Conversational Approach to the Creative Authoring of Intelligent Virtual Characters

ABSTRACT

The authoring of intelligent, personable, unique virtual characters relies on the technical expertise of domain architects and the creative input of writers and artists alike. The conventional authoring workflow is heavily driven by engineering needs and burdens creatives with inconsistent and cumbersome tasks that leave little freedom for imagination and improvisation. To better assist the collaboration between creatives and domain architects, we present an authoring workflow enabled by CO-AUTHOR (COnversational AUTHORing interface), a natural language interface that guides the author to construct the character's domain knowledge and dialogues through a series of conversational interactions. CO-AUTHOR collects data in a scene-based format, which can be easily used to extract domain knowledge. The goal of this work is to explore the potential of conversational interface design in inspiring creativity in virtual character authoring and facilitating structured interdisciplinary communication. User studies showed that CO-AUTHOR is usable, learnable, and effective for designing unique virtual characters.

KEYWORDS

Conversational authoring interface; natural language interface; intelligent virtual agent

ACM Reference format:

. 2019. Conversational Approach to the Creative Authoring of Intelligent Virtual Characters. In *Proceedings of ACM Conference, Washington, DC, USA, July 2017 (Conference'17),* 8 pages.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In video games and other kinds of interactive narratives, intelligent virtual characters have the magical power of carrying us into worlds that transcend everyday life. Different from utility-based virtual assistants such as Siri and Alexa, virtual characters possess not only intelligence, but emotive capabilities and unique personalities.

The collaboration between creatives and domain architects are crucial for bringing such virtual characters to life. Domain architects are technical experts who are tasked with formally designing intelligent virtual characters' domain knowledge, which is a symbolic representation of knowledge that the character uses to reason over its interactions with other agents. In the context of this work, domain knowledge encompasses the mental modeling of the character. Although the creation of interactive narratives requires substantial engineering expertise, it is also necessary to pick the brains of writers, artists, and animators alike to give the characters a boost of peculiarities. This intrinsically collaborative and interdisciplinary process brings about the challenge of bridging different mindsets and workflows in an efficient and effective way.

The conventional authoring process for virtual characters is heavily driven by engineering needs (shown in Figure 1a). This process burdens creative authors with inconsistent and cumbersome tasks, leaving little room for imagination and improvisation. As the intelligent system goes through updates, creatives are forced to adjust to new tools and take on new tasks in order to satisfy demands for creative input. Inconsistency and the lack of formality result in ineffective communication, repetitive tasks, underused data, and consequently, content of compromised quality.

There is a need to design a more consistent, structured, and enjoyable authoring process that not only adheres to engineering requirement, but respects the way creative authors think and work. In this research, we explore the use of a conversational interface to orchestrate a collaborative authoring workflow (shown in Figure 1b) that emphasizes the balance between creative and technical voices.

The design of this interface, CO-AUTHOR (COnversational AUTHoring interface), takes advantage of the power of natural language and facilitates an authoring experience that emphasizes the creative freedom offered by writing. We value the use of natural language because it is the most intuitive medium for human beings to tell stories. Through the art of writing, we are able to brainstorm and experiment with unlimited potential. We get to think deeply about what makes a character unique, personable, and interesting, and what allows a human being to connect emotionally with a story.

To simulate the creative writing experience, CO-AUTHOR guides the author through sessions where (s)he writes short, atomic interactive scenes involving the character and the user. In each session, the author describes the character's action, the consequence of the action, the reasoning behind it, a sample dialogue that accompanies this scene, etc. The data collected through these conversations is modularized in a "scene-based" format. As data accumulates, domain architects refer to this reservoir of creative content when programming a character's domain knowledge and dialogue model. It should be noted that CO-AUTHOR is agnostic to the embodiment of the character.

The contribution of this work includes: a proposition for a collaborative authoring workflow that makes use of CO-AUTHOR; a proof-of-concept prototype of CO-AUTHOR; and an evaluation of the usability and effectiveness of the prototype in the context of the proposed collaborative authoring workflow.

The rest of this paper is structured as follow. Section 2 reviews existing works that address the virtual character authoring problem. Section 3 provides an overview of the proposed workflow for virtual

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM. . . $15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

character authoring, in comparison to the conventional engineering-driven workflow. Section 4 illustrates the user experience design for CO-AUTHOR. Section 5 presents technical details about the natural language processing module which CO-AUTHOR employs. Section 6 describes user studies and evaluations. Section 7 discusses limitations exposed by our experimentation as well as future directions for further improvement upon CO-AUTHOR and the general authoring workflow for virtual characters.

2 RELATED WORK

The authoring challenge that this work hopes to address aligns with existing lines of inquiries which focus on the authoring of non-linear narratives We divide previous work into historical approaches that shed light on the evolution of the creative authoring process and natural language approaches that relate to the novel approach that we propose.

