International Journal *of* Designs for Learning
2025 | Volume 16, Issue 2 | Pages 170-181

SCISTORY: DESIGNING AI-SUPPORTED INQUIRY IN SCIENCE LEARNING GAMES

Megan Humburg[1], Joshua Danish[2], Yeojin Kim[3], Cindy E. Hmelo-Silver[1], Dalila Dragnić-Cindrić[4], Zach Ryan[1], Krista Glazewski[3], & James Lester[3]
[1]*Indiana University;* [2]*University of Illinois Urbana-Champaign;* [3]*North Carolina State University;* [4]*Digital Promise*

This design case explores how an AI-supported, narrative-centered science learning game (SciStory: Pollinators) was designed over multiple iterations to support middle schoolers' socioscientific learning, engagement, and persuasive writing. The case highlights how AI-driven conversational agents were designed to support student-led socioscientific inquiry, and the tensions our team explored as we integrated agents into a story game about community food systems, pollinators, and neighborhood land use.

Megan Humburg is a Postdoctoral Fellow at Indiana University. Her research explores interdisciplinary science learning, engagement, and playful inquiry.

Joshua Danish is a Professor at the University of Illinois. His research focuses on the role of representations in supporting thinking and learning.

Yeojin Kim is a Research Scientist at North Carolina State University. Her research focuses on AI, machine learning, and natural language processing.

Cindy Hmelo-Silver is a Distinguished Professor of Learning Sciences at Indiana University. Her research focuses on how people engage in technology-supported collaborative inquiry in STEM learning.

Dalila Dragnić-Cindrić is a Senior Research Scientist at Digital Promise. Her research focuses on collaboration and regulation of learning in STEM.

Zach Ryan is a PhD graduate from Indiana University. His research focuses on playful learning in natural environments to build relationality to place.

Krista Glazewski is Executive Director of the Friday Institute and Associate Dean for Translational Research. Her research focuses on scaffolding in complex problem-solving environments.

James Lester is the Goodnight Distinguished University Professor in Artificial Intelligence and Machine Learning at North Carolina State University. His research focuses on AI and education.

INTRODUCTION

This paper traces the development of a narrative-centered learning game, SciStory: Pollinators, which our team designed to support socioscientific inquiry and persuasive writing skills for middle school learners. The SciStory design emerged from a research project about narrative-centered learning environments that use AI tools to engage middle school learners (Humburg et al., 2025; 2024). Students engage with SciStory by working through a "choose-your-own-adventure"-style story about a community deciding how to use empty land. Students talk with various game characters who are members of a community, take notes in digital notebooks, and work together to build a persuasive argument to convince the Mayor of the town whether to use the land for a community garden or a parking garage. While AI-infused digital games have been gaining popularity for some time (McLaren & Nguyen, 2023), advances in generative AI technologies (GenAI) have shifted what AI integrations are possible in learning games. Pre-GenAI pedagogical agents could offer targeted support, narrative adjustments, and updated character behaviors based on student gameplay, but they were limited in the scope and detail of how agents could respond to students' ideas. GenAI agents offer the possibility to support more varied, open-ended, and student-centered interactions. Our design focuses on integrating AI-driven conversational agents (which we will

A publication of the Association for Educational Communications & Technology (AECT), published by Indiana University Libraries Journals.

https://doi.org/10.14434/ijdl.v16i2.42049

refer to as "agents" moving forward) into game characters that help players engage in meaningful, student-led socioscientific inquiry into a complex problem.

Our design draws inspiration from work on intelligent agents designed to fill pedagogical roles such as supportive tutor, collaborative inquiry facilitator, teachable peer, and knowledge building partner. Guided by previously designed roles for such agents, we designed three types of agent interactions for SciStory: 1) agents that answer content questions (e.g., "How do bees help pollinate flowers?"), 2) agents that evaluate students' argumentation and offer constructive feedback (e.g., "Your argument is a little vague - try adding some specific examples from your notes."), and 3) agents that offer support when students encounter confusion or frustration (e.g., "My group isn't listening to me.").

We were particularly interested in the second and third types of agents, where our design goal was to avoid giving learners the answers directly and instead provide pedagogically-oriented feedback that would encourage students to further explore the interactive story and the science behind its underlying dilemma. From an education perspective, we believed that these agents would be more beneficial for learning than the content question agents, since they would support student agency in the inquiry process. From a computer science perspective, these agents also appeared more complex to design, given that the agent's behavior must be more nuanced in how it handles different types of inquiries. The goal for this design was that through playing the story game, students would develop a deeper understanding of how to integrate both community needs and scientific evidence into persuasive argumentation to help solve locally meaningful community problems.

DESIGN CONTEXT

The target age group for this design was 5th-8th grade students in the Midwestern United States. The design team had previously created narrative-centered learning games in which players had to solve a particular science problem (e.g., "Why are the fish in our ponds getting sick?"). In discussions about how to increase player agency and engagement with scientific inquiry, we decided to design SciStory around an open-ended dilemma that did not have a single right answer to be discovered. Our hope was that students would be engaged more deeply with a scientific problem that was both relevant to their local community and offered them opportunities to build their own solutions to a complex problem. We selected a socioscientific issue for the game's dilemma to provide this local relevance, since socioscientific issues require learners to explore both the scientific evidence behind a topic as well as the real-world impact of different choices on their community (Sadler, 2004). We chose to integrate learning goals around persuasive writing and argumentation with scientific inquiry to provide students with an interdisciplinary learning experience that could help them practice the skills integration they need to tackle such complex dilemmas outside of school.

The team had done previous work designing learning activities to teach elementary school students about bees and pollination, which led us to the story about building a community garden in order to leverage this pool of existing expertise. "How should we use an empty plot of land in our community?" was chosen as the broader socioscientific issue because it was widely applicable to communities across the U.S., and students could use evidence to generate a variety of viable solutions. Our hope was that after designing a base story, the game's narrative could be easily adapted to different local contexts, so that learners could explore layers of food systems and land use that were most relevant for their particular community.

