Designing an LLM-based Multiagent System for Generating Activities and their Rubrics: A Study on Data Mining

Eryck Silva[1], **Julio Cesar dos Reis**[1]

[1]Universidade Estadual de Campinas (UNICAMP)

eryck@unicamp.br, jreis@ic.unicamp.br

Abstract. *Assessment is the primary way in which instructors evaluate students' progress. However, the development of high-quality assessments and their corresponding rubrics requires a significant workload from instructors. In this context, Artificial Intelligence can be explored to assist in co-creating assessments and rubrics. This study proposes MASGAR, a multi-agent system designed to create activities and rubrics. We define the system's architecture and employ a simulated test study to assess the viability of MASGAR in a Data Mining course by generating two activities and their rubrics. Results indicate that co-creation is essential for conveying human experiences and leveraging LLM-based systems in educational contexts. Students' feedback from the course revealed that activities were coherent and creative, and suggested criteria for improvements.*

1. Introduction

Elaboration and application of assessments are one of the primary methods used by instructors to check if students are reaching the learning objectives [Bloom et al. 1971, Scriven 1967]. These assessments can be either formative (measures learning progression) or summative (measures learning outcomes). In light of this, these educational artifacts must be constructed clearly and concisely, ensuring that they are relevant to students' context because ill-elaborated assessments can negatively impact students' learning progress [Kinnunen and Simon 2012]. For instance, students can escalate the perceived difficulty of an assignment that is not constructed correctly, leading to poor performance and engagement [Linnenbrink and Pintrich 2003].

Rubrics are a set of rules that describe how a summative assignment will be corrected. The development of rubrics is encouraged because it provides a clear and transparent way to evaluate assignments [Lancaster et al. 2019]. However, digital tools that apply rubrics in an automated way require these rules to be made available by specialists, and, by doing so, they do not become clear for students [Izu and Mirolo 2024, Keuning et al. 2021]. We observe that it is imperative to generate both well-formed summative assessments and their rubrics to promote a significant educational context.

Artificial Intelligence (AI) systems can now be explored to assist and enhance teaching and learning, although Computer Assisted Instruction and Intelligent Tutoring Systems first appeared in the 1970s [Carbonell 1970]. AI plays a crucial role in automating processes, analyzing large volumes of data, and providing personalized solutions.

Particularly, Generative AI (GenAI) excels in the field of text generation, as it creates original and coherent textual information [Jo et al. 2023]. This technology is utilized in conversational interfaces, such as chatbots, to support various domains, including

DOI: 10.5753/sbie.2025.12723

education, healthcare, and business [Bahroun et al. 2023]. In education, GenAI offers numerous possibilities, including automating assessment creation and correction, as well as generating formative feedback for students. When considering rubric development and use, integrating them into GenAI could assist both instructors and students.

Large Language Models (LLMs) are an evolution of language models, which utilize statistical methods to predict the next word in a sequence. These models enhance GenAI's capabilities in textual processing and generation. Moreover, several refining techniques are applied to LLMs, such as adding examples of expected answers in the prompt, known as few-shot learning [Brown et al. 2020].

Despite their employability, LLMs are hindered when they are requested in domain-specific scenarios or need to acquire information in real-time. When this happens, the model's answers become less precise or even false [Zhao et al. 2023]. One way to circumvent this is the development of LLM-based Agents [Wang et al. 2023]. These agents are autonomous entities that perceive and respond to events in an environment to reach a specific goal [Russell and Norvig 2016, Jiang et al. 2024, Wang et al. 2024] by using LLMs as their "brain" in decision making. Agents are designed as specialists in a given task, without the need for any specific training [Wang et al. 2024] and are instructed in natural language [Wang et al. 2023]. A system is considered multiagent when it employs multiple agents with specific functions, requiring communication patterns among them [Becker 2024].

GenAI technologies offer several possibilities in education, and recent studies often employ rubrics for automated correction. However, these rubrics are generally developed by specialists beforehand [Phung et al. 2023, Duong and Meng 2024]. Since the development of both summative assessments and rubrics can be time-consuming, leveraging an LLM-based solution to assist in these processes would be beneficial in easing the workload.

This study proposes MASGAR, a novel LLM-based multi-agent system for developing summative educational artifacts (Activities) and their corresponding rubrics in conjunction. MASGAR assists instructors and teaching assistants (TA) by co-creation, that is, using both human experience and LLM-empowered intelligent agents to enhance educational contexts. The study aims to answer the following research questions:

RQ1: How can the development of summative assessments and their rubrics be orchestrated in an LLM-based multi-agent system?

RQ2: How can the system developed in RQ1 be used in a given educational environment?

