study), θ_m = *gpt-3.5-turbo*, with a θ_t = 0.7. However, instructors can decide which parameters to use.

The GRR prompt is designed to accomplish the following tasks:

- *Support:* Providing immediate support/encouragement to students based on their circumstances;
- *Information Gathering:* Asking follow up questions to any student reflection to provide more information about student challenges to instructors, as students often write short responses;
- *Advising:* Providing immediate suggestions to students to help them address their challenges;
- *Informing:* Bringing awareness to students regarding their current performance (for example, performance on exams, missing projects) and standing in the course.

The GRR prompt accomplishes these objectives through the design of its sub-prompts.

1) General Sub-Prompt: The *general sub-prompt* defines the overall goal and requirements of the response-generation task and is constructed in the *General Sub-Prompt Module*. This sub-prompt is used in each student's final prompt unchanged and focuses on addressing the reflection as a whole. The general sub-prompt also defines possible scenarios (such as *struggling students should come to office hours*) and informs the LLM how it should respond to the scenarios. These elements of the general sub-prompt are considered the *base instruction*. In this sub-prompt, we can also inject *course context*, represented by the scalar c, as an additional instruction. Course context can include specific details about the course at the time the instructor administered the reflection. This context can include current topics the lectures cover or events that have occurred during the course so far that could impact students' responses. The instructor manually added the contextual information, c, that students are replying to Reflection 4. The output of the system, represented by θ_p^g, is the final general sub-prompt, a combination of the base instruction and the course context, c. Table I presents the structure for θ_p^g used in CS-A. The modular format of θ_p^g allows instructors to customize the parameters to their specification. For example, an instructor could choose to change the size of the response (p_l) or add different course context (c).

2) Performance Sub-Prompt: The *performance sub-prompt* (θ_p^p) frames the tone of the email to support students based on their performance, constructed in the *Performance Sub-Prompt Module*. We frame the emails differently for students in different performance categories because students who are at risk for failing the course, for example, may not benefit from the same style of email as students performing well; therefore, utilizing separate prompts can help guide the model towards generating a more appropriate response. This sub-prompt is modified based on the current score input g. The scalar value g represents a student's current single-value overall grade in the course during a specific reflection cycle.

We consider the following categories: *satisfactory*, *needs*

θ_p Instructions	Value
θ_p^g	p_i, please generate a detailed email from the point of view of the instructor that provides support to the student, asks follow up questions about topics and challenges the student describes that are unresolved and provides specific advice to help the student with their problems. p_l. Ask specific questions for the student to answer. c
p_i	Given a student reflection response
p_l	Limit response to a paragraph. Be concise and focus on the most important parts to help students succeed.
c	Start the email by saying that you hope the week/class has been going well and thanking them for responding to Reflection 4.

TABLE I
THE GENERAL PROMPT USED IN GENERATING CS-A GRRS FOR R4.

attention, and *critical*. Table II, demonstrates the grade ranges and instructions used in CS-A. These ranges were subjectively selected by the instructor based on their prior experience working with the students and the threshold for passing for their course. Depending on course requirements, however, the grade ranges for each category may vary.

In Table III, we present the pseudocode for how the sub-prompt is selected. The Performance Sub-Prompt Module receives the τ_p as a vector of thresholds for each performance category. Therefore, $\tau_p = \langle \tau_p^1, \ldots, \tau_p^n \rangle$ where *n* is the number of thresholds. In CS-A, for example, τ_p = $\langle 65, 85 \rangle$ with *n* = 2, but this can be changed when implemented in different courses. In an *n* = 2 configuration, τ_p^1 represents the threshold for satisfactory performance (i.e. all students above the threshold are considered to fall within the *satisfactory* group). τ_p^2 is the threshold that defines the *critical* students (i.e, all students below the threshold are in the critical group). The values between the two thresholds represent the students in the *needs attention* category.

Alongside τ_p, the system takes the vector τ_p and a vector of key-value pairs $\mathbf{v}_e$ defined as $\langle$*Satisfactory* : s_1, *needs attention* : s_2, *critical* : $s_3 \rangle$ where s_n represents a set of instructions for each category. For CS-A, we present these values in Table II. The performance sub-prompt selected for the student is the instruction selected based on their performance category. For students in the *satisfactory performance* range, the prompt is designed to address a student who is performing well in the course and to provide encouragement and reassurance about their progress. For students who *need attention*, the focus of the prompt is to get more information about what the student is struggling with and to provide advice regarding how they can improve their performance. For students who fall into the *critical* range, we emphasize to the student that they are at risk of failing and need to make urgent changes. As in the previous category, one key focus of the prompt is to gather more information about the reason why the student is struggling. θ_p^p represents the selected performance sub-prompt and acts as the output of the Performance Sub-Prompt Module.