Before the creation of SciStory: Pollinators, our team had conducted multiple rounds of focus groups with teachers and students to gather their perspectives on the use of AI and games for learning. Teachers were asked about genre, storytelling, and how their students might react to using games for learning. Students were asked about their previous gaming experiences, genres of stories they found compelling, and key features they would want to see in a learning game. Both groups were also asked about their perspectives on AI and how they would want AI to support learning within an educational game.

Key feedback from both groups influenced the SciStory design, including teachers expressing the desire for their students to learn how to ask for help and seek out information that they don't already know but need to learn more about, which is a key goal of problem-based learning environments. This feedback led us to design helper agents as one of our three AI agent types, to support students in developing help-seeking strategies when struggling during open-ended inquiry. Students in our focus groups also expressed a desire for an animal companion during gameplay, which led us to create Tulip the honeybee as the character design for the helper agent. Other key themes from these focus groups that guided our design were the need for high student agency, the desire for more engaging learning activities, concerns about data privacy, and the realistic limitations of classroom IT resources and infrastructure (Humburg et al., 2024). We will explore these themes further in our discussion of decision-making in later sections.

SCISTORY DESIGN

Team Composition & Rationale

The design team is an interdisciplinary, cross-institutional group that includes five learning scientists, one instructional systems technology (IST) researcher, and two computer scientists specializing in AI. The team spans three research

universities and one research nonprofit. Author 1 is a learning scientist, and she was the lead designer of the SciStory narrative and the developer of the 2D visual novel iteration of the game. Author 3 is a computer scientist specializing in natural language processing and machine learning, and she led the technical development of the AI agents for SciStory. Authors 2 and 4 are learning sciences faculty, Author 7 is IST faculty, and Author 8 is a faculty member in computer science specializing in AI in education. These four authors contributed substantially to the conceptualization of the design goals and offered ongoing guidance in terms of improving the SciStory design to support student learning. Authors 5 and 6 are also learning scientists, and they contributed to design discussions as well as data collection and analysis for the design implementations. Authors 1, 2, and 3 met regularly to discuss the design, development, integration, and testing of the AI agents and collaboratively refined agent prompts to improve performance. Additional team members were involved in the design creation, including software engineer Kara Cassell, who built the underlying software and game interactions for the initial 3D version of SciStory, but these team members were not substantially involved in design discussions.

Our design team had a shared goal of creating an educational game that was both effective for student learning and utilized innovative approaches to AI design and integration. The various needs of different stakeholders (including team members from differing fields, school districts with limited resources, and teachers with specific pedagogical goals) drove many of our design decisions. For example, our key school partners include rural districts in Indiana and North Carolina, who often deal with resource issues such as limited internet bandwidth and a lack of cell service, as well as student laptops and tablets with limited processing power. These resources are in stark contrast to the design conditions in which our university team members are designing and testing our tools, who have access to testbed classrooms with high-speed internet and high-budget laptops. We often needed to reorient our design choices by attending to what would be feasible in a technology-scarce environment. We also adjusted the narrative and learning goals as the design evolved to incorporate the needs of our teacher partners. Though scientific inquiry was our initial disciplinary context for the design, our teachers in Indiana were very interested in using SciStory to help students practice their persuasive writing skills, so we ended up integrating key vocabulary and skills from the teachers' persuasive writing curriculum with our existing focus on scientific argumentation.

Given the design team's larger goal of understanding the potential impact of AI on learning, much of SciStory's initial design focused on encouraging learners to interact as often as possible with AI-driven characters. In the following sections, we outline the learner experience of the SciStory game and the key design decisions that our team made to encourage student engagement and learning.

Learner Experience

SciStory: Pollinators was designed as a collaborative, playful learning experience for small groups of middle school students. We chose a collaborative game design because it pairs well with complex, open-ended problem-solving, allowing students to discuss scientific evidence together, negotiate ideas, and engage in metacognitive reflection around their team's progress towards solving the problem. Our team's previous work designing collaborative science games informed this choice, because we have seen the value of engaging students in playful collaborative inquiry for science learning, student agency, and student engagement (Saleh et al., 2019). The game is played on a laptop in a browser window, and students can either share a single laptop and take turns controlling the story, or they can each play on their own laptop but share thoughts and ideas aloud with their groupmates as the story progresses. In implementations so far, we have had students work in dyads around a single computer, in groups that they choose, in order to promote social and emotional engagement.

The game is designed in the style of an interactive visual novel (see Figure 1), where students interact with 2D characters by engaging in dialogue and then making choices about what to ask or where to go next, each of which influences how the story unfolds. In order to maximize student agency, collaboration, and engagement, we designed the SciStory narrative to have a variety of branching dialogue options, so that each group, depending on the choices they make, could uncover different pieces of evidence to share with

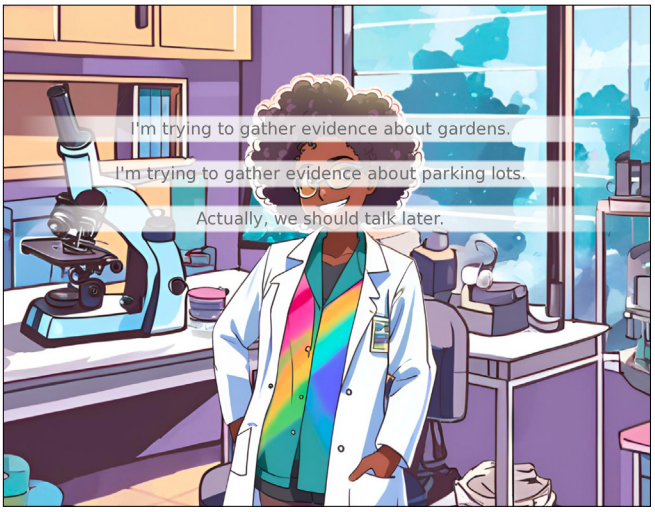

FIGURE 1. Players can click on different characters, like Amara the food scientist (above), to trigger conversations. The player selects from possible dialogue choices to move the story forward, which influences what information they learn and gather.

