Scheherazade: Crowd-Powered Interactive Narrative Generation

Boyang Li and Mark O. Riedl
School of Interactive Computing, Georgia Institute of Technology
{boyangli, riedl}@gatech.edu

Abstract

Interactive narrative is a form of storytelling in which users affect a dramatic storyline through actions by assuming the role of characters in a virtual world. This extended abstract outlines the SCHEHERAZADE-IF system, which uses crowdsourcing and artificial intelligence to automatically construct text-based interactive narrative experiences.

Introduction

Interactive narrative is a form of storytelling in which users affect a dramatic storyline through actions by assuming the role of characters in a virtual world. The simplest interactive narratives are Choose-Your-Own-Adventure books and hypertexts in which each plot point has branching options. More complex systems use artificial intelligence to determine available options for the user.

Prior AI-driven interactive narrative systems (see Riedl and Bulitko (2013) for a survey) rely on an *a priori* known domain model—a description of a fictional world, including characters, objects, places, and the actions that entities can perform to change the world. Once the domain model has been engineered, a story generation system can tell a potentially infinite number of stories involving these characters, places, and actions known to the system.

SCHEHERAZADE (Li et al. 2012; 2013) uses crowdsourcing to automatically learn the domain knowledge needed to construct construct and understanding stories about everyday activities such as going to a restaurant or going to a movie theater. The SCHEHERAZADE-IF system (Li, Lee-Urban, and Riedl 2012) creates interactive narrative experiences by allowing a human user to assume the role of one of the characters in the domain.

Scheherazade-IF

The SCHEHERAZADE-IF system attempts to create a novel, playable interactive fictions about simple, user-provided scenarios. For example, a human designer may request an interactive experience about a "bank robbery". If the system does not have a model of the domain, the system uses crowdsourcing to rapidly acquire a number of linear narrative examples about typical ways in which the topic might occur. In other words, the system collects human experiences and learns a generalized model—a *plot graph*—about the topic domain. Figure 1 shows the system architecture.

A plot graph is a representation in story generation systems that models the author-intended logical flow of events in the virtual world as a set of precedence constraints between plot events (Weyhrauch 1997; Nelson and Mateas 2005). In our work, a plot graph is a tuple $G = \langle E, P, M \rangle$ where E is the set of plot events, P is a set of temporal ordering constraints between events, and M is a set of unordered mutual exclusion relations that indicate which events can never co-occur in the same narrative experience. Mutual exclusion relations indicate *branches* where alternative narratives can unfold. See Figure 3 for a plot graph model of a bank robbery.

By learning plot graphs from the collected examples, our system can generate an interactive narrative about any topic for which a crowd of people can generally agree on the main events that should occur and the sequencing of the events.

Plot Graph Learning

The plot graph learning process is described in detail in Li et al. (2012) and Li et al. (2013) and summarized below.

The process begins with a user request for an interactive narrative on a particular topic. The system generates a query to Amazon's Mechanical Turk to solicit example narratives of the given topic, provided in natural language. To simplify the complexity of natural language processing, crowd workers are asked to segment their narratives such that each sentence contains one event. Crowd workers are instructed to use one verb per sentence and to avoid complexities such as conditionals, compound sentences, and pronouns.

Second, the system analyzes the simplified natural language narrative examples to discover the fundamental plot points on which people agree. Sentences from different narrative examples are clustered together according to semantic similarity to create plot events. The simplified language use allows clustering algorithms to discover plot events with relatively high accuracy.

Third, we identify the precedence constraints between plot events. Crowd workers produce noisy and sometimes er-

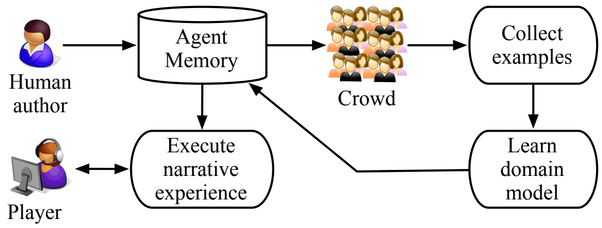

Figure 1: The SCHEHERAZADE-IF architecture.