3) Course Engagement Sub-Prompt: The course engagement sub-prompt (θ_p^e) provides specific instructions to the model based on student's interaction with the course. This part incorporates two sets of data: $\mathbf{v}_k^s$ (syllabus categories) and $\mathbf{v}_i^s$ (performance indicators). One goal of this sub-prompt is to provide specific instructions to students based on their grades in each *syllabus category*. The final grade students achieve in a course often comprises their scores in specific categories worth a specific percent of the grade, such as homework, or participation, and so forth. In this work, we refer to these as *syllabus categories*.

θ_p^e is essentially a list of customized instructions appended to the prompt based on whether or not students meet a certain threshold. For example, if they scored low on Exam 2, the system can append an instruction like "Come to office hours to go over Exam 1" to the sub-prompt. $\mathbf{v}_k^s$ is a vector of key-value pairs of syllabus categories and the coinciding grades students achieved. $\mathbf{v}_k^s = \langle k_1 : s_1, k_2 : s_2, ..., k_n : s_n \rangle$ where *k* represents each syllabus category, *s* represents the grade for each category and *n* represents the number of syllabus categories in the course. τ_e is a vector that defines the thresholds for each syllabus category. This vector can define, for example, what instruction to append if a student has below 75 in the Exam 2 syllabus category. Instructions for each category if a threshold was crossed is defined as a vector of $\langle c_1 : i_1, ..., c_n : i_n \rangle$ where c represents each category, and i represents the instruction for each category. This vector is hidden from the diagram. The syllabus categories and coinciding instructions used in the CS-A course are presented in the Table IV.

$\mathbf{v}_i^s$ represents a vector of *participation indicators* (PIs). Each PI is accompanied by an instruction, represented through a vector of *PI* condition and *instruction* key-value pairs. Depending on the presence of an indicator within a specific threshold τ_e^i, the model is advised to convey specific messages to students. For example, one PI condition could be "Student has over 5 absences", and the coinciding instruction could be "Tell the student that you are concerned with their attendance and that attendance is required for the course." This instruction is concatenated to the *course engagement sub-prompt*, which can have an unlimited set of instructions. For CS-A, the instructor did not include participation indicators.

Once the three sub-prompts are generated, they are linearly combined through $f(\theta_p^p, \theta_p^e, \theta_p^g)$ to create the final prompt θ_p, which is passed into the LLM API Call Module, alongside the other required parameters, to produce the final reflection response.

IV. EXPERIMENT

To test the usability of the GRRs, we implemented them in an active introductory-level computer science course with 45 students referred to as CS-A. The instructor of the course incorporated the generated responses into personalized emails for all eligible students. The instructor then assigned students to complete five reflection cycles total, and then sent emails

Grade	Context Prompt
Below 65 (Critical, F-D)	Focus the response on being concerned about the student's current progress in the course. Inform the student that they may be at risk of not successfully passing the course. Tell them to reach out and work with the instructor to come up with a plan of action to tackle the rest of the semester. Focus on giving specific advice and asking them to communicate their challenges. Inform the student that they can still submit work that they missed and to provide an option for them to make up the work.
65 - 86 (Needs Attention, D-B)	Ask the student how I can help support them better during the course to finish strong, since the semester is ending soon? Ask the student what other challenges they are facing in the course, as they may have more problems than just what they say in the reflections. Inform them that they may benefit from office hours, which are available to view on Canvas, and they still have the opportunity to turn in late work.
Above 86 (Satisfactory Performance, B-A)	Let the student know they are on track for successfully completing the course. Provide this student with encouragement about their progress and reassure them they are doing well. Let them know they can reach out if they face any challenges in the future.

TABLE II

THE CONTEXT PROMPTS USED FOR GRR-R4. STUDENTS WHOSE CURRENT GRADES FALL WITHIN THESE RANGES RECEIVE THE INDICATED SUB-PROMPT AS PART OF THEIR ENTIRE GRR PROMPT. THE LETTER GRADE IS INCLUDED ALONGSIDE THE LABEL FOR EACH RANGE. THE CONTEXT PROMPT WILL BE HIGHLY DEPENDENT ON COURSE POLICY.