FIGURE 2. The prototype 3D version of SciStory, which followed the same general narrative as the 2D version, but allowed players to explore each location physically and investigate books and objects like beehives to gather additional information.

their classmates. Author 1 built the game using the Ren'Py Visual Novel Engine and led the writing of the narrative and character interactions. Backgrounds and character images were created using Canva's AI Image Generator, and Author 1 refined these images after generation. Art for Tulip the bee was drawn by our team's visual artist, Courtney Barron. Other visual assets, such as buttons, were created using Canva's asset library.

Initially, the prototype version of SciStory was built as a 3D game in the Unity game engine (see Figure 2), so that players could walk around in a virtual world as they met and spoke with different characters. However, in pilot tests with students, we found that learners became frustrated by the limited functionality of the 3D game environment, as they expected it to involve traditional video game mechanics like running, jumping, fighting, collecting items and/or exploring. We ran several student focus groups with middle school students at our university, as well as an implementation in a 5th grade classroom across two class sessions. Students initially engaged with the story, but they tended to become distracted and would spend large amounts of time running around in the game or attempting to move their character outside the boundaries of the game locations to see if they could "break" it. While the 3D game did include some books and other objects to explore to gather more information, the majority of gameplay was limited to character dialogue. Given that the core game mechanics for SciStory are about having conversations and gathering information, we shifted to a 2D visual novel format so that players were less focused on movement-based interactions and more focused on talking to characters, synthesizing evidence, and making decisions. This shift from 3D to 2D also allowed us to better support our current and future school partners, since a 2D visual novel is much easier to run on older or lower-powered laptops that are typically found in under-resourced schools.

In contrast to the 3D game environment, where students can roam freely in a game location, a visual novel does not involve any physical movement of the player's own character. Instead, game progression is more like a storybook, where players select dialogue options, and the background scene and characters on screen shift as new conversations are started and the story progresses. Students can choose to visit different locations in the visual novel by navigating an in-game menu, but they do not see their own character or control camera movement.

The main downside of this switch from 3D to 2D was that our team spent significant time developing collaborative features for the 3D environment that could not be quickly imported into the visual novel engine. These features were developed for some of our team's previous collaborative games, and while they were useful features, we decided not to prioritize their development for the 2D version of SciStory given our software development resource limitations. In the 3D game environment, players on different laptops could play as a group and talk with each other using an in-game text chat. In practice, we saw the chat mostly used for off-topic chatter between students rather than game-focused discussions, so we did not prioritize adding chat functionality to the 2D SciStory design. The 3D environment also had a collaborative brainstorming board feature, so that students could share notes they had taken with their peers and organize the notes into columns to sort which notes best supported the group's argument. We were not able to add a collaborative brainstorming board to the 2D visual novel for the current iteration of SciStory, given that this would have required more staff software development resources than our team had available at the time. However, our team has plans to develop a similar note-organizing feature to a future iteration of the design to support students in collaboratively reflecting on what notes they have taken and how they can be synthesized into a persuasive argument.

FIGURE 3. Tulip the honeybee (left) introduces herself as the player's guide and explains how they can use the bee button (right) to call her whenever they need.

Students begin SciStory by meeting Tulip, a honeybee character that serves as their guide through the narrative and a helper agent if they get stuck (see Figure 3). Tulip guides the player through an initial introduction, and then students are able to call Tulip for help at any point afterward by clicking a button with a bee icon. The "bee button" is always available for students in the upper right-hand corner of the screen, along with a button to initiate travel to a new location and a button that opens the player's notebook. Tulip's conversation can even be initiated in the middle of a different character's conversation, and Tulip will appear as an overlay and then allow players to pick up where they left off when they dismiss her. When calling Tulip, students can select from pre-scripted requests for help, such as "I don't know what to do" or "I'd like some help persuading the mayor", or they can choose to type their own question.

Through their initial conversation with Tulip, players choose a community context for their gameplay (rural, suburban, or city), which alters the background images they see as they explore the community. This choice also changes whether characters refer to the proposed parking structure as a parking lot or a parking garage, to reflect differences in city planning norms for rural and suburban vs. urban areas.

After the introduction, players are then dropped into the first game location (the Empty Lot), where they meet a local teen named Elliot who introduces them to the game's core

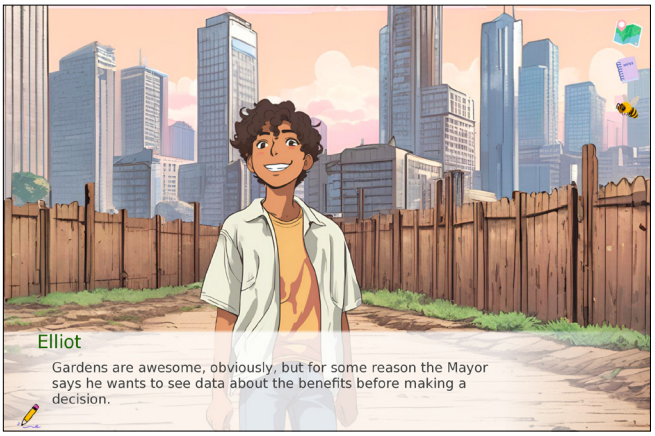

FIGURE 4. Students meet Elliot, a local teen who introduces them to their core task: gather persuasive evidence to convince the Mayor to build either a community garden or a parking structure. Buttons in the top-right corner of the screen allow students to travel to other game locations, access their notebook, and call Tulip for help.

problem: the community has an empty plot of land, and the Mayor is trying to decide what to build on it to best serve the needs of the community (see Figure 4). Two initial options are offered to the learner: 1) side with the community gardeners, who want to build a garden for the neighborhood to grow their own food, or 2) side with CityPark, a corporation who wants to build a parking structure to attract new businesses to the area. Players may offer alternative

