Capturing and Generating Social Behavior with The Restaurant Game

Jeff Orkin
MIT Media Laboratory
75 Amherst Street, E14-574M
Cambridge, MA 02139
+1 (617)253-1908
jorkin@media.mit.edu

Deb K. Roy
MIT Media Laboratory
75 Amherst Street, E14-574G
Cambridge, MA 02139
+1 (617)253-0596
dkroy@media.mit.edu

ABSTRACT

The Restaurant Game demonstrates an end-to-end system that captures and generates social behavior for virtual agents. Over 15,000 people have played *The Restaurant Game*, and we have developed a system to automatically learn patterns of interaction and dialogue from logs of their gameplay sessions. These patterns guide a case-based planning system, which generates behavior and dialogue for a virtual customer or waitress who can interact with a human, or with another agent. *The Restaurant Game* demonstrates a first step toward empowering non-programmers to realize socially intelligent characters for a wide range of applications.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search – *plan execution, formation, and generation.*

General Terms

Algorithms, Measurement, Design, Experimentation.

Keywords

Social simulation, Modeling natural language, Virtual Agents, Agents in games and virtual environments.

1. INTRODUCTION

The Restaurant Game is an online multiplayer game developed as a platform for both data collection and generation of human social interaction. Human players are anonymously paired online to dramatize the role of a customer or waitress in a 3D virtual environment. Players can type open-ended chat text to one another, and interact physically with objects in the environment. Over 15,000 people have played *The Restaurant Game*, and we have developed a system to automatically learn patterns of interaction and dialogue from logs of their gameplay sessions [4, 5]. These patterns guide a case-based planning system, which generates behavior and dialogue for a virtual customer or waitress who can interact with a human, or with another agent. A video demonstration is available online of a human customer interacting with a waitress agent, trained on 5,000 gameplay logs:

http://theRestaurantGame.net/aamas2010_demo.html

Cite as: The title of your paper should be written here, Author1, Author2 and Author3, *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2009)*, van der Hoek, Kaminka, Luck and Sen (eds.), May, 10–14, 2010, Toronto, Canada, pp. XXX-XXX.

Figure 1. Screenshot from *The Restaurant Game*.

Current approaches to implementing intelligent, interactive characters in the game industry are labor intensive, and require a high degree of technical skill and familiarity with artificial intelligence. Furthermore, simulating open-ended dialogue and capturing the fine-grained nuances of social interaction remain as difficult and largely unsolved problems. Meanwhile, an increasing number of people are playing games and socializing in virtual worlds online. We believe that mining interactions from humans online provides a promising solution to the behavior authoring bottleneck, and has the potential to enable new genres of games and social simulations. *The Restaurant Game* demonstrates a first step toward empowering non-programmers to realize socially intelligent characters for applications ranging from entertainment and training to customer service and social facilitation.mature graphics hardware, rendering engines, physics simulators, and path planners have leveled the playing field for near-photorealistic visuals in video games and simulations, artificial intelligence methods for social planning, interaction, and communication are poised to take the lead as the differentiating feature in games of the future. Though much progress has been made in navigation and action selection, natural language communication between agents remains a difficult problem, and communication between agents and humans even more so. Dynamic interactive dialogue poses numerous technical challenges, yet also holds the key to enabling entirely new genres of games, and broadening the reach of games beyond entertainment into new forms of social simulation.

2. BEHAVIOR CAPTURE & GENERATION

The Restaurant Game is built on the *Torque* game engine [2] and can be played on Windows or Mac OSX. Players control characters from a first-person perspective using the mouse, and can type open-end chat text with the keyboard. Everything players do is captured in text files on our servers as time-coded logs of actions, state changes, and utterances. Details of the data collection and behavior generation systems have been described in detail in previous publications [4, 5].