Performance Sub-Prompt Pseudocode
1. procedure **performance_prompt_module**(τ_p, $\mathbf{v}_e$, g)
2. **if** g $> \tau_p^1$ **return** $\mathbf{v}_e$[*Satisfactory*]
3. else if g $< \tau_p^1$ **AND** g $> \tau_p^2$ **return** $\mathbf{v}_e$*[Needs Attention]*
4. else if g $<\tau_p^2$ return $\mathbf{v}_e$[*Critical*]
5. **end if**

TABLE III

PSEUDOCODE OF THE PERFORMANCE SUB-PROMPT. τ_p IS A VECTOR OF THRESHOLDS. G IS THE STUDENT'S CURRENT GRADE. v_i IS A VECTOR OF KEY-VALUE PAIRS OF PERFORMANCE CATEGORY (SATISFACTORY, NEEDS ATTENTION, CRITICAL) AND THE INSTRUCTIONS FOR THOSE CATEGORIES.

based on the GRRs in response to reflections 3 and 4 (referred to as *R3* and *R4*, with the produced GRRs from each reflection cycle referred to as *GRR-R3* and *GRR-R4* consecutively), sending a total of 35 emails over the two reflection periods. R3 and R4 was administered in the second half of the semester. The initial prompt administered on R3 served as a baseline, and was further refined during R4 based on student and instructor reaction. To receive an email, students were required to submit the reflection for each cycle. However, the instructor did not send emails to some students due to logistical or redundancy considerations, such as when students were already in contact with the instructor about a concern at the time. In addition, the instructor of CS-A appended a list of the students' missing assignments to the end of each email.

For R4, the instructor sent most emails with only minor adjustments to alter tone or to remove responses inappropriate to the input, such as instances where the LLM misunderstood students' comments, included unnecessary additions, missed important information, or contained awkward/unnatural phrasing. GRR-R3 required more work to refine, as it was the initial attempt that lead to the development of R4's prompt.

Fig 2 presents an example of the GRR-R4 implementation. This represents instance of the email fully consistent with the instructors' preference, as it was sent to the student without modification. The email demonstrates the capabilities of an LLM-based approach. The LLM accomplished each of the GRR tasks, highlighted in Fig 2's labels, and coherently responded to a student's challenge. The model also seamlessly integrated the participation indicators into the response.

Other GRRs also demonstrated interesting properties. For example, when a student said that it took longer for them to grasp challenges, the model was able to reassure them that this is a common problem. The GRR also provided specific advice such as "summariz[e] each section in your own words." Additionally, the model was able to identify when students did not respond to a specific question and incorporate that understanding through phrasing like *"Since you mentioned that you do not have any current challenges in the course, that's a positive development."* While this phrasing could be improved to sound more natural, this demonstrates the model's ability to extrapolate information independently from the reflection and make assumptions based on context. Another interesting strategy we identified in the GRRs that we did not prompt the GPT model explicitly to do was provide reflective questions to students. For example, one GRR contained the following message: *"Are there any particular strategies or resources you've tried to help overcome this challenge?"* By asking the student to reflect on what strategies they had already tried, the email is encouraging them to investigate the cause or possible solution to their issue, and also support a productive dialogue between instructor and student. Based on these results, we found that the GRRs ability to generate creative emails provides a useful starting point for instructors to build upon, analyze, and modify for the purpose of initiating communication with a student.

The preliminary study suggests that implementing GRRs to an introductory computer science course can help instructors help students. To express the impact of the generated reflection responses, in this section, we present several key examples of positive impact. As part of the final reflection cycle (R5), the instructor asked students to describe their experience with the GRRs to help measure the impact of these responses at a brief glance. In this section, we discuss students' perceptions of the GRRs based on their responses to the final reflection cycle. Out of 45 students, we collected 23 responses to the final reflection assignment from students who agreed to share their responses, which included four multiple choice (MC) ques-

Attribute	Course Engagement Sub-Prompt
Exam 2 Current Score >= 90	Praise a student for doing well on Exam 2!
Exam 2 Current Score <= 75	Ask the student if they would like to come to office hours to go over Exam 2.
Prepwork Current Score <75	Inform the student that prepwork is very important for building foundational knowledge of the concepts for this course. Tell the student they may still be able to get points back for missing prepwork if the student gives me a schedule of when they expect to complete it.
Lab assignments Current Score <75	Inform the student that labs are weighed heavily in this course, and submitting missing labs can greatly improve your grade. Tell them to give you a schedule for when you plan to finish the labs.