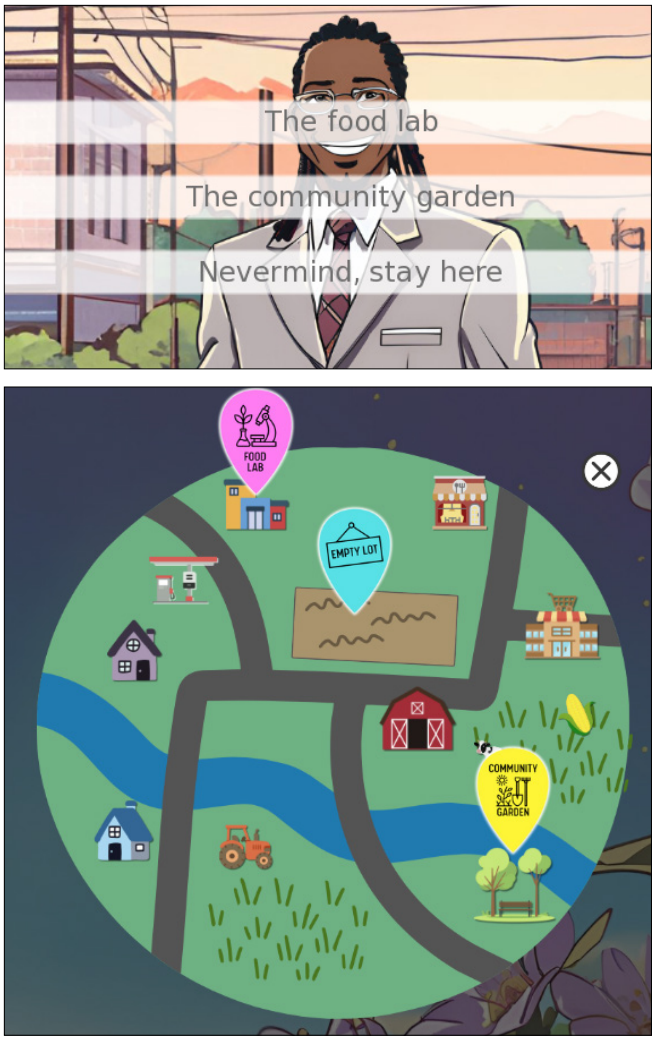

FIGURE 5. Students can click the map button at any time to travel locations, and they will be presented with the choice to select a different location or stay where they are. The top figure shows the initial menu-based travel options, while the bottom figure shows the prototype map-based travel.

suggestions for how to use the empty lot (e.g., build a playground, convince a grocery store to build there), but for the pilot version of the game, the narrative was built around the parking structure vs. community garden dilemma. This decision was made to scaffold learners' exploration of the evidence by simplifying the decision space and to support the design and testing of a targeted set of agents organized around those key ideas.

After receiving their core task, learners are able to decide where they go in the game, who they talk to, and what notes to take in their in-game notebook. They can stay at the empty lot to ask the Mayor about what he needs to be convinced, and they can chat with Cyrus, the CityPark representative, to get his perspective on what a parking lot could do for local businesses. They can travel to a community garden across town, where they can talk with a beekeeper, an expert

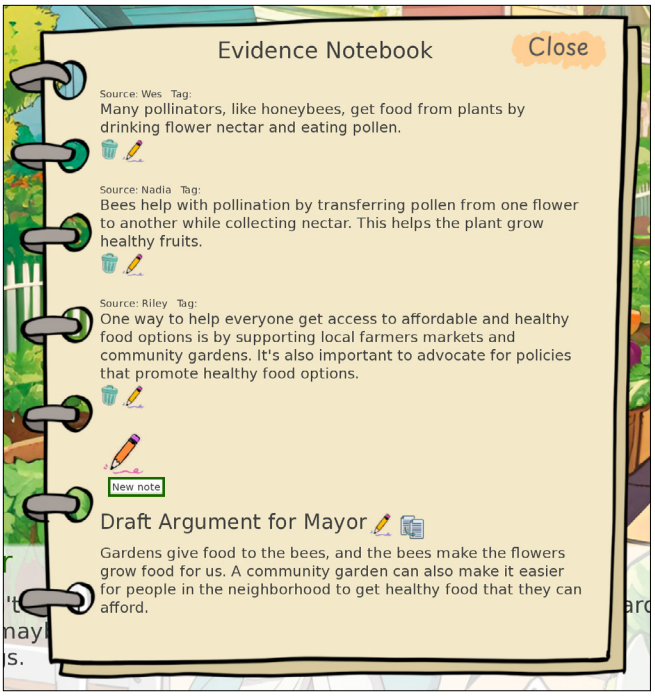

FIGURE 6. Students can take pre-filled notes by clicking a pencil icon next to the characters' dialogue as they speak, or they can write and edit their own custom notes.

gardener, and several community members who share their opinions about having a garden to grow their own food. They can also travel to the food lab, where they can talk with a food scientist about the pros and cons of shipping food versus growing it locally. The food lab also has a community activist named Riley, who can offer advice about how to write a convincing argument to persuade the Mayor. The initial version of the game used a list of menu choices for traveling locations, but we are also testing out an interactive map version of travel, since some students were confused by the travel menu in our pilot tests (see Figure 5).

As the learners talk to these different characters, one of their key tasks is to gather notes that can help them to draft their argument. Anything said by a character can be automatically turned into a note in their in-game notebook with the click of a pencil button, making it more likely they will take notes without feeling their flow of activity is disrupted. They can also edit the pre-made notes or create custom notes to summarize ideas they are hearing about. Each note is labeled with the source of the note (i.e., the speaker), and there is a field for students to optionally add tags to label their notes. Their notebook also contains a section where they can draft and iteratively revise their argument for the Mayor to prepare it for sharing with the agents for feedback (see Figure 6). A core collaborative task for the paired students is to negotiate together what plan for the empty lot they want to support and then discuss what notes they want to highlight as core evidence for their persuasive argument.