A major cause for the difficulty of authoring interactive narratives is their non-linear structure. Authoring tools with graphical interfaces empower the author by visualizing non-linear narrative content. These visualizations tend to follow a graph-based structure, which is generally consistent with the symbolic structure of the target information space. Visual environments have the benefits of transparency and clarity.

2.1 Historical Approaches

In 1996, Improv [19] provided a scripting interface for creative experts to define behavioral rules that agents would follow. It also adopted a separation between the behaviors of an agent and the animations of an agent, which allowed creative experts to alternate focus between behaviors and animations. However, its philosophy was that the influence of these authors was in "carefully tuning [the] probabilities" of an agent executing certain behaviors, showing that at the time, agents were made shallow [19]. A similar division between content and processing can be observed among several markup-based interfaces [2, 8, 12–14, 20]. For example, the BEAT toolkit [3] used a markup language in order to automatically animate an authored natural language script for a virtual character [30, 31].

In 2002, BrainFrame [7] introduced a graphical editor for game designers to program behaviors with, in which behaviors were represented as finite-state machines. The authoring tool then passed behavior descriptions to an engine that implemented the behaviors. Four years later, IN-TALE [23] used a scripting interface for authors to represent behaviors as trees. However, instead of depending on authors to orchestrate behaviors in large like in prior works, IN-TALE relied on an automated story director to immerse the user, lessening some of the burden placed on authors. In 2015, Poulakos et al. [21] formalised event-centric domain knowledge, which consists of re-usable context-specific behaviors. This was demonstrated using Parameterized Behavior Trees, which are graph-based, hierarchical representations of behaviors. Additionally, Poulakos et al. described steps to building a story world that separated the definition of agents' roles and relationships from that of their behaviors, which laid the groundwork for automated story planning systems [11] that would largely remove the need for authors to deal with the non-linearity of interactive narratives. Nearly 2 decades of progression in authoring tools has steadily increased the impact that creatives could have when authoring agents by decreasing low-level (i.e., animations) and high-level (i.e., non-linearity) burdens, but the culminating behavior representation is still too complex for non-programming creative authors.

Each of the prior works can represent agent behaviors as graphs. Therefore, we generalize content authoring as a knowledge acquisition problem for graph-based information. Jambalaya [27] is a proposed solution for this general problem that applies SHriMP [28]—a visualization technique for hierarchically structured information spaces—to a knowledge management environment that helps domain experts navigate through complex information hierarchies. SHriMP faces the challenge of managing visual complexity, and consequently, some users have suggested either (a) hiding the arcs drawn between related components by default or (b) incorporating filtering mechanisms that hide irrelevant arcs. This implies that hiding the complexity would afford better focus to users.

2.2 Natural Language Approaches

CO-AUTHOR can be considered as the total embodiment of this focus, because it hides everything but the scene in progress. This implicitly discourages the user from thinking about the exponential branching of interactions and instead allows the user to concentrate on enriching deep interactions. In doing so, we challenge the assertion made by Perlin and Goldberg [19] that it is insufficient to provide an author with a "tool for scripting long linear sequences," and we explore the potential of an untapped approach. We posit that a natural language interface would more efficiently facilitate this than a graphical or scripting interface.

March22 [16] is a prototype that lies between graphical/scripting and natural language interfaces. Authors use plain language to write the script for a given scene and a programming language to transition between scenes based on a user's decision. This is accompanied by a visualization of the transitions between scenes. Our approach differs in that we incorporate neither a graphical nor a scripting interface. Other natural language interfaces include Façade [18], which performs real-time story comprehension in a gaming environment, interpreting the player's natural language input to push forward the storyline. Chaturvedi et al. [4] understands human relationships from texts. STAR [5] guides the user to write linear narratives and answer logical questions while extracting world knowledge and inference rules. Sanghrajka et al. [25] introduces LISA, which dynamically builds knowledge base while the author is writing. Building upon LISA, CARDINAL [17] is a combination of the graphical and natural language approaches. It allows creatives to write movie scripts using natural language and automatically creates meaningful visualizations and instant 3D animated pre-visualization of the script as it is being written [17].

2.3 In Comparison to Related Works

A common goal among research described above is to automate engineering work that is currently done by humans so that non-technical experts can author interactive narratives without the assistance of domain architects. Similarly, our work aims to empower creatives. However, instead of replacing domain architects in the authoring process, we aim to enable more seamless collaborations between different talents. In particular, we focus on the authoring of creative content, but the interface that we propose can be used for practical knowledge as well. The user experience is designed to be enjoyable and author-centric, which we try to achieve by adopting the natural language approach.

3 OVERVIEW

In this section, we present a collaborative authoring workflow which improves upon the conventional, engineering-oriented workflow. The improved workflow incorporates CO-AUTHOR to better integrate the creative and engineering processes. We will also discuss the benefits of using a conversational interface rather than a graphical one.

