

Simulating the Everyday Creativity of Readers

Brian O’Neill and Mark Riedl

School of Interactive Computing
Georgia Institute of Technology
{boneill, riedl}@cc.gatech.edu

Abstract

Sense-making is an act of everyday creativity. Research suggests that comprehending the world is an act of story construction. Story comprehension, the process of modeling the world of a fictional narrative, thus involves creative story construction ability. In this paper, we present an intelligent system that “reads” a story and incrementally builds a model of the story world. Based on psychological theories of story comprehension, our system computationally simulates the everyday creative process of a human reader with a combination of story generation search and strategies for inferring character goals. We describe the work in the context of a Synthetic Audience – a system that assists amateur story-writers by reading the story, analyzing the resultant sense-making models, and providing critique.

Introduction

Humans exhibit creativity in a vast range of domains such as music, art, dance, and storytelling. On a regular basis, humans exhibit creativity in ways that are overlooked: problem-solving, inference, and, in general, sense-making are all creative acts that humans carry out on a regular, and sometimes unconscious, basis. Sense-making is an act of human cognitive creativity: the construction of a narrative that explains what is happening around us (Bruner 1990; Gerrig 1994). We call this *everyday creativity*. Consider the following example:

It’s nighttime; Jesse is standing above Marlow, a gun to his head. The trigger slowly squeezes... The next morning, we see William, Marlow’s brother, digging a shallow grave. He drops a body into the hole in the ground. It’s Jesse’s lifeless body...

What happened that night? Who are these characters and what were their goals? These are questions that arise in the mind of the reader. A reader must effectively reconstruct the narrative, and infer concepts and events not explicitly read, causal relationships between events, and the goals and motivations of characters (Graesser et al 1994).

Boden (2009) suggests that not all creativity is high art. If this *everyday creativity* could be computationally harnessed what could a system do? Systems could employ human-analogous sense-making processes in order to build a model of the world – the real world or a fictional world observed in a book or movie. This model can be used to simulate human responses to stimuli. With respect to everyday creativity in story comprehension, we could “read” stories or “watch” movies and produce cognitive and affec-

tive responses equivalent to those of a human audience. Those responses can, in turn be used to provide feedback to amateur human creators that need assistance with storytelling ability by simulating the responses of a human reader/viewer receiving it for the first time. With that final concept in mind, this paper describes initial steps towards a “synthetic audience.” A synthetic audience aims to assist creators by modeling the cognitive processes of recipients of a creative artifact and providing feedback. Feedback from a synthetic audience could be given to the human creator faster and more frequently than feedback from another human source. The synthetic audience must have sufficiently robust ability in everyday creativity. For the purposes of the synthetic audience, we computationally model human creative sense-making processes in the context of story comprehension, focusing on the ability of a system to reconstruct a narrative from the events it reads.

Readers/viewers (we will use the media-agnostic term “audience”) are actively engaged in cognition when reading/viewing a narrative (Gerrig 1993). The audience engages in problem solving from the perspective of story world characters, attempts to resolve (intentionally or unintentionally placed) gaps in the narrative (called ellipses), and forecasts future events. The inference processes applied by the audience provide an explanation for what has been observed, but unlike conventional problem-solving, the results of these processes cannot be declared right or wrong. These inference processes can be exploited by authors to enable many of the more interesting cognitive phenomena of storytelling: ellipses, suspense, surprise, and genre expectations, among others.

In this paper, we present a component of the synthetic audience: a model builder that constructs a possible mental model for a human reader. The model builder “reads” the story as it is authored by the human creator. After each read event, it builds or revises its mental model by hypothesizing the goals of the story characters using a number of strategies, and reconstructs the story using a narrative generation planner. The model is then a source of feedback for the larger synthetic audience system. Different model structures are indicative of possible comprehension issues, such as changing character goals, diverging storylines, or unmotivated actions by the characters.

In the remainder of this paper, we discuss related work in the psychology of narrative comprehension, story understanding, story generation, and creativity support. We then describe our model builder, in the context of a synthetic audience. Finally, we will show that our model builder

produces a cognitively plausible mental model of the story-so-far, improving on approaches that do not model human creative processes.