AGENT TYPE	DESCRIPTION & CHARACTERS
#1: Knowledge Agents	Provide answers to content questions
• Pollination & Plants	Nadia, Wes
• Food Systems	Riley
#2: Feedback Agents	Evaluate students' persuasive writing
• Advice for draft arguments	Riley, Elliot, Tulip
• Evaluation of final argument	Mayor Watson
#3: Helper Agents	Offer advice in response to confusion
• Gameplay Issues • Collaboration Issues	Tulip

TABLE 1. The agent types and sub-types designed for SciStory, with characters that students could interact with to access the agent types Table 1. The agent types and sub-types designed for SciStory, with characters that students could interact with to access the agent types

The driving force for this story design was to create many open-ended moments for students to engage with AI-driven characters in different conversational contexts to support students' multifaceted engagement with scientific inquiry. We designed three key types of agents that are outlined in Table 1.

When talking with different characters that utilized AI to power their responses, learners did not see them as "Knowledge Agent" or "Feedback Agent." Rather, they would simply engage with a story character through pre-scripted dialogue, and at certain points in the conversation, the character would prompt the player to ask their own questions, which were then processed by the AI agent.

For example, when talking with Riley at the food lab, the player would learn about Riley's job as a community activist, and she would share some pre-scripted ideas about the community garden and its benefits for the neighborhood. Depending on what dialogue options the learner chooses next, Riley could offer to answer questions about food justice and food systems, or she could offer to give the player advice on their draft argument for the Mayor. If the player chooses "I have some ideas for the Mayor", Riley will then ask them to share their evidence in an open text box (see Figure 7).

The contents of what students type are then sent to the corresponding agent. The AI-driven conversational agents consist of modules that manage the game context, pedagogical policy, and a knowledge base. When students type input

FIGURE 7. When interacting with an agent, learners see an open-ended prompt followed by a text area where they can type a question or idea (shown in green) that they want the character to answer or respond to.

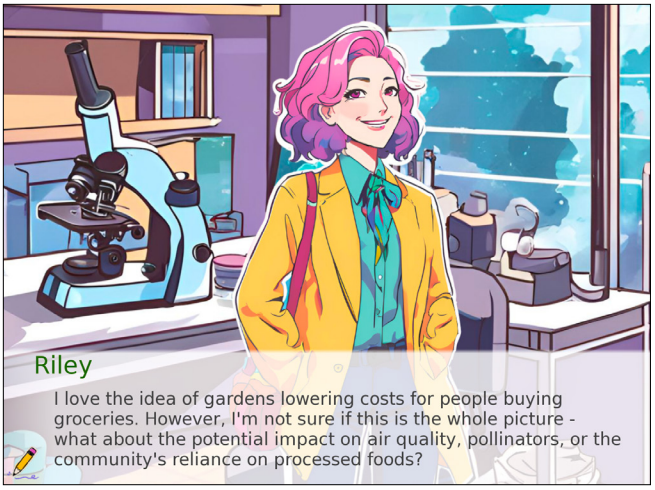

FIGURE 8. Riley provides feedback that encourages students to consider more aspects of the dilemma that they have not addressed in their argument, while acknowledging what ideas they have touched on.

to an agent, the contents of their message are sent along with relevant game data—such as which characters have been spoken to, notes taken, and the number of arguments submitted—so that the agent can retrieve appropriate policies and knowledge and generate a suitable response. The student would then receive this response from the feedback agent (through Riley's character dialogue) that would commend strong features of their argument and suggest ways to further improve its persuasiveness (see Figure 8). The other agents worked in similar ways, with learners able to move seamlessly between pre-scripted pieces of dialogue and cycles of sharing their own questions and ideas. Student pairs in our pilot tests would often take turns speaking to the agents, with one student typing a question or idea while

the other offered suggestions. Though the intended design encouraged students to draft their persuasive writing in their notebooks before sharing with the agents, in practice many groups drafted their arguments out loud, iterating on ideas verbally as one student typed out (and often iteratively re-typed) an argument to share with Riley.

Given the open-ended nature of the game, learners do not have to reach the "end" of the game at any particular time, but when they feel they have written a strong, persuasive argument, they can present it to the Mayor to get an evaluation. The Mayor (using a version of the feedback agent) assesses whether or not the player has convinced him to support their idea, and if he is not convinced, he shares advice about weaknesses in the learner's argument. Based on whether or not the Mayor is convinced to build a garden or a parking structure, the story's narrative toggles between one of two endings (building a garden or building a parking structure).

If the Mayor is persuaded, the player can choose to immediately end the game. They can also choose to continue exploring if they would like to speak to other characters or refine their argument further. If the Mayor is not persuaded, the option to end the game does not appear until the player has spoken to all characters in the game, has taken at least six notes in their notebook, and has tried to persuade the Mayor three times. In the meantime, the player is offered the opportunity to self-evaluate (e.g., "How do you think it went?") and Tulip shares advice for how to better convince the Mayor depending on where the player is struggling. For example, if the player has spoken to everyone but only taken one or two notes, Tulip will encourage them to synthesize their ideas by taking more notes before sharing with the Mayor again.

DESIGNING AI AGENT INTERACTIONS

Training Agents To Guide, Not Tell

The agents embedded in SciStory's characters went through multiple rounds of testing, refinement, and focus grouping before they were implemented in a classroom setting. Our focus on inquiry meant that characters needed to produce accurate information that matched the target content (e.g., facts about bees relevant to a local garden, not everything that might be present in a college level textbook), and they were also required to suggest pedagogically relevant next steps for inquiry. This behavior was intended to mirror the facilitation techniques of a trained teacher, recognizing that a teacher cannot simultaneously work with all students at once as they play the game.