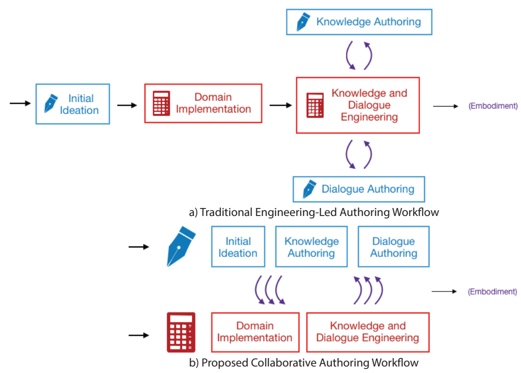

Figure 1: (a) Conventional engineering-led authoring workflow which decouples creative tasks including initial ideation, knowledge authoring, and dialogue authoring, which center around engineering-related tasks. (b) Proposed workflow which tightly couples all creative tasks and places them in parallel to engineering tasks. Creative tasks are in blue; engineering tasks are in red.

3.1 Traditional engineering-oriented authoring workflow

Since the intelligent system is the most fundamental component of any intelligent virtual character, the authoring pipeline is heavily driven by engineering needs. Figure 1a provides a high-level visualization of this engineering-led authoring pipeline.

The intuitive order of the authoring process goes from left to right. Through blue-sky ideation, the author designs the story world and creates initial outlines for the character. This lays the foundation for developing a domain structure which stores the mechanisms that dictate how the virtual character acts emotionally, physically, and verbally in interactive narratives. Domain architects will then populate the domain with content provided by creative authors. Traditionally, the authoring of knowledge and dialogues are done separately. Depending on engineering needs, the author is often asked to provide content in multiple formats using a variety of tools. Lastly, behaviors and dialogues of the characters in the context of each narrative are embodied and performed in visual and audio means.

A main disadvantage of this workflow is the lack of balance between creative and engineering voices. As shown in Figure 1a, the central thread of the authoring pipeline is mostly led by engineering (in red). Creatives (in blue) are tasked to provide content that serves the system. To cater to the system's needs, they often need to switch between tools and platforms or use multiple at the same time. For example, an author may be asked to use commercial softwares to write non-linear dialogues, while they are also tasked to write character sheets and narrative scripts in Microsoft Word. This process results in inefficient collaboration, cumbersome authoring experiences for creatives, and eventually, output of compromised quality.

An idealistic approach to empower creative authors in this heavily engineering-driven workflow could be to give them the ability to intervene in technical tasks. Conceptually, the intervention of creatives would involve coupling or automating multiple tasks in the authoring pipeline. This synthesis would be extremely challenging because the authoring tasks in this workflow involve different cognitive processes. For example, the *initial ideation* step values creativity and imagination, which are most effectively carried through writing, drawing, or other traditional means of storytelling that creatives are more familiar with, whereas *domain implementation* demands abstract, high-level thinking and requires a different set of technical expertise. Additionally, a writer may attempt to author *knowledge* and *dialogue* in the same workflow, possibly by writing utterances while giving "stage directions." However, these stage directions must translate to states in the domain knowledge, and it would be extremely burdensome for the author to think in the logical order of the domain specification and consider the combinatorics of state variables. With sufficiently-robust automation tools, it could potentially be possible for creative authors without technical expertise to directly maneuver the intelligent system. Practically, however, the work of domain architects is irreplaceable to ensure optimal domain implementation and accurate translation from the author's creative vision to code.

3.2 Proposed Collaborative Authoring Workflow

We acknowledge the technical and cognitive difficulties of automating engineering work or synthesizing cognitively different tasks. We propose an alternative workflow, shown in Figure 1b, which bypasses these challenges. While still treating the same set of tasks in a discrete manner, this new workflow *reorganizes* them to orchestrate a more structured, fluent, and enjoyable collaborative authoring experience.

This workflow makes uses of CO-AUTHOR, a conversational interface which incorporates all tasks involving creative authors—initial ideation, knowledge authoring, and dialogue authoring—into one single platform. Within the same overall conversational interaction, it first leads the author through an ideation process where (s)he describes the character's personality and abilities, followed by an iterative narrative scripting process. In each scripting session, the author is prompted to write a short interactive "scene" between the virtual character and a human being, as well as a sample dialogue in this narrative. Essentially, the writer experiences *initial ideation* as a starting step, and then in each session goes through *knowledge* and *dialogue authoring* sequentially. The three modes are still treated as discrete cognitive tasks but are integrated within one experience.

Using this tool, the overall authoring pipeline becomes a fully collaborative cycle where creative authors and domain architects work in parallel. The creative content provided by authors is modularized such that each scene can be viewed as an independent entity. As the author accumulatively generates scenes, domain architects can work simultaneously and use the available creative content to build the character's domain knowledge and dialogue model. From each scene, meaningful information corresponding to different aspects of the character's domain knowledge—affordances, emotion rules, motivation rules—can be manually or automatically extracted.