Our contributions include an architectural design for MASGAR, composed of five specific agents that interact with one another and the user to elaborate and evaluate summative educational artifacts. We conducted an experimental evaluation of MASGAR's workflow in a Data Mining course by leveraging LLMs in the co-creation process. Our results suggest that these models can be utilized to support instructors and teaching assistants. Results also indicate that the output from LLM models still requires adaptation from humans, as the instructions are sometimes unclear or ambiguous. It is also stated that, in terms of rubric development, these models may benefit from specific training, as they may not have a clear understanding of rubric formats.

The remaining of this article is organized as follows: Section 2 presents related work; Section 3 describes MASGAR's architecture, eliciting its components and agents. Section 4 details the experimental evaluation methods; Section 5 describes the obtained results. Section 6 presents a discussion of our findings and limitations. Section 7 concludes this study and presents future work.

2. Related Work

Rockembach and Thom (2024) investigated how LLMs could be employed to automate question generation within the context of Business Process Modeling (BPM) [Aguilar-Savén 2004]. The authors conducted a study within an undergraduate BPM class in which both GPT-3.5-Turbo[1] and Llama-2[2] to automatically create questions within different levels of the Revised Bloom's Taxonomy [Krathwohl 2002]. The exploratory case study revealed the relevance that question prompts, which are the expected question model present in the prompt, have especially when using less powerful LLM models, such as Llama-2. In a similar research, Chico *et al.* (2024) explored how refining models could develop Multiple-Choice Questionnaires (MCQ) in an encoder-decoder architecture to generate the question and its alternatives, both correct and incorrect, as separate tasks. The authors used didactic materials as input text to create the context. The results indicated that whether the proposed architecture is suited for this task, the generation of alternatives can be hindered by the limited context provided by the input.

Other studies have employed LLMs in classroom contexts to assess models' capabilities. For instance, Martins *et al.* (2023) used OpenAI's ChatGPT to investigate whether the model could solve Natural Deduction exercises in propositional logic. The authors argued that while generally powerful, ChatGPT did not perform well on the series of exercises, thus indicating that a refined model would be better suited for this purpose. On the other hand, Villa *et al.* (2024) developed CoderBot, a pedagogical agent, created within the Example-Based Learning paradigm, designed to assist students in Introductory Programming. Both studies used these LLMs as chatbots, harnessing the paradigm of conversational task-solving to interact with the models. These studies are essential for assessing the strengths and weaknesses currently present in these digital technologies, as well as for identifying key aspects that require enhancement.

Concerning automating correction by using predefined rubrics, Kumar and Boulanger (2021) conducted a study referred to as essay correction. The authors aimed to establish a concise framework that presented a high level of agreement with human raters (this level was usually measured via a Quadratic Weighted Kappa (QWK)). The framework used feature-based deep learning to predict rubric scores, refining a previous work. Their work has achieved a QWK of 0.78, a competitive indicator when compared to cutting-edge similar systems. However, this study used already predefined rubrics to predict the essays' scores.

To the best of our knowledge, research on automated rubric development is somewhat scarce. For example, Alves *et al.* (2020) conducted a study to propose rubrics to identify the originality of a product. Their research targeted the education of Algorithms via the assessment of creativity and originality of computational artifacts developed by K-

[1] `https://platform.openai.com/docs/models/gpt-3.5-turbo`
[2] `https://www.llama.com/llama2/`

12 students using pre-established frameworks, such as App Inventor [Wolber et al. 2011]. While the authors' work was essential to paving the way for rubric development and assessment of computational skills, the proposed rubric was created manually.

The somewhat "lack" of rubric use and development is reinforced, although limited to Introductory Programming, by the systematic literature review conducted by Lima *et al.* (2024) . The process analyzed over 231 articles with seven being selected for in-depth analysis. The authors argued that rubrics are seldom employed to support regulatory aspects in programming learning.

Our present study aims to provide the foundation for the automated development of rubrics using LLM-based agents as a co-creation tool. Since manual development is generally considered difficult, requiring the aid of professionals [Alves et al. 2020, Lima et al. 2024], we believe it can ease this burden that may hinder its use in general classrooms. Moreover, the co-creation of rubrics can facilitate the application of automated correction, thereby advancing studies in this area of GenAI in education.

3. MASGAR

MASGAR refers to our proposed **M**ulti-**A**gent **S**ystem for **G**enerating **A**ctivities and their **R**ubrics. MASGAR's primary purpose is to assist students and teaching assistants in developing educational artifacts through co-creation. More specifically, through co-creation, it is expected that both human and LLM agents interact with each other to produce the final product, which, in this case, is Activities and their corresponding rubrics.

We define an Activity as an assignment that aims to evaluate different competencies within a given subject. For this purpose, an Activity is composed of a series of Tasks, which, for instance, are smaller objectives that must be completed sequentially to reach a final goal. Rubrics are generally defined as a fixed set of rules designed to assess a summative assignment. These rules need to be created by providing precise and concise descriptions of the performance of the evaluated artifact [Lima et al. 2024]. The use of rubrics is not only targeted at the instructor or TA for evaluation, but students can also benefit from them to better develop their solutions.