In our initial design conversations with the interdisciplinary team, we encountered differences in how the education team was conceptualizing the role of agents and how our AI developers were imagining the character roles. In some of our team's previous work, the role of agents had been that of a content expert who could answer science questions and provide detailed information for learners. For these designs, the key approach to evaluating AI agent effectiveness was looking at the accuracy and detail of responses. This was the design that guided our knowledge agents, which was fairly straightforward. Agents were designed around a set of scientific knowledge, and they answered questions bounded by that knowledge and closely related topics. These knowledge bases were developed by the design team in collaboration with colleagues who had expertise in food systems, food justice, pollination, honeybees, and related ecosystems topics. Knowledge was selected to cover a general basis of science content appropriate for middle grades students, and the knowledge base was expanded iteratively as students tested the game and asked their own science questions about these topics.

Several education team members were interested in designing agents that, rather than giving students answers directly, would be able to guide students towards discovering their own answers, much like an expert teacher can. This required a very different kind of evaluation of agent responses, because we wanted agents to offer students suggestions and encouraging questions without giving away answers or overwhelming them with too much guidance. To refine our prompts before testing with students, we engaged in weekly testing sessions with the design team and graduate student volunteer play testers to ask agents a variety of questions and assess responses. Volunteers played through the prototype game and engaged with each agent type in the context of the narrative. We would record all inputs and responses and assess each in terms of factual accuracy, logical alignment with what was asked, and alignment with pedagogical design goals. Prompts were then refined and tested again to look for improvements or declines in response quality. Over these iterations of design and testing, we found that it was difficult to prompt the agents to *not* give students information when asked a direct question, but rather to offer them guidance on how to find the information they want. As a part of this design process, the education team created input-response pairs for the CS team to use as guiding examples of how a teacher would respond to students' questions and the response logic behind different kinds of scaffolds.

For example, an early response from the feedback agent sounded like: "Great start! You've mentioned an important benefit of community gardens. To make your argument even stronger, you could add how this helps people who don't have cars and can't easily travel to grocery stores." This response commends what students have already done and directly offers them more evidence to add to their argument. However, we wanted the agents to encourage students to continue exploring the story in order to gather more

evidence, rather than directly offering them ideas to add to their writing.

After adjusting the agent guidance, an updated response would be: "That's a good start! You mentioned that parking lots cause pollution and gardens are good for our air. To make your argument stronger, you could add more specific evidence. For example, you could talk to Nadia about how a community garden would help bees and other pollinators, or to Amara about the health benefits of locally grown food. Keep going!" In this updated response, the agent highlights specifically what was strong about the existing argument. Additionally, rather than directly offering the student evidence, the agent suggests two possible characters they could speak with next, hinting at what kinds of knowledge those characters have but not directly sharing the key ideas. This response gives students an idea of what to do next without directly filling in the details they need to add to make their persuasive writing stronger. This highlights our general design goal for the feedback agent guidance, which was to scaffold student inquiry but not to offload any of the important cognitive work to the AI.

Agents are also adaptive in their responses based on student gameplay. Along with the student's input, the agent receives information about who the student has spoken to, which locations they have visited, how many notes they have taken, and how many arguments they have already shared with the feedback agents. While students early in their investigation might be prompted to go speak to more characters to gather additional information, students who have already spoken to all the NPCs would instead be directed to take more notes in their notebook or to work on synthesizing their notes with their peers to strengthen their argument.

Honoring Student Ideas

Another aspect of learner-AI interactions that our team had to grapple with was training the agent to notice and value students' relevant ideas, even if they didn't necessarily align with traditional scientific evidence. The initial versions of our agents were often too restrictive, treating individual and local knowledge as off-topic, where a teacher would recognize and honor such contributions. The agents tended to shut down students' ideas if they did not align with the agent's knowledge base. For example, a student might say "My grandpa has a farm, and he says it's good to get your hands dirty and grow your own food." However, if the agent's knowledge base is focused on evidence about the impact of pollinators, it might classify the student's idea as off-topic and tell them to focus on gathering scientific evidence about the benefits of gardens for the ecosystem.

Our team had to iterate on the prompts given to the agents in order to note that agents should consider students' ideas from their home lives and cultures as relevant to the conversation if they were related to the topics of gardens, even if they weren't necessarily scientific or fully accurate. After validating the knowledge shared by the student, the agent could then encourage them to gather information from game characters to find new ideas. In this way, our design conversations and revisions centered on balancing content ideas from our knowledge base with unanticipated yet valuable contributions by learners. Again, this is a skill that teachers are well-practiced in, but a type of knowledge that we found rather difficult to train an agent to reliably notice and respond to appropriately when you can't always predict what kinds of ideas students will share.

Ethical Tensions: Hallucinations, Sustainability, & Data Privacy

In designing AI-driven conversational agents, we had to consider the needs of learners (for engagement, for learning) and the expectations of teachers (for pedagogical appropriateness, for student safety), which led to a variety of ethical tensions to explore. One of the initial tensions was the size of the various large language models (LLMs) at our disposal and the pedagogical challenges and ethical conundrums associated with each.

First, we had to consider LLM hallucinations (Huang et al., 2025) and the potential risks of having an educational agent share inaccurate or inappropriate responses with students. To mitigate the risk of hallucinations, our team initially developed our agents using the LLM Flan-T5-small (Chung et al., 2022). Rather than allowing the model to generate responses for students, we used the LLM to interpret students' input and match it with a human-written response from a knowledge bank. This approach required a significant amount of work by the design team to develop the knowledge bank, but it offered 100% certainty that the students would receive an appropriate, educator-written response.

Despite initial effectiveness in pilot tests, our team realized that the Flan-T5 model would require much more training data to effectively respond to the wide variety of possible questions and ideas shared by students. This encouraged our team to explore the potential trade-offs of testing generative approaches using larger LLMs. Our intent was to have more consistent, natural responses to the wide range of questions learners might pose, directing them toward next steps within the story, particularly suggesting when they might engage with other characters.

Transitioning to larger models meant dealing more urgently with issues of data privacy, sustainability, and environmental impact. Cutting-edge Llama and GPT models can (potentially) address our conversational needs, and yet each has their own tradeoffs. For example, while GPT-4o is a high-performing LLM due to its large number of parameters, it is a third-party model and thus raised concerns about student data privacy and how student input might be collected and stored amongst our focus group teachers (Hua et al., 2024).