Background

Reader Inference

While reading a story, readers continuously make inferences about aspects of the story that have not been explicitly stated in order to make sense of the narrative (Graesser et al. 1994). Some inferences can be made with little effort while reading, while other types of inference occur only when the audience has been given time to reason. The former group, described as *online inferences*, includes:

- Superordinate goals – Inferring the overarching goal motivating a particular character’s actions.
- Causal antecedents – Inferring the causal relationship between the current action and information that appeared previously in the text.

Conversely, *offline inferences*, those made when the audience is afforded time to reason, are as follows:

- Subordinate goals – Inferring the lesser goals or plan of action used to achieve the current event or state.
- Causal consequences – Inferring the effects of the current action.

In particular, the online processes, drive the creative search for a narrative explanation of observed events. That is, the inference of a character’s superordinate goal is a projection of that character’s actions into the future, resulting in the construction of narrative structure explaining *why* a character has performed observed actions. Likewise, inference of causal antecedents results in the construction of narrative structure that fills in the gaps between observed events in order to explain *how* particular events came to pass.

How does one represent a mental model of a story? Graesser and Franklin (1990) developed QUEST to model the human question-answering process as a theory of sense-making. QUEST was demonstrated in the context of story comprehension (Graesser et al. 1991). Stories are represented as directed graphs, where nodes represent events of the story, character goals, and world states. Edges represent relationships such as causality and the motivation or completion of goals. Traversing the arcs in a QUEST diagram allows one to answer questions, such as what enabled an event to come to pass, or why a character performed an action. Chains of causally-linked goals and events are called goal hierarchies, in which the last goal in the chain is the superordinate goal, the motivating goal for the entire sequence.

Figure 1 shows a QUEST structure for a story. Event nodes E_3 and E_4 , and goal nodes G_3 and G_4 make up a goal hierarchy. The superordinate goal, node G_4 , is *Initiated* (I) by the state node E_2 . That is, because of E_2 the hero has the indicated goal. Causality is shown in QUEST by *Consequence* (C) arcs between event nodes. In the example, E_4 occurred as a consequence of E_3 , and E_5 happened as a consequence of both E_3 and E_4 . Goal hierarchies are

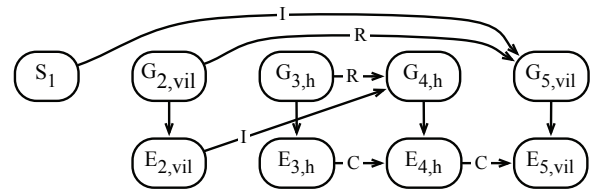


Figure 1. A QUEST structure for a story. Nodes represent states (S), character goals (G), and events (E). 1: Villain wants to be powerful. 2: Villain coerces Hero into agreeing to help. 3: Hero robs a bank. 4: Hero gives money to Villain. 5: Villain bribes the president.

formed by chains of *Reason* (R) arcs, indicating subordinate/superordinate relationships between goals. In the example, Goal G_3 is subordinate to Goal G_4 .

Related Work

Story Understanding. Sense-making is an example of everyday creativity, a process that humans use on a daily basis to explain the world around them. Sense-making in the context of stories shares many similarities to *story understanding*, a process by which a computational system extracts knowledge from a narrative text. Mueller (2002) summarizes many of the approaches taken in story understanding, including the application of known scripts and plans, the inclusion of plot units, and connectionist approaches. Other approaches include generating questions while reading and attempting to answer them with subsequent text (Ram 1994). In general, story understanding systems extract knowledge from complete texts, whereas we infer through abduction events and character goals in incomplete stories in order to assist amateurs.

Creativity Support. There are two general approaches to computational creativity support. The first are tools that assist creators by providing an appropriate interface for complicated creation processes without using AI (e.g., Skorupski et al. 2007). The second approach leverages AI to form a team made of the human creator and the computational tool. These mixed-initiative approaches (e.g., Si et al. 2008) lead to artifacts that have equal contributions from both the human and the AI. With the synthetic audience agent, we aim to leverage AI, as mixed-initiative tools do, without having explicit involvement in the creation of the product. In our approach, which we describe as “computer-as-audience” (Riedl and O’Neill 2009), the agent provides feedback to a human creator from the perspective of the recipient of a creative artifact.