3.3 Benefits of the Conversational Approach

In the context of virtual character authoring, conversational interfaces have two main advantages over visual interfaces. Firstly, conversational interface has less cognitive overhead. The user focuses on one narrative at a time, instead of managing a visual system of content. Secondly, conversational authoring encourages the author to translate a linear storytelling experience into scenes, creating one rich path through the interactive narrative at a time. A fundamental mission of this tool is to encourage authors to write creatively. Thus, we adopt the conversational approach to more effectively bring the author into his or her creative mindset.

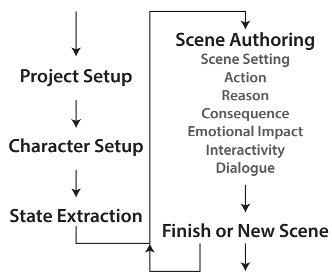

Figure 2: CO-AUTHOR orchestrates a creative-facing authoring experience that consists of five dialogue segments. Starting with an ideation process (left), the author is guided to write small scenes involving the character and a human being while having the option to save and edit after writing each scene (right).

4 USER EXPERIENCE DESIGN

Using CO-AUTHOR, the author goes through five dialogue segments in an authoring session (Figure 2). In the first three segments—project setup, character setup, and state extraction—the author engages in a process of ideation regarding the character and a human audience. The author describes possibilities in the story world, which would be extracted into *states* in the domain knowledge. In scene authoring, the author provides detailed information to paint the picture of a small scene, serving the goal of knowledge and dialogue authoring. In the last segment, the author has the choice of saving and exit, or going back to the previous segment and author a new scene.

Figure 3 illustrates the scene authoring segment for a character named "Roberto the Raccoon." For the sake of demonstration, the author's inputs in this example are very simple, although in practice, authors can provide much more.

Project Setup. The author is prompted to name the project and the main character of the scene. The input for character name, which is "Roberto" in this example, will be used to generate more personalized questions in later interactions.

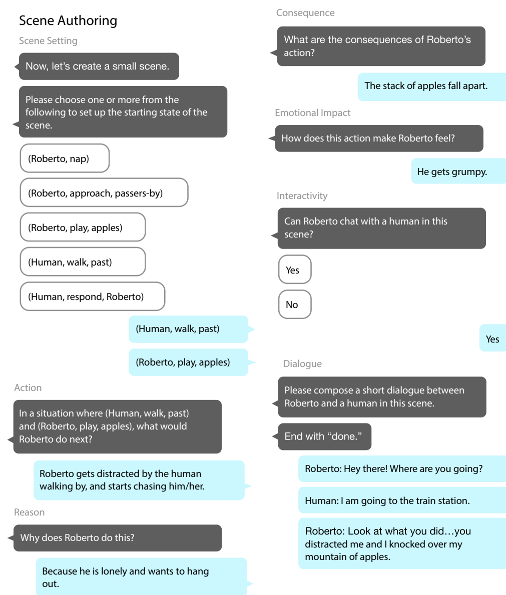

Figure 3: The author uses CO-AUTHOR (black) to create a character named "Roberto the Raccoon." This image illustrates a simple authoring session where the author (blue) writes a small scene involving Roberto and a passer-by. White bubbles are options given by the interface in multiple choice questions.

Character Setup. The interface asks the author, "What is Roberto like? Please give a general description." The author is allowed to submit multiple responses and end with the keyword "done."

State Extraction. The author is asked three questions regarding: the physical and emotional capabilities of the character; the physical and emotional capabilities of the user; and things that can happen in the environment that do not involve the agents, e.g., weather and traffic. The goal of this segment is to obtain information about possible states in the world. Using a state extraction module, which is described in Section 5.1, the interface converts the author's natural language responses into abstract representations of states to be used in the domain knowledge.

Scene Authoring. In this step, the author is guided to describe a scene by answering a series of questions.

Scene Setting has a multiple choice question where the author is asked to choose from the states extracted from the previous segment in order to set up the beginning of a scene. As shown in Figure 2, multiple choice options are presented as white, clickable buttons. Each state is shown as a triplet. Although this abstract state representation is an outlier within the overall interaction, which is in natural language, we elected to use this format to present a clear view of all discrete possibilities in the story world. In Figure 2, we are showing an incomplete list of all states parsed from the author's previous response for demonstrative purposes. By selecting *(Human, walk, past)* and *(Roberto, play, apples)*, the author is essentially describing a beginning where "a human being is walking past Roberto while he is playing with his apples."

In the Action segment, the author describes what the character would do in this situation. The next segments, ask the author to describe the reasoning behind, consequence of, and emotional impact of the character's action. In the Interactivity segment, the author indicates that this interaction can potentially have a dialogue component to it. Consequently, the interface asks the author to compose a short dialogue between Roberto and the human in this scene.

Finish or New Scene. After the scene is finished, the author has the choice of either saving progress and exiting, or starting a new scene, returning to Scene Authoring. Given that this work is prototypical, there are currently no editing capabilities. This is better suited for a visual interface.