Larger LLMs also require consideration of environmental impact, in terms of server energy usage, water consumption to cool servers, and long-term sustainability of using high-parameter models with large numbers of students (Selwyn, 2022).

Our team explored comparisons between Llama-3.1-8 billion (Grattafiori et al., 2024) and GPT-4o (OpenAI, 2023) to better understand whether Llama models, which are open-source and use fewer parameters, might be a reasonable alternative to mitigate privacy concerns while offering a more effective agent experience than Flan-T5-small. A lower number of parameters means that the model has a reduced capacity for language understanding and less detailed language knowledge to draw upon when analyzing student input and generating responses. However, how many parameters are necessary to generate useful responses for specific educational use cases remains an open question. We wanted to ensure we had an evidence-based argument for if, when, and why it might be worth it to use more powerful yet potentially problematic systems.

Our eventual decision to use the Llama model for our agents in SciStory was made because due to its open-source nature, we could run the SciStory agents on a university-owned server, meaning that all student conversation data would remain private and only accessible by the design team. The responses from the Llama model were more personalized and useful for learners than the responses from the smaller Flan-T5-small model, making the increase in energy usage worth the investment.

We also mitigated some of our concerns about students reading AI-generated responses through iterative testing with volunteer play testers. We recruited graduate students to "red team" the agents, meaning they intentionally submitted inappropriate questions and ideas to attempt to get the agent to say something inappropriate, to do the work for them, or to otherwise support off-topic or undesirable conversations. Using this iterative testing, we were able to implement an effective dialogue manager that limited conversations to topics that were student-appropriate, which reduced concerns about student safety when interacting with open-ended generative AI agents. The dialogue manager was also able to effectively catch and redirect use of language that was not school-appropriate, with the agents refusing to discuss ideas that used curse words or other disrespectful language. These safety features required us to test out several different, increasingly complex versions of prompt guidance that noted exactly what counted as "inappropriate" or "off-topic" and gave the agent strict instructions about how to respond to such inputs. While the agents are not perfect and sometimes miss some inappropriate slang words or other niche insults, in practice we saw high fidelity in their ability to note and redirect student attempts to steer conversations away from relevant game topics.

Unfortunately, hallucinations are one feature of generative AI agents for which our team has not yet found a satisfying solution. For science content questions, we found the agents to be quite accurate and reliable, given that they were successfully drawing upon the knowledge bases to answer student questions about pollinators, food systems, and ecosystem health. However, we noticed that when providing feedback to students about their arguments, the agents sometimes hallucinated possible evidence for students to consider. In the prompts, agents were given an outline of key ideas in the game and forms of evidence that students should draw on for their persuasive arguments. Yet we found the agents suggesting students talk about neighborhood housing prices in their argument or draw upon studies and statistics about the impact of gardens, even though neither of these types of evidence was listed in the agent's knowledge base for evidence available to students in SciStory. While these suggestions are not necessarily factually inaccurate, the agents appeared to invent new standards to use to assess the quality of students' argumentation, despite the fact that the prompt directed them to only consider the forms of evidence that are available to students in the game's narrative. The number of hallucinations overall remained low, and so we felt the usage of the Llama-based agent was still more beneficial for student learning than the more limited Flan-T5 agent. However, we are in the process of further refining the Llama agent prompts to find ways to reduce the frequency of hallucinated feedback standards.

CHALLENGES AND FUTURE DESIGNS

After our pilot design iterations were complete, our team tested SciStory: Pollinators with three 5th grade classrooms in rural Indiana in the spring of 2025. The experiences students shared with us about their interactions with the agents led us to several key pain points in the learner experience that we will be tackling for the next version of the SciStory design.

Balancing Feedback And Frustration

One of the key design challenges we identified was that it was difficult to get the agents to both critique students' argumentation sufficiently while also providing enough support and encouragement to avoid student frustration. We found that students were engaging in multiple rounds of argument revision and improvement while incorporating the feedback they received from the agents. This suggests that the agents were fulfilling their designed purpose of encouraging students to deepen their exploration of the problem and strengthen the quality of their persuasive writing.

However, students became frustrated if the agent continued to tell them that their argument was too vague or did not consider enough layers of the problem. Students would add detail to their argument that they believed addressed the

agent's issue, but then would receive fairly similar feedback a second or third time. This highlights a potential limitation of the current design, which is that it does not always acknowledge the nuance in students' improved responses (which is needed to offer emotionally supportive feedback that both notices strengths and points out actionable, specific ways to improve). While the system tracks argument history and improvements, its performance is constrained by the limited capacity (e.g., fewer parameters) of the Llama-3.1-8B 4-bit quantized model, which struggles with complex prompts. Prompt complexity increases with the amount, variety, and structural depth of information and instructions that the model must interpret simultaneously. It also tends to rise when the prompt includes multiple goals, conditional logic, or ambiguous language that requires contextual interpretation. Larger models such as GPT-4o may perform better with complex prompts, though with potential trade-offs in terms of environmental impact and long-term sustainability. Our team is currently exploring approaches to improve the performance of mid-size models such as Llama for these targeted learning tasks.

Altering the interaction structure might also impact student frustration. In the current design, students have to submit a new argument or new question in order to start another interaction with the agent. Only the current statement or response is shown on the screen as dialogue. However, we have considered moving to a back-and-forth style dialogue between learner and agent (similar to a text message exchange), which could offer time and space for more emotional support and more detailed comparisons between argument revisions to help students look back at their argument, consider the agent feedback, and locate places for improvement. In our next iteration of SciStory, we will be exploring ways to diversify agent responses and help students engage in more back-and-forth feedback cycles with the agents to reduce frustration.