TABLE IV
A SAMPLE OF COURSE ENGAGEMENT PROMPT PHRASES WE USED BASED ON CERTAIN STUDENT SUBMISSION BEHAVIORS. THESE PHRASES ARE ADDED TO THE GGR PROMPTS IF THEY FULFILL THE BEHAVIOR CRITERIA.

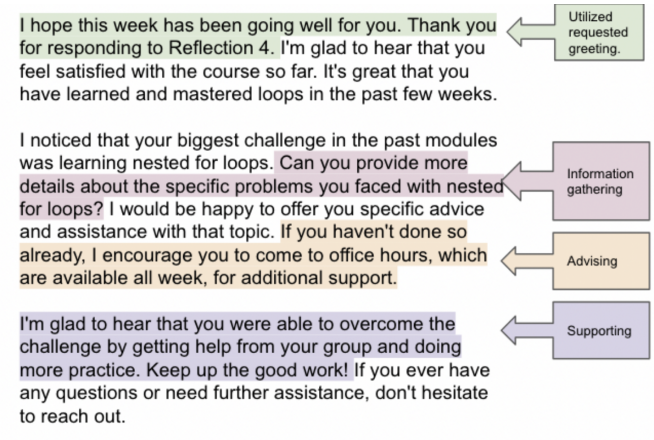

Fig. 2. Example of part of a GRR sent to a student, unedited, via email. While these emails require instructor supervision, in some cases, the response originally cultivated by the model sufficiently reflected the instructors' needs. This response also commented on the students' recent performance.

tions about their experience with the responses, and one free-response question (FRQ) that asked students to provide general feedback about the emails. One MC question asked students if they found the emails useful and 78.2% (18 students) reported they were useful, and 17.4% (4 students) reported they were somewhat useful. Furthermore, another question asked if the emails helped improve their motivation (i.e, to complete work, to succeed) during the course, and 15 (68%) responded that they did, and 22.7% (5 students) reported they somewhat did. 9.1% (2 students) reported that the emails did not impact their motivation. Additionally, nobody selected that the email decreased their motivation. Another question asked whether the emails prompted the student to complete coursework they may have not done otherwise. 59.1% (13 students) reported yes, 18.2% (4 students) reported no, and 22.7% (5 students) reported that the question did not apply to them. Regardless of the students previous responses, all 23 (100%) of respondents noted that the instructional team should send similar emails to students in the future. These reflections demonstrate promising implications for implementing first-contact intervention emails into the curriculum.

While the MC responses demonstrate potentially significant impacts, the FRQ affords students the freedom to uniquely express their experiences and elaborate on their responses. Several relevant themes emerged within the FRQ reflection questions. Firstly, three reflections described an impact on motivation. Students expressed that the email responses motivated them to put effort into the course, complete work on time, learn, and complete assignments. Additionally, six reflections described the responses as a progress check to assess their current standing in the course and noted its use as a reminder system for resources and assignments. Four reflections also described the impact of the responses on the students' ability to seek help. For example, students appreciated that the responses were personalized to target their specific area of difficulty, and that they provided useful advice/tips and resources to students. Several students described that these emails provided an open space/opportunity for students to request help, and one student mentioned the responses prompted them to attend office hours. In addition to academic support, six reflections suggest that the responses can impact students' perception of the instructor. Students enjoyed the personal one-on-one contact and expressed that it made them feel cared for and feel more integrated in the class. They also expressed appreciation that the professor was aware of their difficulties, was open to communication, and was willing to help. While there were a few critical comments about the reflection responses – for example, some students noted that they did not find them useful – these students expressed appreciation for them nonetheless or noted that they could see other struggling students benefit from them. Overall, the students expressed a positive perception of the responses, and recommend they are continued in the future.

In addition to the feedback expressed by students, the instructor also identified interactions producing to learning while administering the GRRs. When the instructor administered the 34 GRRs via email, they received 16 replies total. Amongst these interactions, the instructor described one case where the first-contact interventions lead into a discussion about higher education opportunities, and other cases where the students updated the instructor about their progress in the course; in one such case, a student explained which elements of the course resulted in their improved understanding. The instructor also reported that 62% of the students responded to either express appreciation for the email itself or the instructor and their teaching style.