Story Generation. Computational approaches to narrative-generation typically address the problem of creating content as either a search problem or an adaptation problem. See Gervás (2009) for a history of story generation research. Our system uses elements of both of these approaches, incorporating a case-based reasoner to infer character goals and a search-based story generator to construct narratives that explain the inferred goals.

We utilize the IPOCL narrative generation algorithm (Riedl and Young 2010), a refinement search approach to

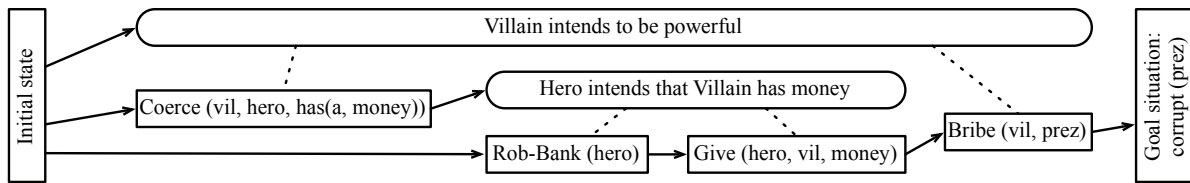


Figure 2. An IPOCL narrative plan corresponding to the QUEST structure from Figure 1.

constructing novel narrative structures that are both causally coherent and believable, as part of our system. IPOCL requires character actions to be justified both causally and by motivations and intentions. Specifically, it utilizes special data structures called frames of commitment to enforce the constraint that all events must be motivated by some preceding event. That is, every frame, representing a character goal, must be caused by some event as means of explaining why that character has the goal in question. As a refinement search process, IPOCL works backwards from a goal state, using causal and intentional requirements to guide the selection and instantiation of new events. Figure 2 shows the frames and events of an IPOCL plan, describing the same story as the QUEST diagram in Figure 1. Events are rectangles, while rounded boxes represent frames of commitment. Solid lines between events indicate a causal relationship. Dashed lines indicate that the event was carried out in service of that frame of commitment – as part of the character’s attempt at achieving its goal.

IPOCL shares similarities to the above psychological theories of narrative comprehension. IPOCL narrative plans can be converted to QUEST structures, and vice versa. Christian and Young (2004) present an algorithm for converting partial-order plans into QUEST structures. This algorithm has been updated to IPOCL plans (Riedl and Young 2010), specifically translating frames of commitment into goal hierarchies.

Synthetic Audience

The goal of a synthetic audience is to provide an amateur storyteller with feedback from the perspective of a recipient. That is, the synthetic audience “reads” the story as it is being written and produces cognitive and emotive responses based on theories of human story comprehension. A synthetic audience is able to provide feedback faster, and with greater frequency, than a human critic. In order to provide such feedback, it is necessary to model human responses to creative artifacts. When working with stories as they are being written, the system must derive a mental model of the story-in-progress based solely on what has been authored. The synthetic audience has to make inferences about events that are missing from the story, the causal relationships between events, and character goals. These inferences are comparable to human gap-filling and sense-making processes, both of which are carried out during reading comprehension. Thus, the synthetic audience system derives a mental model of the story as it is written, based on recognized human creative processes.

A storyteller using the synthetic audience authors states

– facts and descriptions – and events by selecting event templates from a list of options. These templates allow the author to fill in the specifics of the state or event, such as people, locations, or objectives. Authors can add states or events at any point in the story, regardless of chronology. The synthetic audience continually re-reads the story as it is authored, constructing a model from the audience’s perspective and uses that model to provide feedback. When there is feedback from the synthetic audience, it is displayed to the author in a non-intrusive manner; the author may respond to the feedback or ignore it.