By posing questions in a defined sequence, this interface guides the author to think within confined scenarios. Despite the rigidness of the sequence, each question is designed to be open-ended and invites improvisation within the scope of each individual answer.

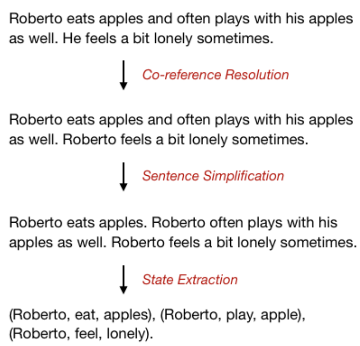

Figure 4: The natural language processing module parses the author's natural language description of the character's physical and emotive capabilities into abstract representations of states.

5 IMPLEMENTATION DETAILS

5.1 Natural Language Processing for Domain Knowledge Extraction

We use a natural language processing module to parse the author's natural language input—descriptions of the character's emotive and physical capabilities and possible happenings in the interactive story world—into states in the form of triplets: subject, verb (+ object) and prepositional complement (+ object). The extraction steps are illustrated in Figure 4.

Coreference Resolution. The first step of the extraction task is to resolve ambiguities in entity references. This involves finding all expressions that refer to the same entity in a text [22].

Sentence Simplification. The co-reference normalized sentences are then simplified into sentences with a single main verb using rules based on lexical substitution and syntactic reduction techniques [24]. Each simple sentence is parsed for syntactic dependencies between words, which can be used to extract state information [9].

State Extraction. Each simplified sentence corresponds to a state that is either a binary or fluent type. This identification is made based on the presence of certain keywords; e.g., "such as" indicates a fluent state. Depending on the inferred state of the sentence, extraction is done with different approaches using Spacy [10].

Fluent sentences often contain other key phrases such as "including" and "consist of," which make their syntactic structure more predictable. Generally, key phrases in fluent sentences are prepositional modifiers, and the words that immediately follow tend to carry important information and have propositional complement dependency relationship with the previous key phrase. For these sentences, we take into consideration the prepositional modifier and prepositional compliments when extracting states. By contrast, binary sentences have more diverse syntactic patterns. Other than the subject, verbs, and objects, we take into consideration prepositional modifiers, adjectival compliments, and open clausal complements in the extraction process.

Through this process, we form a list of states—each containing a discrete possibility in the story world—from the author's natural language descriptions.

5.2 Prototype

The prototype of `CO-AUTHOR` was created as a web application that operates on the browser. The visual graphics is generated using p5.js, a client-side Javascript library for graphic design [15]. The web backend uses the Node.js run-time environment [6].

6 USER STUDY AND EVALUATIONS

We performed user testing to evaluate the usability and effectiveness of the conversational interface. We performed the same test across 14 subjects within the age range 18-35 from the Greater Los Angeles area. Five subjects were familiar with the problem space of virtual character authoring, while the other nine were introduced to it for the first time. All subjects used the same desktop computer in the same location and environment for the tests. Every testing session consisted of two rounds. In the first round, all subjects were introduced to `CO-AUTHOR` for the first time and they were all instructed to author the same character. The purpose of this round of structured testing was to evaluate the usability and learnability of the tool, and to get the subjects' initial responses to the user experience. In the second round, as the subjects were more familiar with the tool, their task became authoring a fictional character of their own choice. The goal was to evaluate whether the tool provides enjoyment and encourages creativity. The subjects were tasked to fill out a different questionnaire after each round.

Based on qualitative and quantitative analyses over the questionnaire results, we found that the tool offers a favorable user experience and gives authors the ability to create a variety of unique virtual characters. In this section, we will present the process and results of both rounds of user testing.

6.1 Structured testing

6.1.1 Methods. The structured testing round was designed to ensure a level of consistency across the authoring experiences of all subjects so that we could get a fair evaluation of the usability and effectiveness of the system. The subject was first given a short

Property	Statement	Agreement (0-4)
Usability	I think that I would like to use this system frequently.	2.86
	I found the system unnecessarily complex.	0.64
	I thought the system was easy to use.	3.36
	I think that I would need the support of a technical person to be able to use this system.	0.64
	I found the various functions in this system were well integrated.	3.21
	I thought there was too much inconsistency in this system.	0.64
	I would imagine that most people would learn to use this system very quickly.	3.43
	I found the system very cumbersome to use.	0.79
	I felt very confident using the system.	2.79
	I needed to learn a lot of things before I could get going with this system.	0.50
Redundancy	Some questions were redundant in this experience.	1.14
Clarity	Some questions were unclear in this experience.	1.43
Completeness	I wish more questions were asked to help me author the character.	1.64

Table 1: In the structured testing round, all subjects were asked to author the same character. The goal was to evaluate the usability, redundancy, clarity, and completeness of the system. The subject was asked to rate their agreement to each of the statements above from 0 (strongly disagree) to 4 (strongly agree). The average score across all subjects to each statement is listed in the right-most column.