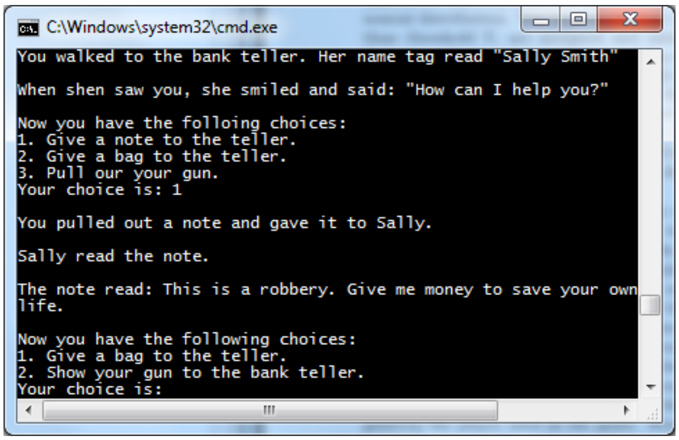

Figure 2: A screenshot of game play.

roneous answers such as omitting steps, requiring resilience against noise. For all pairs of plot events we select between the two orderings $e_1 \rightarrow e_2$ or $e_2 \rightarrow e_1$ based on statistical frequency (or neither if both orderings are equally likely or there is not enough data to make a conclusion). Precedence relations with statistically significant frequency and don't create cycles are recognized.

Fourth, we identify events that can never co-occur in a single narrative experience. We measure the mutual information between events to determine whether they belong to alternative, inconsistent procedures.

Interactive Execution

The plot graph ensures that players always experience valid stories on the topic, regardless of their choices in the interactive narrative. Once a plot graph has been constructed, the remaining task is to determine what options—the set of events that can happen next—are available to the player at any given time point.

At each step in the game, we compute the set of executable events. A plot event is executable when all of its direct, non-optional predecessors have been executed, except those parents excluded by mutual exclusion relations.

Executable events may contain options for the player to select from and also options for non-player characters (NPCs). Player options are presented to the player as a numbered list of actions to select from. When there are NPC options, SCHEHERAZADE-IF waits a predetermined period of time for the player to make a selection. If the player hasn't acted by the end of the period the system randomly executes an NPC option. This introduces race-condition and non-determinism to the game. See Figure 2 for an example of the text-based interface.

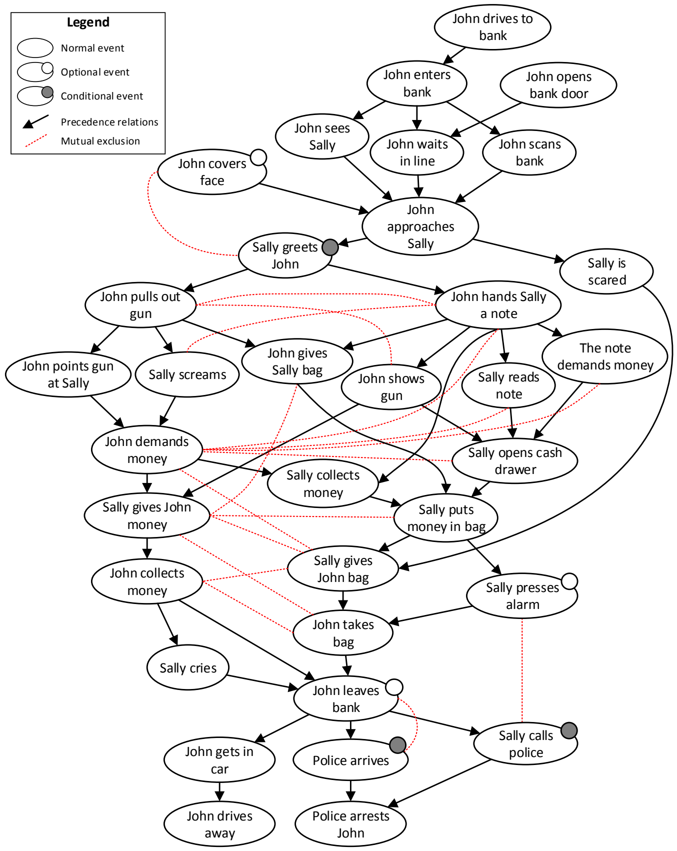

Figure 3: A plot graph of a bank robbery.

References

Li, B.; Lee-Urban, S.; Appling, D. S.; and Riedl, M. O. 2012. Crowdsourcing narrative intelligence. *Advances in Cognitive Systems* 2:25–42.

Li, B.; Lee-Urban, S.; Johnston, G.; and Riedl, M. O. 2013. Story generation with crowdsourced plot graphs. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*.

Li, B.; Lee-Urban, S.; and Riedl, M. O. 2012. Toward autonomous crowd-powered creation of interactive narratives. In *Proceedings of the 5th Workshop on Intelligent Narrative Technologies*.

Nelson, M., and Mateas, M. 2005. Search-based drama management in the interactive fiction Anchorhead. In *Proceedings of the 1st Conference on Artificial Intelligence and Interactive Digital Entertainment*.

Riedl, M. O., and Bulitko, V. 2013. Interactive narrative: An intelligent systems approach. *AI Magazine* 34(1):67–77.

Weyhrauch, P. 1997. *Guiding Interactive Fiction*. Ph.D. Dissertation, Carnegie Mellon University.