Knowledge Representation

The synthetic audience requires knowledge about the semantics of events in order to make inferences. That is, the synthetic audience requires a domain theory – a description of how the story can change. We use a domain theory comprised of STRIPS-like event templates that provides information about preconditions and effects of any event the human authors. This is the same representation used by IPOCL. Because we employ a narrative planner, we also require every story to have a precondition. The user declares the facts of the initial state, as expository states where it makes sense to do so. If the initial state is incomplete, the Synthetic Audience uses a special operator, *Assert*, that causes facts to be inferred as true in the initial state, as a last resort to avoid failure. The use of *Assert* operator to modify the initial state is equivalent to the technique described by Riedl and Young (2006).

Model Builder Algorithm

The core of the synthetic audience is the Model Builder. The purpose of the Model Builder is to construct a possible mental model of a human audience, revising this mental model as each event is “read” in chronological order. The model is constructed by hypothesizing the goals of each character in the story, and constructing a narrative that explains what has been read. Once the story is read, the model is used to generate feedback in the form of critique.

The Model Builder starts by reading the newest event. If the newest event is not at the end of the story, the Model Builder rewinds the construction process to the latest point before the newly authored event and processes the remainder of the story as if encountered for the first time.

The search for superordinate character goals drives the Model Building process because of their importance in story comprehension and sense-making (Graesser et al. 1994). The Model Builder uses four strategies to hypothesize the superordinate goal for each character that is ac-

tively engaged in the current event. When characters are not actively engaged, previously inferred goals for those characters are retained from earlier iterations.

Character Goal Inference Strategies. The model builder uses the following four strategies, in the order given, to infer goals for characters actively engaged in the event.

- 1. Declared Goals (D).** The Declared Goals strategy hypothesizes goals that are explicitly declared in the new event for other characters. For example, if a character states its intention, then that goal is accepted at face value. Likewise, if a character instructs a subordinate character to do something, that goal is accepted for the latter character.
- 2. Existing Goals (E).** The Existing Goals strategy tracks goals that were hypothesized at an earlier point and remain unresolved based on the authored story. This strategy merely tries to place the new event into the hypothesized mental model from the previous iteration.
- 3. Proposed Goals (P).** The Proposed Goals strategy uses a case-based goal recognizer to infer character goals, based on that character's existing goal hierarchies. The recognizer is given a QUEST model containing only the acting character's goal hierarchy, contextual state nodes, and the most recently added event. The recognizer searches its case library for a QUEST model of a story with a similar chain of events as those in the given event and hierarchy. For the best match, the recognizer returns the goal at the top of the relevant goal hierarchy.
- 4. Top-of-Hierarchy (T).** The final strategy is the Top-of-Hierarchy strategy, which assumes that the most recently authored event is the goal of the characters involved. Top-of-Hierarchy is a last-resort strategy that is tantamount to "wait and see what happens next." The name of the strategy refers to the notion that the new event is the top of a QUEST goal hierarchy.

When more than one character is actively engaged in an event, the goal inference process is applied to each character, one at a time, in arbitrary order. The hypothesized goals and authored events are given to the IPOCL planner in order to test the goal hypothesis by generating a narrative sequence that explains the goals.

Testing the Hypothesis. Once the model builder has identified the goals of the characters, it tests the hypothesis by generating a narrative that links together all authored events to the hypothesized goals of the characters. This is achieved as follows. First, an instance of an IPOCL plan is created by instantiating every authored event in the QUEST model. Narratives generated during the prior iteration may have events that were generated but not written by the human author; these events are discarded. Temporal constraints enforce chronological ordering of authored events. The model builder constructs a goal situation consisting of newly hypothesized goals for the current character as well as un-realized goals for other characters carried over from prior iterations. Additionally, a frame of commitment is created for each un-realized hypothesized character goal across all characters.

A modified IPOCL planner is instructed to satisfy the preconditions for each event and each proposition of the goal situation, and to find a motivating event for each frame. As a refinement search algorithm, IPOCL takes a plan in any stage of completeness, finds a flaw – a reason why the plan is not complete – and resolves it. In the process, other flaws may be created, resulting in an iterative process. In this case, unresolved preconditions are solved by instantiating a new event, reusing an existing event, or by the initial world state. If IPOCL fails to find a plan, or if a plan is found that does not link all events in causal chains terminating with character goals, the current hypothesis is rejected and the model builder tries the next strategy.