Figure 1 presents the general architecture designed for MASGAR. It depicts how the end-user interacts with the system to request an Activity. The process is mediated by a user interface (UI), such as a chatbot powered by LLM-based agents. Dark green arrows symbolize the interaction between humans and MASGAR, whereas purple arrows represent how artificial agents from different contexts communicate with each other. Black lines represent interactions with agents from the same context.

3.1. Human Interaction

The instructor or TA should interact with MASGAR via the UI. In this phase, the user can pass detailed instructions on how the Activity and rubric should be created. The system's purpose is to be agnostic of any subject. Here, it is possible to include didactic materials, such as videos, audio files, or documents, to create contexts for the agents. A specific agent, denominated as *Supervisor*, mediates the information passed between the user (via UI) and the other remaining LLM-based agents in the system.

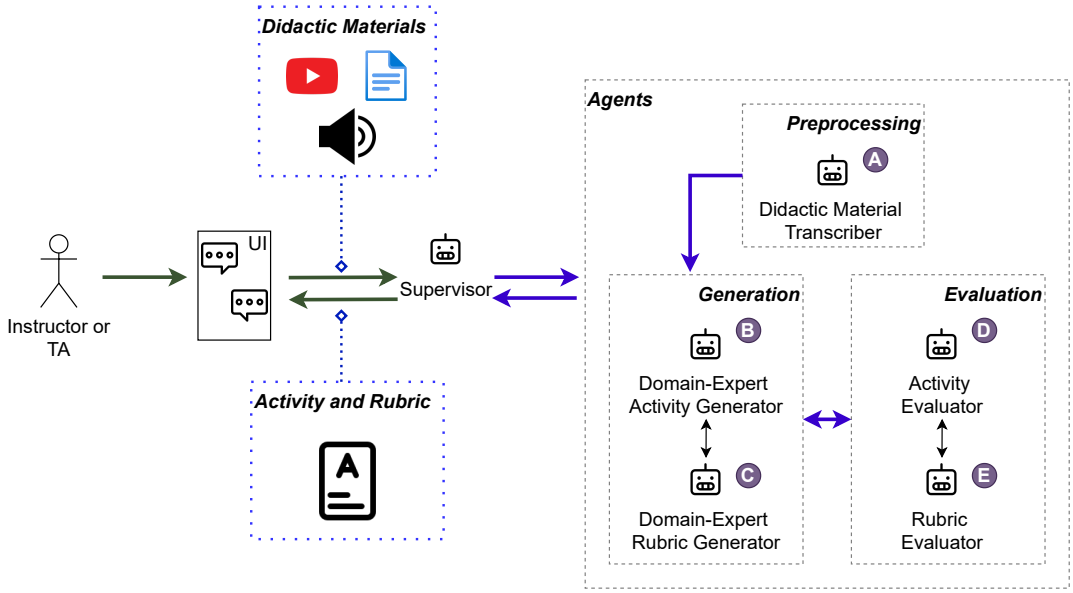

Figure 1. MASGAR's general architecture.

3.2. Agents Interaction

MASGAR is designed with the concept of expert intelligent agents that, as the name suggests, excel at a particular task. For the generation of Activities and Rubrics, a total of five agents were envisioned as follows:

- The **Didactic Material Transcriber** (A) is responsible for parsing any didactic material that the user uploads via the UI. Transcription refers to the process of converting a specific media file into textual output, specifically for use with LLMs. Leaving this responsibility to an agent helps the user avoid having to use transcribing tools (such as *MoviePy*, *SpeechRecognition*, or *PyMuPDF*) beforehand.
- The **Domain-Expert Activity Generator** (B) is responsible for generating the Activity. This agent is informed of any domain specified by the transcribed didactic materials, as well as other requests provided by the user (such as the number of Tasks or difficulty setting). This agent works collaboratively with the **Domain-Expert Rubric Generator** (C), which is responsible for generating rubrics for each Task that composes the Activity.
- The **Domain-Expert Activity** (D) and **Rubric** (E) **Evaluators** are agents responsible for conducting a preliminary review of these artifacts before sending them to the user. Literature often refers to generated questions that, unintentionally, include their answers within the question, especially multiple-choice ones [Chico et al. 2024]. Agents D and E are designed to identify these issues by checking questions and answers, and then request adaptations from Agents B and C.

3.3. Architectural Details

Although MASGAR has not been fully implemented, some key frameworks have already been selected for use. The agentic framework will be derived from LangGraph[3], which

[3]https://langchain-ai.github.io/langgraph/

is gaining recent popularity for the development of more robust applications that employ LLM agents. These same features are also why the UI is pre-established to follow a chatbot design, leveraging options for prompt writing and uploading didactic materials.