Mean	Median	Standard Deviation
81.07	82.5	11.55

Table 2: Result of SUS testing. SUS scores are scaled from 0 to 100. The score of 68 is considered average.

tutorial to get familiarized with the user interface and the general workflow. Then, they were tasked with using CO-AUTHOR to author a specific fictional character—the same across all subjects—and finish one complete authoring session (at least one scene). After the subject finished using the tool, we asked them to fill out a questionnaire. The questionnaire included the System Usability Scale (SUS) to measure the usability and learnability of the tool. It also included agreement-scale questions about clarity (e.g., "Some questions were unclear"), redundancy (e.g., "Some questions were redundant"), and completeness (e.g., "I wish more questions were asked to help me author the character"). The subject was allowed to provide general comments and suggestions at the end of the questionnaire.

6.1.2 Results. A statistical summary of structured testing is shown in Table 1. The result from the SUS is shown in Table 2.

Usability. SUS as a method for evaluating usability is fitting for this research because it is proven to be robust, reliable, and generalizable across a variety of tasks [26], even if there are as few as 8-12 test subjects [29]. SUS scores are scaled from 0 to 100, where a score of 68 is considered average. Based on a mapping of scaled SUS scores to common adjectives [1], the mean and medium scores across 14 subjects indicate that CO-AUTHOR falls within the range of "good" and "excellent," which implies that the tool is highly usable and learnable.

Redundancy. The average agreement score to the statement "some questions were redundant in this experience" is 1.14 out of 4.0. The questions that were identified as redundant were those of the Emotional Impact and Interactivity segments. Subjects found Emotional Impact redundant because it has overlaps with Consequence. When an author uses the tool for the first time, (s)he might include information on the emotional impact of the character's action as part of the answer to the Consequence segment. This issue could be addressed by preconditioning the author with more specific constraints on the two segments.

The Interactivity segment asks the author whether the character and the human can carry a conversation in this scene. The answer to this question can potentially be implied from answers to previous questions. For example, several subjects in the testing session set up scenes where the character greets a human. In this case, we can safely assume that a conversation exists in the scene. This issue could be avoided by dynamic question generation, which takes into account earlier input when creating new questions.

Clarity. The average agreement score to the statement "some questions were unclear in this experience" is 1.5 out of 4.0. The question that most subjects identified as unclear was "In the story world, what is the character capable of doing and feeling?" from the Character Setup. Subjects found this question to be awkwardly phrased and not immediately clear as to what kind of answer it expected. We designed this question to elicit information about possible states of the character within the story world. We intentionally phrased the question in an open-ended way to invite creativity. However, this decision has led to confusions.

Completeness. The average agreement score to the statement "I wish more questions were asked to help me author the character" is 1.7 out of 4.0. Several subjects wished to author additional information about the environment and the demographics of the humans in these scenes.

6.2 Unstructured testing

6.2.1 Methods. The goal of unstructured testing was to evaluate user experience when the author was able to use the tool creatively without constraint. Each subject was instructed to author a fictional character of their own choice. Then, they were asked to fill out a survey that included agreement-scale questions about enjoyment (e.g., "I felt natural chatting in this tool"), creative freedom (e.g., "The tool

Property	Statement	Agreement (0-4)
Character generalizability	I was able to use this tool to author characters that are distinct from each other.	3.43
Enjoyment and creative freedom	I felt natural chatting in this tool.	2.71
	I enjoyed the conversational flow.	2.57
	The interaction felt awkward.	1.00
	The experience was repetitive.	1.71
	I was given creative freedom when I used this tool to author the characters.	3.43
	The tool invited me to o er creative insights.	2.93

Table 3: In the unstructured testing round, all subjects were asked to author a character of their choice. The goal was to evaluate the general enjoyment of the user experience and whether the tool can be used for authoring a variety of characters. The subject was asked to rate their agreement to each of the statements above from 0 (strongly disagree) to 4 (strongly agree). The average score across all subjects to each statement is listed in the right-most column.

invited me to offer creative insights"), and character generalizability ("I was able to use this tool to author characters that are distinct from each other"). Subject were invited to provide general feedback.

6.2.2 Results. A statistical summary of unstructured testing is shown in Table 3.

Character Generalizability. Most subjects agreed that they were able to create multiple characters with distinct personalities using CO-AUTHOR, giving an average score of 3.43 out of 4.

Enjoyment and Creative Freedom. Subjects had mediocre responses when asked questions in this category. According to the subjects' feedback, even though the interaction is designed as a conversation, it is still highly scaffolded and repetitive. The authors are allowed to write freely when answering individual questions but they do not have higher-level autonomy over each scene or the authoring experience as a whole.