We modified the IPOCL algorithm in the following ways. First, we bound the search depth to approximate cognitive limitations. Second, we provide a heuristic that strongly prefers to reuse authored events. Third, we add the *Assert* operator described above to declare unstated facts to be part of the initials state. Finally, we provide a special event, *Decide*, which has no preconditions and has the effect of giving a character an intention. *Decide* is equivalent to the system admitting that it does not know why a character has performed an action without failing the search. The Model Builder heuristic highly penalizes the inclusion of *Decide* events in the narrative, thus relegating its use to a last resort.

When the hypothesis is accepted, the plan generated by IPOCL is converted to a QUEST model using the previously mentioned IPOCL-to-QUEST algorithm.

Characters with Multiple Goals. It is possible for characters to pursue multiple goals throughout the course of a story. If goals hypothesized by the model builder using the P or T strategies are not accepted, then the Model Builder attempts to create a new goal hierarchy with any events that were not linked. The hypothesis is retested using the above technique.

Synthetic Audience Feedback

The synthetic audience generates feedback based on the QUEST model that resulted from hypothesis testing. Various QUEST structures indicate potential comprehension problems including:

- Diverging storylines – Indicated by disjoint causal chains.
- Unmotivated goals – Indicated by the need to use *Decide* events to construct missing Initiates arcs.
- Unexplained events or motivations – Indicated by the use of *Assert* operators to modify the initial state because missing information must be inferred.
- Sudden shifts in model – Indicated by a sudden change in goals hypothesized by the Model Builder from the reading of one event to the next.

If any potential mental model indicates possible reader comprehension issues, then this is feedback that we would aim to provide to the creator. The mental model constructed by the Model Builder is only one possible model. However, this single model remains useful in the context of a larger synthetic audience system, as it can be instanti-

ated with different domain theories and background knowledge to explore a variety of audiences.

Example

Consider the following scenario, selected to illustrate the goal inference strategies, in which a human author inputs a story with gaps. The story involves Aladdin, Jasmine, a genie, and a king. For purposes of illustration of the Model Builder, we will assume that the author has declared the characters up front, and provided various additional facts about the story world. The facts include: the genie is trapped in a magic lamp; a dragon possesses the lamp; and the king hates and fears the genie. The author writes the first few events:

1. The King *orders* Aladdin to retrieve the magic lamp.
2. Aladdin *travels* from the palace to the mountains.
3. Aladdin *gives* the magic lamp to the King.

The Synthetic Audience “reads” the events in order. The first event, *Order*, involves multiple characters. It arbitrarily chooses to attempt to infer Aladdin’s goals first. Using the Declared Goal (D) strategy, and based on the semantics of the order event, it hypothesizes that the orderee – Aladdin – will adopt the given goal: that the King should have the magic lamp. Next the Model Builder processes the King. Strategies D and E are not applicable as the event does not declare a goal for the King, nor is there a prior hypothesis about the King’s goal. Invoking the Proposed Goal (P) strategy, the case-based goal recognizer hypothesizes that the King’s goal could be to kill the Genie. This is because our case-base includes a sample story in which one character hires another to kill someone he hates. Hypothesis testing produces a plausible narrative in which Aladdin slays the dragon, takes the lamp, and gives it to the King who destroys it, thus killing the Genie. The resultant narrative is converted to a QUEST model and stored for later reference.

The Model Builder processes the second and third events involving Aladdin traveling to the mountains and then giving the lamp to the King. In both cases, the E strategy verifies that this is consistent with the previously hypothesized goal for Aladdin. The Model Builder infers that the dragon lives in the mountains. In each case, the King’s goal is retained from the first iteration because he is not an active character in either of the second or third events. Figure 3(a) shows the QUEST structure of the model constructed after the first three events.