4. Experimental Evaluation

We evaluated the viability of MASGAR by applying the system's work within an educational context. A small exploratory study was conducted during the offering of a Data Mining course. We present the methods employed during this study. We begin by providing details on how the course was taught. We then detail the creation of activities, Tasks, and their respective rubrics. Lastly, we describe how we implemented feedback to assess the small study. We notice that this experimental evaluation was designed by the instructor at their discretion to enhance teaching and learning for all students. Moreover, the institutional program expects instructors to ask for students' feedback at the end of each course. Based on this, ethical consent to conduct this research was not required from an Ethics Research Committee.

4.1. The Data Mining Course

The Data Mining course is part of an Engineering and Management of Databases program from an IT School located in São Paulo. The program is targeted at students who have already graduated and are seeking to complement their knowledge in a specific field. Courses are remotely ministered throughout a set number of Saturdays (typically three), having an average of seven hours per week. Each instructor is responsible for determining their summative procedures. The main researcher of this study acted as a teaching assistant during one offering of the course in Fall 2025.

When the study was conducted, the course had the following syllabus: Introduction to Data Discovery and Treatment, Data Selection and Evaluation Techniques for Training, Introduction to Python, NumPy, and Pandas, Data Visualization, Data Classification Algorithms, Data Grouping Algorithms, and Evaluation Methods. Students' grading was divided into three summative artifacts: a set of MCQs; one Activity dedicated to Data Classification (A1); and one Activity focused on Data Grouping (A2).

The course was taught in Brazilian Portuguese. Classes were ministered remotely in a synchronous format, recorded, and made available for further consultation. Students could (and were encouraged to) interact at any given time. Regarding the summative assessments, both A1 and A2 could be done individually or in pairs.

4.2. Activities and Rubrics Generation

The execution of MASGAR was simulated using OpenAI's ChatGPT[4] as a medium to test Activity and Rubric generation. All requests were made using the "GPT-4-Turbo" model. Two different approaches were followed, one for A1 and the other for A2, to assess various scenarios of co-creation. A1 and A2 were not generated in the same chat window.

[4]https://chatgpt.com/

4.2.1. Activity 1 (A1) - Classification

For A1, the approach targeted evaluating whether the LLM model could create Tasks by using only the knowledge it was trained on, that is, no extra theoretical didactic material would be passed. A1 aimed to assess Binary Decision Trees. The co-creation process involved requesting the LLM by both using a specific prompt and providing two additional files for context: a previously curated dataset, selected by the instructor; and a previous Activity, from another course offering, to serve as a document base. The model was expected to use both its training knowledge and the curated dataset to create the Tasks.

The document base served as a template for the Activity. It has specified fields that define title, scenario, objectives, a list of Tasks, dataset, tools, and evaluation criteria. This means that the Activity had an expected formatted output. This was not the same for the rubrics. Their creation prompt was given in the same chat window in which A1 was generated. This was intentional, as the LLM could remember previous instances generated in earlier requests. Prompt 4.2.1 presents an aggregation of both prompts for space convenience, but they were requested separately in the experiment.

Prompt 4.2.1. Generation of A1 and its rubrics

Você é um professor ministrando um curso de **{Mineração de Dados}** e deseja construir uma atividade prática para seus alunos. Note que seus alunos não necessariamente têm uma base forte em matemática e ciência da computação, então o nível de dificuldade da tarefa deve ser **{INTERMEDIÁRIO}**.

Os principais aspectos a serem avaliados na atividade são:
- Leitura e tratamento simples dos dados (20% da nota)
- Visualização de conjuntos dos dados (20% da nota)
- Tratamento e seleção de dados para Classificação com base em Árvores Binárias de Decisão (30% da nota)
- Avaliação das Classificações obtidas com as métricas de Accuracy, Precision, Recall, F1-Score e AUC (30% da nota)

As ferramentas disponibilizadas para os alunos para a atividade são: Python 3, Matplotlib, Pandas, NumPy, Scikit-Learn. O dataset alvo da atividade está em anexo, de nome **{"titanic.csv"}**, você deve utilizá-lo para gerar as atividades. Em anexo também está contido o arquivo **{"example.pdf"}** que tem um modelo de atividade de um assunto distinto. Você deve consultá-lo para se basear na construção da nossa atividade proposta.

Previamente, havíamos gerado a Atividade Mineracao Titanic nesta conversa. Você poderia elaborar uma rúbrica para correção de cada uma das 4 tarefas da atividade?

4.2.2. Activity 2 (A2) - Grouping

The co-creation process of A2 differed slightly from that of A1. First, because we already had a previous experience from the first Activity, we decided to take it into account. Given this, we aimed to assess a scenario in which the instructor did not define the specific sub-

jects of the activity. Activity generation into two steps: one for determining the objectives (which would later become Tasks), and the other for generating the activity based on what was previously co-created. This initial setup consisted of a detailed prompt, along with lecture slides dedicated to Data Grouping. This prompt did not have Task objectives, but maintained weight, difficulty level, and quantity of different criteria to be assessed.