V. LIMITATIONS

This work provides an early first-look at a reflection response generation model, so there is room for improvement

and development. For example, we only administered emails for reflection cycles 3 and 4, near the end of the semester, rather than the entire course. Additionally, the language model technology generates text based on the probabilistic relationships between words and may make significant errors in delivering new information. We observed that the current GRRs do not always fulfill all prompted requirements and cannot always address all the nuances of student reflections, though we have found that it does provide a good starting point for an instructor to administer a first-contact intervention. To mitigate the impact of these errors, in its current form, the instructional team must inspect the emails before sending them.

We also recognize the narrow scope and limited number of experiments of this initial investigation, with only a single class of 45 students. Additionally, we need to investigate whether the emails would impact students differently if they were explicitly made aware that the emails were supported by an AI tool. Since the instructor of this experiment read and modified the reflections and generated responses to meet the needs of each student, they are still aware of student challenges, facilitating enhanced student-instruction communications; students are not speaking, unregulated, to an AI. Moreover, any subsequent interactions between instructor and student do not use AI. In future work, we intend to develop a model with a higher level of reliable automation, as through a dialogue system (chatbot). Students would be made aware that they are interacting with an AI system for such a method.

Finally, the interventions are based primarily on the content and quality of student reflections, which could affect the usefulness of the reflection. For poor quality reflections, the model has less context produce a useful response. Furthermore, if a student uses AI to generate responses to the reflection, the intervention would be less valuable. However, the instructor has not identified such cases. Moreover, the scope of the GRR intervention in this version of our work does not encompass struggling students who failed to complete the reflections, though such students in the experiment were provided support outside the reported study.

A. Evaluation

In this work, we use students opinion to evaluate usefulness of GRRs after instructor refinement. This work lacks, however, a formal evaluation of the responses generated by the model directly. In future work, we intend to formalize an evaluation method based on existing literature to measure the ability of the model to align with instructors' preferences. Similar works leveraging LLMs in educational contexts, such as creating virtual TAs [36], dialogue systems to explain educational path recommendation suggestions [37], and generating coding exercises [38] typically leverage a mix of human and automatic evaluation. In [36], researchers compare answers to student questions crafted by human TAs and virtual TA and ask course alumni to evaluate the responses based on engagement, clarity, and accuracy. [37] reports on a user study to measure the outputs of responses on metrics like perception of correctness and quality of answers on a Likert scale. We can use similar techniques to evaluate our GRRs in the future.

VI. Conclusion and Future Work

In this work, we propose an LLM-based framework to produce generated reflection responses (GRRs) that help instructors develop first-contact email interventions. To define the scope of the intervention, we identify four major tasks for the GRR to accomplish: support, information gathering, advising, and informing. We incorporate these task descriptions into the prompt to generate the response templates. Following our study, we have determined that our GRR prompts and framework produce coherent and useful email templates that consider both objective factors from students' behavioral data (e.g. students' submissions) and subjective student reflection responses. In this initial study, we observe positive outcomes that resulted from deploying GRR. For example, our work provides preliminary evidence to suggest the produced emails may increase some students' motivation, provide students with a space to communicate their needs and experiences, influence help-seeking behavior, and produce other positive behavioral outcomes. To provide further evidence towards the impact of this method, we intend to facilitate a larger study in the future by administering reflections on a larger pool of students across numerous classes using a refined GRR framework. We can refine the framework by training it to produce more specialized outputs and generate more meaningful suggestions, such as recommending specific resources. Furthermore, we can develop a more well-defined evaluation system. In future work, we can also experiment with different modes of delivery. For example, we can consider developing a conversational agent system that identify students needs through a dialogue, rather than a single email. In this paper, we present our findings as preliminary evidence to justify a deeper investigation of this area.

References

[1] E. M. Sozer, Z. Zeybekoglu, and M. Kaya, "Using mid-semester course evaluation as a feedback tool for improving learning and teaching in higher education," *Assessment & Evaluation in Higher Education*, vol. 44, no. 7, pp. 1003–1016, 2019. [Online]. Available: https://doi.org/10.1080/02602938.2018.1564810

[2] M. Dorodchi, A. Benedict, D. Desai, M. J. Mahzoon, S. MacNeil, and N. Dehbozorgi, "Design and implementation of an activity-based introductory computer science course (cs1) with periodic reflections validated by learning analytics," in *2018 IEEE Frontiers in Education Conference (FIE)*, 2018, pp. 1–8.