Now suppose that the author adds one last event:

4. The King *commands* the Genie to make Jasmine love him.

The Model Builder arbitrarily decides to process the King first. The D strategy does not apply. The model builder attempts the (E) strategy, re-using the King’s goal of killing the Genie. However, it cannot find any plausible narrative in which the King commands the Genie as part of a goal hierarchy resulting in the death of the Genie. Therefore, the (E) strategy fails. The model builder again tries the (P) strategy. The case-based goal recognizer hypothe-

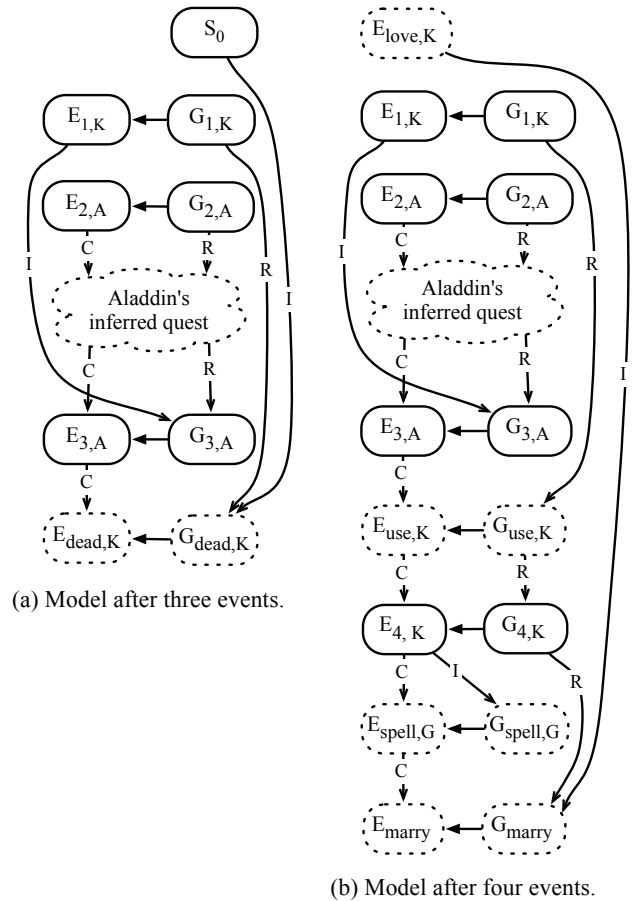


Figure 3. QUEST models of the example story. Numbers correspond to numbered events in the text. Nodes with dashed lines were inferred during narrative reconstruction.

sizes that the King’s goal could be to marry Jasmine, replacing the earlier hypothesized goal. The model builder processes the Genie’s involvement in event 4, and using the (D) strategy, determines that the Genie will adopt his given goal: to make Jasmine love the King. Hypothesis testing produces a narrative in which the King falls in love with Jasmine, sends Aladdin to retrieve the lamp. The Genie, under the influence of the King, casts a love spell on Jasmine. Finally, Jasmine and the King get married. Figure 3(b) shows the QUEST model for the new model.

Discussion

The Synthetic Audience is a cognitively plausible process for computationally constructing a model of the story-so-far. The system employs the everyday creativity of inference and future prediction. Graesser et al. (1994) are vague about the exact inference process used by human readers, proposing spreading activation; we assert that IPOCL, which reasons over representations that are analogous to QUEST structures, is a plausible substitution.

For any domain in which there are gaps (ellipses when referring to stories) in the events that can be observed, it is potentially non-trivial to create a well-formed sense-

making model in which the relationships between adjacent and non-adjacent events are found. This is true of observations of the noisy world around us and especially true of stories authored by amateur storywriters. Searching for the connections is a creative act because a complete narrative explanation is created as a by-product.

We believe that the approaches taken by the Model Builder are applicable in many domains, so long as the planner and case-based reasoner contains appropriate domain knowledge. Our approach works with stories that have (a) strong causal relationships (e.g., few non-sequiturs or random events), and (b) highly goal-driven character behavior – characters have a few top-level goals that are motivated by prior world events and not arbitrarily adopted. While not appropriate for all genres of story, these properties are common enough in popular mass-consumption stories and video games.