Prompt 4.2.2. Initial generation prompt for A2 and its rubrics

Você é um professor ministrando um curso de **{Mineração de Dados}** e deseja construir uma atividade prática para seus alunos. Porém, antes da atividade em si, você quer estabelecer critérios do que gostaria de avaliar e propor rúbricas para tal avaliação. Note que seus alunos não necessariamente têm uma base forte em matemática e ciência da computação, então o nível de dificuldade da tarefa deve ser **{INTERMEDIÁRIO}**.

O principal objetivo da atividade é o estudo sobre o tema Agrupamento de Dados. Os alunos terão as seguintes ferramentas disponibilizadas para trabalho: Python 3, Matplotlib, Pandas, NumPy, Scikit-Learn.

O material utilizado na aula se encontra em anexo, de nome **{"agrupamento.pdf"}**. Você deve utilizar esse material para elaborar entre 4 e 5 itens-chave para avaliação numa atividade a ser posteriormente construída. Todos esses itens devem ter sua dificuldade incrementada, mas ainda se mantendo em **{INTERMEDIÁRIO}**. Para cada item gerado, proponha um peso parcial da nota, em porcentagem.

Você deve elaborar uma rúbrica no formato de uma matriz com o objetivo de auxiliar o instrutor na correção. A matriz deve ser composta por quatro colunas, representando milestones obtidos com a resposta do aluno para cada tarefa. Cada linha deverá representar uma tarefa **{(T1, T2, T3, T4)}**. Para cada linha, preencha com uma descrição dos objetivos a serem atingidos pela resposta do aluno para alcançar cada milestone, com base na tarefa e suas descrições correspondentes.

With the results obtained from the initial setup, another prompt was requested in the same chat window to generate A2. This final prompt (omitted for space purposes) was similar to the one from A1, with the difference that it instructed the use of the previously defined criteria. As supplementary files, another curated dataset was provided, as well as A1, to serve as a document base.

For A2's rubrics co-creation, since it was observed that not specifying an expected output format in A1 (element further discussed in Section 5) had unwanted results, a fixed output was set: a table grouping each rubric that followed a classification into four categories: Excellent, Good, Fair, and Unsatisfactory. We present another aggregation of the initial and rubric generation prompts in Prompt 4.2.2.

5. Results

In this section, the results obtained from the experimental evaluation are presented. It begins with the Activities and rubrics generated, followed by feedback obtained from the students of the Data Mining course.

5.1. Co-creation of A1 and A2

For this paper, we will present excerpts from A1 and A2 which are important for the results. Full Activities and rubrics are available in an Anonymous GitHub[5]. Regarding Task creation, the first fact noticed was that the LLM model chose to subdivide each Task by itself, but it was not requested to do so. This was an interesting outcome, but, at the same time, it warranted attention since these subtasks could probably deviate from the original Task objective. Based on this, an initial analysis was conducted to assess if these subtasks were coherent with their objective. Table 1 presents the results obtained.

Table 1. Description of the number of subtasks and Task adaptation need.

Activity 1 - Classification		**Activity 2 - Grouping**	
Task	*Subtasks*	*Task*	*Subtasks*
1	4	1	4
2	3	2	3
3	**3**	3	**2**
4	**3**	4	**2**
-	-	5	**2**

Subtasks in **bold** & underline had to be adapted.

Table 1 shows that the initial Tasks for both Activities were perfectly adherent to their objectives, requiring no need of adaptation. However, the final Tasks of A1 had minor mistakes that we thought could create confusion among students. These mistakes were caused by an imprecise or incomplete description of commands. Excerpt 1 exemplifies this by showing what was obtained from the LLM model and adapted by the instructor before assigning to students.

Excerpt 1 - Adaptations made to A1's Tasks 3 and 4.

Task 3: Data treatment and selection (3.0 points)
- Select relevant attributes for data classification (avoiding **characteristics/**columns that are not relevant to prediction)
...

Task 4: Evaluation of Data Classification (3.0 points)
...
- Compute and interpret Accuracy, Precision, Recall, F1-Score, and **ROC/**AUC for each generated model.
...

Analysis of A2 had to be conducted in two steps because of its specific co-creation process. The only problem encountered in the initial prompt was that the LLM model suggested comparing the K-Means algorithm with at least one other, such as DBSCAN or hierarchical grouping. However, the lecture slides did not cover these other algorithms. So, because of this, we considered this suggestion as faulty and further requested the model for adaptation. This time, it suggested comparing K-Means with K-Means++,

[5]https://anonymous.4open.science/r/masgar-sbie-C47C/

which was mentioned in the slides. Moving on to the generation of A2, which was based upon the adapted results from the aforementioned initial prompt, the same mistakes from A1 were repeated. This time, most of the confusion was caused by the LLM model not using a specific word for "clustering": it sometimes used the term in English and, in other times, Portuguese. However, one occurrence warranted attention in this latter approach: the model once again misunderstood the Task related to interact with the K-Means algorithm, requiring students to implement it and not use the one implemented in Scikit-Learn, which was the intention. Excerpt 2 exemplifies this occurrence.