[3] J. T. E. Richardson, "Instruments for obtaining student feedback: a review of the literature," *Assessment & Evaluation in Higher Education*, vol. 30, no. 4, pp. 387–415, 2005. [Online]. Available: https://doi.org/10.1080/02602930500099193

[4] S. Wickramasinghe and W. Timpson, "Mid-semester student feedback enhances student learning," *Education for Chemical Engineers*, vol. 1, no. 1, pp. 126–133, 2006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1749772806700173

[5] D. Chaplot, E. Rhim, and J. Kim, "Predicting student attrition in moocs using sentiment analysis and neural networks," vol. 1432, 06 2015.

[6] S. Chen, J. Dai, and Y. Yan, "Classroom teaching feedback system based on emotion detection," *2019 9th International Conference on Education and Social Science*, May 2019.

[7] S. H. Edwards, J. Martin, and C. A. Shaffer, "Examining classroom interventions to reduce procrastination," in *Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*, ser. ITiCSE '15. New York, NY, USA: Association for Computing Machinery, 2015, p. 254–259. [Online]. Available: https://doi.org/10.1145/2729094.2742632

[8] C. Kim and J. M. Keller, "Effects of motivational and volitional email messages (mvem) with personal messages on undergraduate students' motivation, study habits and achievement," *British Journal of Educational Technology*, vol. 39, no. 1, pp. 36–51, 2008. [Online]. Available: https://bera-journals.onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8535.2007.00701.x

[9] A. Salguero, W. G. Griswold, C. Alvarado, and L. Porter, "Understanding sources of student struggle in early computer science courses," in *Proceedings of the 17th ACM Conference on International Computing Education Research*, ser. ICER 2021. New York, NY, USA: Association for Computing Machinery, 2021, p. 319–333. [Online]. Available: https://doi.org/10.1145/3446871.3469755

[10] M. Dorodchi, A. Benedict, E. Al-Hossami, A. Quinn, S. Wiktor, A. Benedict, and M. Fallahian, "Clustering students' short text reflections: A software engineering course case study (full paper)," in *Joint Proceedings of the Workshops at the International Conference on Educational Data Mining 2021 co-located with 14th International Conference on Educational Data Mining (EDM 2021), Held Virtually, 2021*, ser. CEUR Workshop Proceedings, T. W. Price and S. S. Pedro, Eds., vol. 3051. CEUR-WS.org, 2021. [Online]. Available: https://ceur-ws.org/Vol-3051/CSEDM_4.pdf

[11] J. Huang and M. Tan, "The role of chatgpt in scientific communication: Writing better scientific review articles," Apr 2023. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC10164801/

[12] L. Mandouit, "Using student feedback to improve teaching," *Educational Action Research*, vol. 26, no. 5, pp. 755–769, 2018. [Online]. Available: https://doi.org/10.1080/09650792.2018.1426470

[13] J. Floden, "The impact of student feedback on teaching in higher education," *Assessment & Evaluation in Higher Education*, vol. 42, no. 7, pp. 1054–1068, 2017. [Online]. Available: https://doi.org/10.1080/02602938.2016.1224997

[14] L. M. Molee, M. E. Henry, V. I. Sessa, and E. R. Mckinney-Prupis, "Assessing learning in service-learning courses through critical reflection," *Journal of Experiential Education*, vol. 33, no. 3, pp. 239–257, 2011. [Online]. Available: https://doi.org/10.1177/105382590113300304

[15] C. . T. Y. Hew, Khe Foon ; Qiao, "Understanding student engagement in large-scale open online courses: A machine learning facilitated analysis of student's reflections in 18 highly rated moocs," *International Review of Research in Open and Distributed Learning*, vol. 19, no. 3, 2018.

[16] E. Hudlicka, "To feel or not to feel: The role of affect in human-computer interaction," vol. 59, 2003, pp. 1–32.

[17] J. Plass and S. Kalyuga, "Four ways of considering emotion in cognitive load theory," *Educational Psychology Review*, vol. 31, 06 2019.

[18] Z. A. Pardos, R. S. Baker, M. San Pedro, S. M. Gowda, and S. M. Gowda, "Affective states and state tests: Investigating how affect and engagement during the school year predict end-of-year learning outcomes," *Journal of Learning Analytics*, vol. 1, no. 1, pp. 107–128, May 2014. [Online]. Available: https://learning-analytics.info/index.php/JLA/article/view/3536

[19] N. Thompson and T. McGill, "Genetics with jean: the design, development and evaluation of an affective tutoring system," *Educational Technology Research and Development*, 08 2016.