Is it enough just to employ a story planner or other search process to fill gaps? If one were to employ a story generator such as IPOCL after each event read, one would fill gaps between events; this is equivalent to the Model Builder's Top-of-Hierarchy strategy. However, such an approach would miss opportunities. First, any such model would not be representative of human models because the inference of superordinate character goals is one of the foremost online processes of an active reader. There are some events that are rarely ever superordinate goals, and thus rarely ever tops of goal hierarchies. Second, inference of superordinate character goals is a form of future prediction. By looking into the future and tracking back, one can often find connections between seemingly disparate causal chains; well-formed stories frequently tie plotlines together and human readers expect it. Of course, how closely the Model Builder matches human audience performance depends on the case library.

Third, and most significantly, the Model Builder constructs the sense-making model incrementally. One could wait until the story is complete, in which case a naïve gap-filler and the Model Builder would likely produce the same result. By reading the story one event at a time and building the model incrementally, the Synthetic Audience can trace changes in the model over the course of the story, thus providing feedback about surprises, suspense, and other cognitive and emotive effects on audiences that result from drastic revisions of the model. We conclude that the Model Builder's character goal inference strategies are critical. The D and P strategies provide superordinate goals that drive the creative explanation process. The E strategy provides continuity. The T strategy is a "catch all" when all the audience can do is wait and see.

The synthetic audience models human everyday creative processes. The synthetic audience reconstructs the narrative, making inferences about event causality and character goals, the same kinds of inferences that human readers make while reading a story. The synthetic audience performs incrementally, revising the model as the story is authored, rather than comprehending only complete stories, like typical story understanding systems. This model of

human everyday creative processes can be applied to recipients of creative artifacts, allowing feedback to be offered to creators from the audience perspective.

References

- Boden, M. A. 2009. Computer models of creativity. *AI Magazine* 30(3): 23–34. AAAI Press.
- Bruner, J. 1990. *Acts of Meaning*. Cambridge, MA: Harvard University Press.
- Christian, D., and Young, R. 2004. Comparing cognitive and computational models of narrative structure. In *Proc. of the 19th National Conference on Artificial Intelligence*.
- Gerrig, R. J. 1993. *Experiencing Narrative Worlds: On the Psychological Activities of Reading*. Yale University Press.
- Gerrig, R. J. 1994. Narrative thought?. *Personality and Social Psychology Bulletin* 20(6): 712–715.
- Gervás, P. 2009. Computational approaches to storytelling and creativity. *AI Magazine* 30(3): 49–62.
- Graesser, A.; Lang, K. L.; and Roberts R. M. 1991. Question answering in the context of stories. *Journal of Experimental Psychology: General* 120(3): 254–277.
- Graesser, A. C., and Franklin, S. P. 1990. QUEST: A cognitive model of question answering. *Discourse Processes* 13: 279–303.
- Graesser, A. C.; Singer, M.; and Trabasso, T. 1994. Constructing inferences during narrative text comprehension. *Psychological Review* 101(3): 371–395.
- Mueller, E.T. 2002. Story understanding. In *Encyclopedia of Cognitive Science*, London: MacMillan Reference.
- Ram, A. 1994. AQUA: Questions that drive the explanation process. In Schank, R.; Kass, A; and Riesbeck, C., eds., *Inside Case-Based Explanation*, Hillsdale, NJ: Lawrence Erlbaum Associates. 207–261.
- Riedl, M., and O'Neill, B. 2009. Computer as audience: A strategy for artificial intelligence support of human creativity. In *Proc. of the CHI 2009 Workshop on Computational Creativity Support*.
- Riedl, M.O. and Young, R.M. 2006. Story Planning as Exploratory Creativity: Techniques for Expanding the Narrative Search Space. *New Generation Computing*, 24(3).
- Riedl, M. O., and Young, R. M. 2010. Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research* 39: 217–267.
- Si, M., Marsella, S.C., and Riedl, M.O. 2008. Integrating story-centric and character-centric processes for authoring interactive drama. In *Proc. of the 4th Conference on Artificial Intelligence and Interactive Digital Entertainment*.
- Skorupski, J.; Jayapalan, L.; Marquez, S.; and Mateas, M. 2007. Wide ruled: A friendly interface to author-goal based story generation. In *Proc. of the 4th International Conference on Virtual Storytelling*.