Excerpt 2 - Adaptations made to A2's Task 3.

Task 3: Usage and Execution ~~Implementation~~ of Data Grouping Algorithm (2.5 points):
- **Use** ~~Implement~~ the K-Means algorithm in Scikit-Learn:
    - Use the elbow method and test for different values of K that are coherent with the method's results
    - Set different values of the `init` parameter to test both K-Means and K-Means++
...

As for the rubrics, we briefly mentioned in Section 4 that A1 had not an output format described in the generation prompt. Because of this, the LLM model generated a long checklist as output: each one with a small, simple evaluation criteria that needed to be present. Sometimes, other advice was given such as "if the code does not compile, deduct points accordingly". Since this format was not adherent to how literature often represents rubrics [Alves et al. 2020, Lima et al. 2024], a second prompt specified the table-like format, which was requested in A2. Table 2 presents an excerpt of the resulting rubric for A2. The excerpt is from Task 2: data and cluster visualization.

Table 2. Rubrics for A2's Task 2: data and cluster visualization.

Score	Interval (%)	Requirements
Excellent	[86, 100]	Well-constructed plots, distinct colors, highlighted and interpretable centroids.
Good	[51, 75]	Understandable plots, but without proper centroid detailing.
Fair	[26, 50]	Confusing or incomplete plots.
Unsatisfactory	[0, 25]	No visualization or uninterpretable visualization.

5.2. Students' Feedback

Since the instructor needed to apply an anonymous feedback survey to students at the end of the course, we leveraged the opportunity to enhance this study. The survey was elaborated to collect general course evaluation data alongside specific questions about A1 and A2. We were especially interested in understanding if the Activities were adequate to course format, aiming at cohesion and conciseness, difficulty, and chance of skill development. In total, 15 students answered the anonymous survey.

Regarding A1, 13 students stated that they were able to develop essential skills in Data Mining (analysis, modeling, result interpretation, etc.), while only two remained neutral. Most of these students corroborated by saying they were able to connect theory and practice, and the provided tools and didactic materials were sufficient to solve the

Activity. One thing to notice here is that some students mentioned that providing more details on the dataset would be helpful. Aside from this, 13 students stated that A1 was coherent with the course syllabus, while two remained neutral. About the difficulty, 11 students affirmed it was adequate to what the instructor taught in the classes, while three remained neutral and one said it was not adequate. Lastly, students rated the creativity of A1 with an average of 4.53 out of 5.

As for A2, 11 students said to have developed essential skills, while three remained neutral and one said to have not developed any skill. It became clear that students identified A2 as more complex and difficult than A1, with the need for constant didactic material consultation. 11 students deemed the difficulty level adequate for the course, while three said the opposite, and one remained neutral. This result is corroborated with 11 students stating the Activity was coherent with what was taught, with two saying the opposite, and two remaining neutral. The answers were divided into stating that the complexity of the Activity would be better tailored by permitting large student groups (more than pairs) and with more details on certain topics like outliers. The average creativity level did not vary much from A1, since this time it was rated as 4.33 out of 5.

6. Discussion

The conduction of the preliminary experimental evaluation (cf. Section 4) revealed interesting results. In general, we agree that the feasibility of the methods was made possible because there were only two Activities with four or five Tasks. Overall, the experiment aimed to simulate how instructors and TAs would interact with MASGAR. Although it was not in a multi-agent scenario, the conversations with the LLM model can be seen as different agents interacting with each other (cf. Figure 1). Moreover, both the instructor and the TA stated that they had spent more time searching for adequate datasets than interacting with and adapting the LLM output, so this did not significantly impact their workload. In the future, a specialized agent could also be constructed to search for adequate datasets, both synthetic, for class examples, and real ones, for Activities, to ease the instructor's workload.

These adaptations in LLM output were expected, since literature says that development of more complex educational artifacts often results in hallucinations [Huang et al. 2023] or commands being unclear or imprecise. When considering a subject agnostic scenario, it must be regarded as what Martins *et al.* (2023) stated. These occurrences also depend on LLM's training data, as even ChatGPT apparently was not capable of solving Natural Deduction. Given that we aim for MASGAR to be employed in any educational context, the ability to develop domain-specific agents can be leveraged by having them tailored to a specific subject dataset beforehand.

We agree that this experimental setup is mainly limited to students' feedback on the Activities. A perspective from other Data Mining instructors and TAs would undoubtedly benefit the evaluation of A1 and A2, allowing for more studies on the agreement level with LLM and human raters [Kumar and Boulanger 2021]. Additionally, feedback from students and the results from the inter-human raters could also be fed back to MASGAR to further improve its generating capabilities of Activities and rubrics. Our results have shown that cohesion and creativity levels were surprisingly efficient in both co-creation scenarios. However, it is worth noting that they were executed on a powerful LLM model

(GPT-4 Turbo). Further studies are needed to assess how different models, particularly open-source ones, would respond.