Encouraging Help Seeking

Though teachers in our focus groups noted how important it was for students to learn how to identify when they need help and seek out resources, we found it was difficult to encourage students to interact with our help-seeking agent, Tulip. At any point during the game, players could click on a bee button in the corner of the screen to call Tulip and talk with her (see Figure 9). This design choice was made to address teacher suggestions that the agent should not force students to interact with it, but rather students should decide how and when they need assistance to promote agency. However, in practice we found that students rarely chose to interact with Tulip and thus did not often ask questions of the helper agent when it was easier to simply call over an adult.

FIGURE 9. When the player clicks on the button that looks like a bee, Tulip appears and offers multiple suggestions for what students can ask her. Depending on the choice, students will interact with either the helper agent or feedback agent.

Though students rarely sought out help from Tulip, we noticed growing frustration in several groups about where to go or what to do next that could have been addressed through conversations with the helper agent. For example, a group would get negative feedback from the Mayor agent that they needed more evidence for their argument, and they would complain that they didn't know what more evidence they needed or where to find it. Groups would also complain that they thought they had already talked to every NPC, but they would forget to take notes in their notebook and not understand why the game wouldn't allow them to progress. This suggests to us that we need to design further scaffolds to help students connect with the helper agent when it might be useful and to provide them with timely guidance when they get stuck. Our team has discussed implementing check-in points in the story where the player has to chat with Tulip before progressing, as well as more opportunities for Tulip to pop in during conversations to offer reminders of how she can help them when they are having difficulties.

CONCLUSION

Overall, the hope for the design of SciStory: Pollinators was to give students a playful, engaging learning experience with the following features:

- Agency to decide how to navigate a complex, real-world, open-ended problem
- Practice gathering, evaluating, and synthesizing a combination of facts and opinions in order to write a persuasive, nuanced argument
- Experience collaborating with peers to come to a consensus about potential solutions to a community problem

The integration of AI-driven conversational agents into the SciStory narrative was meant to fulfill three key roles:

1) answer questions about key topics, 2) provide feedback on argumentation, and 3) provide help and support when students encountered challenges. We found that our agents in their current design were quite effective at the first two roles, but they struggled to provide adequate and timely support to students as they encountered frustration during gameplay.

This raises questions for us about how to improve the design of AI agents to better mimic and supplement the pedagogical skills of teachers, while also leading us to consider the potential limitations of generative AI to fill these complex instructional roles. The agents provided important scaffolds to students' inquiry during gameplay, to an extent that would not be possible for one teacher in a classroom of 30 students. But we also encountered key points during gameplay where the facilitation of a skilled teacher could have encouraged students to deepen their collaboration and overcome frustration.

Through continued co-design with students and teachers, our design team hopes to further improve the effectiveness of SciStory for encouraging playful learning and engagement. We will also continue the refinement of AI agents to extend our understanding of the strengths and limitations of generative AI for supporting open-ended, student-led scientific inquiry.

ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Award No. DRL-2112635. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

We would like to thank Kara Cassell for her engineering and development work on the SciStory Unity version, as well as Justin Phillips and Courtney Barron for their 2D and 3D artwork.

Note on AI usage: Some images for SciStory: Pollinators (2D location backgrounds and 2D human character sprites) were generated using Canva's AI Image Generator. Generated images were digitally edited and revised by the design team. Tulip the honeybee was not AI-generated and was designed by artist Courtney Barron.

REFERENCES

Chung, H. W., Hou, L., Longpre, S., Zoph, B., Tay, Y., Fedus, W., ... & Dean, J. (2022). *Scaling instruction-finetuned language models.* https://doi.org/10.48550/arXiv.2210.11416

Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., ... & Vasic, P. (2024). The LLaMA 3 herd of models. https://doi.org/10.48550/arXiv.2407.21783

Hua, S., Jin, S., & Jiang, S. (2024). The limitations and ethical considerations of ChatGPT. *Data Intelligence, 6*(1), 201–239. https://doi.org/10.1162/dint_a_00243

Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., ... & Liu, T. (2025). A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems, 43*(2), Article 18. https://doi.org/10.1145/3703155

Humburg, M., Hong, D., Zhou, M., Feng, C., Danish, J., Hmelo-Silver, C. E., Wang, T., Glazewski, K., Kim, Y. J., Kumaran, V., & Lester, J. C. (2025). Using AI-driven conversational agents to support student-led collaborative scientific inquiry. In Oshima, J., Chen, B., Vogel, F., & Järvelä, J. (Eds.), *Proceedings of the 18th International Conference on Computer-Supported Collaborative Learning (CSCL 2025) (pp. 551–553).* International Society of the Learning Sciences. https://doi.org/10.22318/cscl2025.140384

Humburg, M., Dragnić-Cindrić, D., Hmelo-Silver, C. E., Glazewski, K., Lester, J. C., & Danish, J. A. (2024). Integrating youth perspectives into the design of AI-supported collaborative learning environments. *Education Sciences*, *14*(11), 1197. https://doi.org/10.3390/educsci14111197

McLaren, B. M., & Nguyen, H. A. (2023). Digital learning games in artificial intelligence in education (AIED): A review. In U. Cress, C. Rosé, A. F. Wise, & J. Malmberg (Eds.), *Handbook of Artificial Intelligence in Education*, (pp. 440–484). Edward Elgar Publishing. https://doi.org/10.4337/9781800375413.00032

OpenAI. (2023). GPT-4 technical report. https://doi.org/10.48550/arXiv.2303.08774

Sadler, T. D. (2004). Informal reasoning regarding socioscientific issues: A critical review of research. *Journal of Research in Science Teaching, 41*(5), 513–536.

https://doi.org/10.1002/tea.20009

Saleh, A., Hmelo-Silver, C. E., Glazewski, K. D., Mott, B., Chen, Y., Rowe, J. P., & Lester, J. C. (2019). Collaborative inquiry play: A design case to frame integration of collaborative problem solving with story-centric games. *Information and Learning Sciences, 120*(9/10), 547–566. https://doi.org/10.1108/ILS-03-2019-0024

Selwyn, N. (2022). The future of AI and education: Some cautionary notes. *European Journal of Education, 57*(4), 620-631. https://doi.org/10.1111/ejed.12532