7 CONCLUSIONS

This work demonstrates an efficient means to orchestrate a fluent conversational authoring experience that amplifies creative voices in the engineering-oriented authoring pipeline for intelligent virtual characters. In order to take the leap from utility-based conversational agents to unique, interesting characters that bring immersion into narrative experiences, it is crucial that input from creative talent is properly gathered and employed in the overall authoring process. The design of CO-AUTHOR is rooted in the belief that human creativity can be better incorporated into intelligent virtual characters by using the appropriate interface, which we have hypothesized is conversational. With the proof-of-concept prototype, we have demonstrated the value of a conversational interface in facilitating an author-centric workflow.

7.1 Limitations

For actual usage within an authoring pipeline, CO-AUTHOR would benefit from more robust automation which allows data to be more seamlessly absorbed into the intelligent system.

Additionally, user studies exposed minor issues with the conversational user experience, such as question generation and dialogue flow. The phrasing of a few questions reduced clarity and caused some redundancy in authors' question answering. In large, the responses from CO-AUTHOR are repeated for every scene that an author creates, which may result in redundancy in CO-AUTHOR . However, we are less concerned about the latter type of redundancy, because the author will become familiarized with the order of questions enough not to read them every time.

Finally, we recognize that both editing previous answers for the scene in progress and deleting or modifying already authored scenes are more cumbersome with a conversational user interface, than with a graphical interface. However, the benefits of a conversational approach far outweigh this limitation. Eventually, it will be beneficial to consider a hybrid conversational-graphical interface model to address this concern.

7.2 Future work

Dynamic question generation could be an effective solution for the redundancy concerns of conversational interaction. For example, based on known information about the character and the story world, CO-AUTHOR can learn to avoid unnecessary questions and highlight those that are more interesting or important. To enrich each scene, it may also ask the author for additional input about time of day, day of week, and demographic information of the human that this interaction targets towards. As the conversational experience becomes more personalized and refreshing, the author would be more likely to enjoy the process.

The current design of CO-AUTHOR is mostly as a data collection machine, which shifts the responsibility of knowledge extraction entirely to domain architects. The overall authoring pipeline could benefit from additional NLP functionality that alleviates domain architects' burden and make the overall collaboration more fluent and efficient.

As a chatbot, CO-AUTHOR does not have any distinct characteristics in itself, different from the intelligent virtual character that it is being used to author. A future version of this interface could embody a personality of its own or the personality of the character being authored. The user may engage with the character in *performance mode* —teaching it new behaviors and utterances through chatting with it directly, like a teacher talking to a student. Explorations along this direction could lead to new paradigms of authoring and interactive story experiences.

REFERENCES

[1] Aaron Bangor, Philip Kortum, and James Miller. 2009. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *J. Usability Studies* 4, 3 (May 2009), 114–123. http://dl.acm.org/citation.cfm?id=2835587.2835589

[2] Berardina De Carolis, Catherine Pelachaud, Isabella Poggi, and Mark Steedman. 2004. APML, a Markup Language for Believable Behavior Generation. In *Life-like characters*.

[3] Justine Cassell, Hannes Högni Vilhjálmsson, and Timothy Bickmore. 2001. BEAT: The Behavior Expression Animation Toolkit. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '01)*.

[4] Snigdha Chaturvedi, Mohit Iyyer, and Hal Daumé III. 2017. Unsupervised Learning of Evolving Relationships Between Literary Characters.. In *AAAI*. 3159–3165.

[5] Irene-Anna Diakidoy, Antonis Kakas, Loizos Michael, and Rob Miller. 2015. STAR: A system of argumentation for story comprehension and beyond. In *Proceedings of the 12th International Symposium on Logical Formalizations of Commonsense Reasoning*. 64–70.

[6] Node.js Foundation. 2018. Node.js. (2018). https://nodejs.org/en/

[7] Daniel Fu and Ryan T. Houlette. 2002. Putting AI in Entertainment: An AI Authoring Tool for Simulation and Games. *IEEE Intelligent Systems* 17 (2002), 81–84.

[8] Dirk Heylen, Stefan Kopp, Stacy Marsella, Catherine Pelachaud, and Hannes Högni Vilhjálmsson. 2008. The Next Step towards a Function Markup Language. In *IVA*.

[9] Matthew Honnibal and Mark Johnson. 2015. An Improved Non-monotonic Transition System for Dependency Parsing. In *EMNLP*.

[10] M. Honnibal and I. Montani. 2017. spaCy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. (2017). https://spacy.io/

[11] Mubbasir Kapadia, Seth Frey, Alexander Shoulson, Robert W. Sumner, and Markus H. Gross. 2016. CANVAS: computer-assisted narrative animation synthesis. In *Symposium on Computer Animation*.

[12] Stefan Kopp, Bernhard Jung, Nadine Pfeiffer-Le, and Ipke Wachsmuth. 2003. Max - A Multimodal Assistant in Virtual Reality Construction. *KI* (2003).