Rubric development also needed some adaptation. While the instructor and TA agreed on the core requirements, their interval score was not appropriate. The model divided the four scores into four equidistant intervals, and, as that did not meet the program standards, they were altered to: Unsatisfactory (0 – 35%), Fair (36 – 60%), Good (61 – 85%), and Excellent (86 – 100%). We argue that this initial LLM output may have resulted from a lack of detailed prompting or a "fair and benevolent" aspect that was employed. Nevertheless, this was a minor adjustment, easily spotted and corrected.

We posit that these obtained results pave the way for the creation of specific LLM-based agents in MASGAR or any other similar architecture that researchers and instructors might develop. Our definition of co-creation aims to clarify how teachers and students can utilize LLM models with a Human-in-the-Loop approach [Wu et al. 2022] to incorporate generative AI into their educational contexts, regardless of subject or school level.

7. Conclusion

While literature on using Generative Artificial Intelligence for automating assessment generation and grading is steadily growing, research on rubric generation and application is still scarce. This study proposed MASGAR, an LLM-based Multiagent System for Generating Activities and their Rubrics. The system assists instructors and teaching assistants in co-creating summative assessments with LLMs. MASGAR's architecture was defined specifying its core LLM-based agents. Our solution was assessed in an experimental setup where an LLM model was requested by a user via a chatbot, simulating an end-user for MASGAR for the co-creation of two Activities and their rubrics in a Data Mining course. Results revealed that while the LLM model performed well in generating these educational artifacts, minor manual improvements were needed before assigning them to students. The study further revealed that LLMs may require additional training for rubric development, as they did not consistently produce a fixed output format for rubric creation. Future studies will include the comprehensive implementation of MASGAR and its in-depth evaluation in various educational contexts, while also utilizing different LLMs within the multi-agent system. We plan to extend the architecture to provide automated grading and feedback generation to students, primarily based on rubrics co-created between the LLM and instructors. Another objective is to expand MASGAR to undergraduates from different courses, since this study was limited to post-graduate students with a different studying scale.

Ethical Considerations

This study was partially conducted in a course from an university in São Paulo, Brazil. However, the development of each educational assessment through LLMs was chosen at the instructor's behest to enhance teaching and learning. Additionally, since each instructor is requested to ask for students' feedback at the end of the course, the study leveraged this opportunity to collect opinions on the Activities. However, the survey was anonymous and optional for students. Since all methods were within the instructor's expected responsibilities to minister the course, ethical approval for a Research Ethics Committee was not requested.

Acknowledgments

We thank the National Council of Technological and Scientific Development (CNPq), Brazil, grant #301337/2025-0.

References

Aguilar-Savén, R. S. (2004). Business process modelling: Review and framework. *International Journal of production economics*, 90(2):129–149.

Alves, N. d. C., von Wangenheim, C. G., Alberto, M., and Martins-Pacheco, L. H. (2020). Uma Proposta de Avaliação da Originalidade do Produto no Ensino de Algoritmos e Programação na Educação Básica. In *Simpósio Brasileiro de Informática na Educação (SBIE)*, pages 41–50. SBC.

Bahroun, Z., Anane, C., Ahmed, V., and Zacca, A. (2023). Transforming Education: A Comprehensive Review of Generative Artificial Intelligence in Educational Settings through Bibliometric and Content Analysis. *Sustainability*, 15(17):12983.

Becker, J. (2024). Multi-agent large language models for conversational task-solving. *arXiv preprint arXiv:2410.22932*.

Bloom, B., Hastings, J., and Madaus, G. (1971). *Handbook on Formative and Summative Evaluation of Student Learning*. McGraw-Hill.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Carbonell, J. (1970). AI in CAI: An Artificial-Intelligence Approach to Computer-Assisted Instruction. *IEEE Transactions on Man Machine Systems*, 11(4):190–202.

Chico, V. J. S., Tessler, J. F., Bonacin, R., and dos Reis, J. C. (2024). BEQuizzer: AI-Based Quiz Automatic Generation in the Portuguese Language. In Rapp, A., Di Caro, L., Meziane, F., and Sugumaran, V., editors, *Natural Language Processing and Information Systems*, pages 237–248, Cham. Springer Nature Switzerland.

Duong, T. N. B. and Meng, C. Y. (2024). Automatic grading of short answers using large language models in software engineering courses. In *2024 IEEE Global Engineering Education Conference (EDUCON)*, pages 1–10.

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., and Liu, T. (2023). A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions.