We have developed software in Java that parses the log files, and learns a lexicon of unique context-sensitive role-specific actions (e.g. waitress-picks-up-pie-from-counter) based on observed actions and state changes. Actions are stored in a STRIPS-inspired format [1], with lists of preconditions and effects. Actions are clustered automatically based on the learned affordances of their associated objects. For example, eating salmon and eating pie are clustered as eating *food,* where 'food' is a human label applied after-the-fact to an automatically learned concept. The clustered lexicon has over 7,000 unique actions. Similarly, we learn a phrase dictionary based on recurring patterns (between one and five words in length) found in the text of utterances.

Using the action lexicon and phrase dictionary, we compile each log file into a time-coded sequence of action lexicon indices, interspersed with utterance abstractions. Utterances are abstracted as unordered sets of phrases dictionary indices for all (possibly overlapping) phrases found within the utterance. This compiled form of the game logs is efficient for the case-based planning system [3] to compare at run-time, when it needs to retrieve a log file that matches well with the observed recent history.

The case-based planning system guides an agent, playing the role of a customer or waitress, as s/he imitates the actions and utterances taken by a human player in a particular gameplay session. A collection of *critics* monitors the interaction, and swaps the imitated log file when observed actions taken by the other player do not meet expectation set by the current log file. Critics can also detect when the agent's next intended action is impossible (due to the current state of the world), or statistically unlikely (based on n-gram models learned from the compiled logs). When critics force the agent to switch to a new log file, candidate logs are retrieved based on similarity to the recently observed history of physical actions.

Agents respond to natural language chat text input by compiling the text input into a set of phrase dictionary indices, and searching for similar candidate utterances in compiled log files. Consecutive utterance sequences from log files are preprocessed into *dialogue libraries*, indexed by the preceding physical action. Selecting an utterance response for an agent involves first selecting a dialogue library, based on the most recently observed physical action, and then searching the dialogue library for dialogues containing similar utterances to the input. From these dialogues, a response is selected based on the best matching utterance, favoring dialogues whose previous utterances also bear similarity to utterances observed recently in the real-time interaction.

3. FUTURE WORK

The current implementation of the interactive system allows a human to interact with an agent, or an agent to interact with another agent, and delivers a human-like social experience. However, the system makes a number of mistakes, due to its simplicity, which we are currently working to address. Currently, utterance responses are selected based solely on surface similarity to text input, and logs are compared based on similarity of low level action sequences. Making comparisons at such a low level, without any semantic information can lead the system to select the wrong response, or fail to find a suitable log file to imitate based on recent history. We are working toward semi-automated annotation solutions that will allow designers to associate semantic information with utterances, and group actions into higher-level event structures. These abstractions will allow development of more powerful critics, expected to produce more robust behavior and dialogue. Introducing annotation tools does mean that the process will no longer be fully automated. However, these tools will be designed to be accessible to non-programmers, and will only require annotating a small percentage of the corpus.

4. CONCLUSION

The Restaurant Game demonstrates an end-to-end system that captures and generates social behavior for virtual agents. Agents can interact and converse with humans, and interact with each other spontaneously through a decentralized system. While semantic annotation is required to make the system more robust, the current implementation illustrates the potential for generating behavior from data mined from human gameplay, and empowering non-programmers to populate games and simulations with agents capable of rich social interaction.

5. REFERENCES

[1] Richard E. Fikes and Nils J. Nilsson. STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2(3–4), 189–208, 1971.

[2] Garage Games. *Torque Game Engine* v1.5 (Windows and Universal Binary Mac OSX). www.garagegames.com.

[3] Kristian J. Hammond. CHEF: A model of case-based planning. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, Philadelphia, PA, 986.

[4] Jeff Orkin and Deb Roy. The Restaurant Game: Learning social behavior and language from thousands of players online. *Journal of Game Development*, 3(1), 39-60, December 2007.

[5] Jeff Orkin and Deb Roy. Automatic Learning and Generation of Social Behavior from Collective Human Gameplay. In *Proceedings of the 8th International Conference on Autonomous Agents and Multiagent Systems*, Budapest, Hungary, 2009.