[13] Stefan Kopp, Brigitte Krenn, Stacy Marsella, Andrew N. Marshall, Catherine Pelachaud, Hannes Pirker, Kristinn R. Thórisson, and Hannes Vilhjálmsson. 2006. Towards a Common Framework for Multimodal Generation: The Behavior Markup Language. In *Intelligent Virtual Agents*.

[14] Stefan Kopp and Ipke Wachsmuth. 2004. Synthesizing Multimodal Utterances for Conversational Agents: Research Articles. *Comput. Animat. Virtual Worlds* 15, 1 (March 2004).

[15] NYU ITP et al. Lauren McCarthy, Processing Foundation. 2018. P5.js. (2018). https://p5js.org/

[16] Samuel Lynch, Charlie Hargood, and Fred Charles. 2017. Textual Authoring for Interactive Narrative. (2017).

[17] Marcel Marti, Jodok Vieli, Wojciech Witoń, Rushit Sanghrajka, Daniel Inversini, Diana Wotruba, Isabel Simo, Sasha Schriber, Mubbasir Kapadia, and Markus Gross. 2018. CARDINAL: Computer Assisted Authoring of Movie Scripts. In *23rd International Conference on Intelligent User Interfaces (IUI '18)*. ACM, New York, NY, USA, 509–519. https://doi.org/10.1145/3172944.3172972

[18] Michael Mateas and Andrew Stern. 2003. FaÃğade: An Experiment in Building a Fully-Realized Interactive Drama. (04 2003).

[19] Ken Perlin and Athomas Goldberg. 1996. Improv: A System for Scripting Interactive Actors in Virtual Worlds. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '96)*. ACM, New York, NY, USA, 205–216. https://doi.org/10.1145/237170.237258

[20] Paul Piwek, Brigitte Krenn, Marc Schröder, Martine Grice, Stefan Baumann, and Hannes Pirker. 2002. RRL: A Rich Representation Language for the Description of Agent Behaviour in NECA. *CoRR* (2002).

[21] Steven Poulakos, Mubbasir Kapadia, Andrea Schüpfer, Fabio Züd, Robert Sumner, and Markus Gross. 2015. Towards an Accessible Interface for Story World Building. (2015). http://aaai.org/ocs/index.php/AIIDE/AIIDE15/paper/view/11583

[22] Marta Recasens, Lluís Màrquez, Emili Sapena, M. Antònia Martí, Mariona Taulé, Véronique Hoste, Massimo Poesio, and Yannick Versley. 2010. SemEval-2010 Task 1: Coreference Resolution in Multiple Languages. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval '10)*. Association for Computational Linguistics, Stroudsburg, PA, USA, 1–8. http://dl.acm.org/citation.cfm?id=1859664.1859665

[23] Mark O. Riedl and Andrew Stern. 2006. Believable Agents and Intelligent Story Adaptation for Interactive Storytelling. In *Proceedings of the Third International Conference on Technologies for Interactive Digital Storytelling and Entertainment (TIDSE'06)*. Springer-Verlag, Berlin, Heidelberg, 1–12. https://doi.org/10.1007/11944577_1

[24] Horacio Saggion. 2017. *Automatic Text Simplification*. Morgan & Claypool Publishers. https://doi.org/10.2200/S00700ED1V01Y201602HLT032

[25] Rushit Sanghrajka, Daniel Hidalgo, Patrick P Chen, and Mubbasir Kapadia. 2017. Lisa: Lexically intelligent story assistant. In *Proceedings of the 13th Artificial Intelligence and Interactive Digital Entertainment Conference*.

[26] Jeff Sauro and James R. Lewis. 2011. When Designing Usability Questionnaires, Does It Hurt to Be Positive?. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2215–2224. https://doi.org/10.1145/1978942.1979266

[27] Margaret-Anne Storey, Mark Musen, John Silva, Casey Best, Neil Ernst, Ray Fergerson, and Natasha Noy. 2001. Jambalaya: Interactive visualization to enhance ontology authoring and knowledge acquisition in Protégé. In *in Protégé. Workshop on Interactive Tools for Knowledge Capture K-CAP-2001*.

[28] M.-A. D. Storey, K. Wong, F. D. Fracchia, and H. A. Mueller. 1997. On Integrating Visualization Techniques for Effective Software Exploration. In *Proceedings of the 1997 IEEE Symposium on Information Visualization (InfoVis '97) (INFOVIS '97)*. IEEE Computer Society, Washington, DC, USA, 38–. http://dl.acm.org/citation.cfm?id=857188.857642

[29] Thomas S Tullis and Jacqueline N Stetson. 2004. A comparison of questionnaires for assessing website usability. In *Usability professional association conference*, Vol. 1.

[30] Hannes Högni Vilhjálmsson. 2004. Animating Conversation in Online Games. In *Entertainment Computing – ICEC 2004*.

[31] H. H. Vilhjalmsson. 2005. Augmenting Online Conversation through Automated Discourse Tagging. In *Proceedings of the 38th Annual Hawaii International Conference on System Sciences*.