Izu, C. and Mirolo, C. (2024). Towards comprehensive assessment of code quality at cs1-level: Tools, rubrics and refactoring rules. In *2024 IEEE Global Engineering Education Conference (EDUCON)*, pages 1–10.

Jiang, B., Xie, Y., Wang, X., Yuan, Y., Hao, Z., Bai, X., Su, W. J., Taylor, C. J., and Mallick, T. (2024). Towards rationality in language and multimodal agents: A survey. *arXiv preprint arXiv:2406.00252*.

Jo, E., Epstein, D. A., Jung, H., and Kim, Y.-H. (2023). Understanding the Benefits and Challenges of Deploying Conversational AI Leveraging Large Language Models for

Public Health Intervention. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–16, Hamburg Germany. ACM.

Keuning, H., Heeren, B., and Jeuring, J. (2021). A tutoring system to learn code refactoring. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, pages 562–568.

Kinnunen, P. and Simon, B. (2012). My program is ok – am i? computing freshmen's experiences of doing programming assignments. *Computer Science Education*, 22(1):1–28.

Krathwohl, D. R. (2002). A Revision of Bloom's Taxonomy: An Overview. *Theory Into Practice*, 41(4):212–218.

Kumar, V. S. and Boulanger, D. (2021). Automated Essay Scoring and the Deep Learning Black Box: How Are Rubric Scores Determined? *International Journal of Artificial Intelligence in Education*, 31(3):538–584.

Lancaster, T., Robins, A. V., and Fincher, S. A. (2019). *Assessment and Plagiarism*, page 414–444. Cambridge Handbooks in Psychology. Cambridge University Press.

Lima, M. R., Ferreira, D. J., and Dias, E. S. (2024). Uso de Rubricas em Disciplinas de Programação Introdutória: Uma Revisão Sistemática da Literatura. In *Simpósio Brasileiro de Informática na Educação (SBIE)*, pages 1–14. SBC.

Linnenbrink, E. A. and Pintrich, P. R. (2003). The role of self-efficacy beliefs instudent engagement and learning in the classroom. *Reading & Writing Quarterly*, 19(2):119–137.

Martins, F. L. B., de Oliveira, A. C. A., de Vasconcelos, D. R., and de Menezes, M. V. (2023). Avaliando a habilidade do ChatGPT de realizar provas de Dedução Natural em Lógica Proposicional. In *Simpósio Brasileiro de Informática na Educação (SBIE)*, pages 1282–1292. SBC.

Phung, T., Pădurean, V.-A., Cambronero, J., Gulwani, S., Kohn, T., Majumdar, R., Singla, A., and Soares, G. (2023). Generative ai for programming education: Benchmarking chatgpt, gpt-4, and human tutors. In *Proceedings of the 2023 ACM Conference on International Computing Education Research-Volume 2*, pages 41–42.

Rockembach, G. R. and Thom, L. H. (2024). Investigating the Use of Intelligent Tutors Based on Large Language Models: Automated generation of Business Process Management questions using the Revised Bloom's Taxonomy. In *Simpósio Brasileiro de Informática Na Educação (SBIE)*, pages 1587–1601. SBC.

Russell, S. J. and Norvig, P. (2016). *Artificial Intelligence: A Modern Approach*. Prentice Hall Series in Artificial Intelligence. Pearson, Boston Columbus Indianapolis, third edition, global edition edition.

Scriven, M. (1967). The methodology of evaluation. In Tyler, R., Gagné, R., and Scriven, M., editors, *Perspectives of Curriculum Evaluation, AERA Monograph Series on Curriculum Evaluation*, volume 1, pages 39–83. Rand McNally, Chicago.

Villa, J. E. A., Garcia, R., Miranda, A. L. M., Oran, A., Guedes, G. T. A., Santana, B. S., Silva, D. G., Valle, P., and Silva, W. (2024). Perspectiva dos Estudantes sobre um Agente Pedagógico Baseado em Exemplos para a Aprendizagem de Programação:

uma análise qualitativa. In *Simpósio Brasileiro de Informática na Educação (SBIE)*, pages 459–473. SBC.

Wang, L., Ma, C., Feng, X., Zhang, Z., Yang, H., Zhang, J., Chen, Z., Tang, J., Chen, X., Lin, Y., Zhao, W. X., Wei, Z., and Wen, J. (2024). A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6):186345.

Wang, Z., Cai, S., Chen, G., Liu, A., Ma, X., and Liang, Y. (2023). Describe, explain, plan and select: Interactive planning with large language models enables open-world multi-task agents. *arXiv preprint arXiv:2302.01560*.

Wolber, D., Abelson, H., Spertus, E., and Looney, L. (2011). *App inventor*. " O'Reilly Media, Inc.".

Wu, X., Xiao, L., Sun, Y., Zhang, J., Ma, T., and He, L. (2022). A survey of human-in-the-loop for machine learning. *Future Generation Computer Systems*, 135:364–381.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., et al. (2023). A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2).