[20] Z. Liu, W. Zhang, J. Sun, H. N. Cheng, X. Peng, and S. Liu, "Emotion and associated topic detection for course comments in a mooc platform," in *2016 International Conference on Educational Innovation through Technology (EITT)*, 2016, pp. 15–19.

[21] R. Zatarain Cabada, M. Barron Estrada, J. García Lizarraga, G. Muñoz-Sandoval, and J. Ríos, "Java tutoring system with facial and text emotion recognition," *International Journal of Advanced Research in Computer Science*, vol. 106, p. 49, 10 2015.

[22] M. L. Barrón Estrada, R. Zatarain Cabada, R. Oramas Bustillos, and M. Graff, "Opinion mining and emotion recognition applied to learning environments," *Expert Systems with Applications*, vol. 150, p. 113265, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417420300907

[23] A. Khan and S. Khan, "Communication skills of a teacher and its role in the development of the students' academic success," *Journal of Education and Practice*, vol. 8, 01 2017.

[24] V. C. Sheer and T. K. Fung, "Can email communication enhance professor-student relationship and student evaluation of professor?: Some empirical evidence," *Journal of Educational Computing Research*, vol. 37, no. 3, pp. 289–306, 2007. [Online]. Available: https://doi.org/10.2190/EC.37.3.d

[25] J. Meyer, R. Urbanowicz, P. Martin, K. O'Connor, R. Li, P.-C. Peng, T. Bright, N. Tatonetti, K. Won, G. Gonzalez, and J. Moore, "Chatgpt and large language models in academia: opportunities and challenges," *BioData Mining*, vol. 16, 07 2023.

[26] X. Zhai, "Chatgpt and ai: The game changer for education," 03 2023.

[27] M. Liu and F. M'Hiri, "Beyond traditional teaching: Large language models as simulated teaching assistants in computer science," in *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, ser. SIGCSE 2024. New York, NY, USA: Association for Computing Machinery, 2024, p. 743–749. [Online]. Available: https://doi.org/10.1145/3626252.3630789

[28] J. Rajala, J. Hukkanen, M. Hartikainen, and P. Niemelä, ""čall me kiran

[29] J. K. Matelsky, F. Parodi, T. Liu, R. D. Lange, and K. P. Kording, "A large language model-assisted education tool to provide feedback on open-ended responses," 2023.

[30] J. Schneider, B. Schenk, C. Niklaus, and M. Vlachos, "Towards llm-based autograding for short textual answers," 2023.

[31] A. Del Carpio Gutierrez, P. Denny, and A. Luxton-Reilly, "Evaluating automatically generated contextualised programming exercises," in *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, ser. SIGCSE 2024. New York, NY, USA: Association for Computing Machinery, 2024, p. 289–295. [Online]. Available: https://doi.org/10.1145/3626252.3630863

[32] K. Hinett, *Improving learning through reflection—part one*. New York, NY: Higher Education Academy, 2002.

[33] H. Machost and M. Stains, "Reflective practices in education: A primer for practitioners," *CBE Life Sci Educ*, vol. 22, no. 2, p. es2, Jun 2023.

[34] R. Pekrun, "Emotions and learning; educational practices series; vol.:24; 2014," 2014.

[35] T. Hailikari, R. Kordts-Freudinger, and L. Postareff, "Feel the progress: Second-year students' reflections on their first-year experience," *International Journal of Higher Education*, vol. 5, no. 3, pp. 79–90, 2016.

[36] M. Liu and F. M'Hiri, "Beyond traditional teaching: Large language models as simulated teaching assistants in computer science," in *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, ser. SIGCSE 2024. New York, NY, USA: Association for Computing Machinery, 2024, p. 743–749. [Online]. Available: https://doi.org/10.1145/3626252.3630789

[37] H. Abu-Rasheed, M. H. Abdulsalam, C. Weber, and M. Fathi, "Supporting student decisions on learning recommendations: An llm-based chatbot with knowledge graph contextualization for conversational explainability and mentoring," 2024. [Online]. Available: https://arxiv.org/abs/2401.08517

[38] A. Gutierrez, P. Denny, and A. Luxton-Reilly, "Evaluating automatically generated contextualised programming exercises," 03 2024, pp. 289–295.