Shared Task Proposal: Instruction Giving in Virtual Worlds

Alexander Koller, Johanna Moore,
Barbara di Eugenio, James Lester, Laura Stoia,
Donna Byron, Jon Oberlander, and Kristina Striegnitz

October 9, 2007

0.1 Introduction

This paper reports on the results of the working group "Virtual Environments" at the Workshop on Shared Tasks and Comparative Evaluation for NLG. This working group discussed the use of virtual environments as a platform for NLG evaluation, and more specifically of the generation of instructions in virtual environments as a shared task. It is based on the task proposal by [3], which a variety of workshop participants expressed interest in.

The use of virtual environments (VEs) as a platform for NLG evaluation addresses the need for cheap, human-based evaluation methodologies in NLG. Using VEs, it is possible to collect data from a human experimental subject that is physically in a different place than the NLG system. This means we can leverage a huge population of potential subjects, in a way similar to "web experiments" in psycholinguistics and psychology [7] or to systems that collect data by observing people playing games [9]. Many existing tasks, such as the generation of referring expressions, can be implemented in a VE framework; in addition, the framework can situate the human user in a simulated physical world, allowing us to study the effects of such a setting on NLG, with potential implications for human-robot interaction. Finally, the use of virtual worlds adds a "fun" factor to the scenario which we hope will attract attention, especially from students, to NLG.

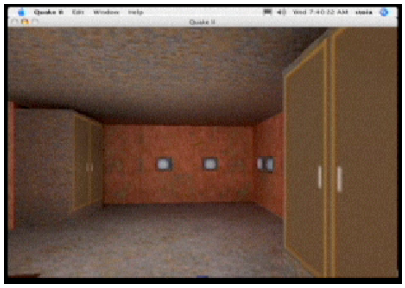
Rather than proposing a single shared task in this paper, we actually propose two different things:

1. a general "virtual environments" setting for NLG systems which can serve as a platform for many different shared tasks; and

2. a concrete shared task, in which the computer's job is to generate instructions for helping the human user solve puzzles in a virtual environment.

Moreover, we see the concrete task as scalable. We propose to start with a "baby steps" version of the task, which is perhaps less complicated than the final task but can be executed with comparatively little effort. We then propose to develop the task further based on the experiences of the first version, scale it up or down, and make it a recurring shared task in a couple of years. In doing so, we want to emphasize the collaborative rather than the competitive aspects of a shared task, and hope that the shared task would give rise to de facto standard modules for NLG.

The paper follows the standard structure for shared task proposals discussed at the workshop: We will first define the task and discuss how it can



(a)

(b)

Figure 1: Sample virtual environments: (a) the Quake 2 engine used in [8], (b) a disaster response scenario in Second Life [2].

be evaluated. Then we will explain what aims we hope to achieve with this task, and what subcommunities might find it interesting. Finally, we will describe our plan for carrying out the first round of the challenge. Wherever appropriate, we will distinguish between the general VE setting and the concrete instruction giving task.

0.2 Definition of the task

The object of the instruction giving task is to assist a human user in solving a problem in a virtual environment. The user controls a character in a simulated 3D space (see Fig. 1); they can move and turn freely, and manipulate and pick up objects in the world. Their goal is to solve a certain problem in the virtual world, e.g. to find an object and move it to a different location. The NLG system has access to complete information about the virtual world and to a plan for achieving the user's goal. The system's job is to generate instructions that assist the user in achieving this goal. At least in the first version of the task, the user will only be able to communicate back to the system by acting in the world and perhaps by pushing buttons on a GUI to signal that they didn't understand an instruction. This will simplify the task, compared to a full-blown dialogue system.

We envision a system architecture in which the NLG server, a central game server, and the graphical 3D client can all run on separate machines and are connected over the Internet (Fig. 2). In this architecture, the game server is responsible for keeping track of the state of the world and mediating the communication between the NLG server and the client, and perhaps for matchmaking, i.e. the pairing of users and NLG servers. The virtual world

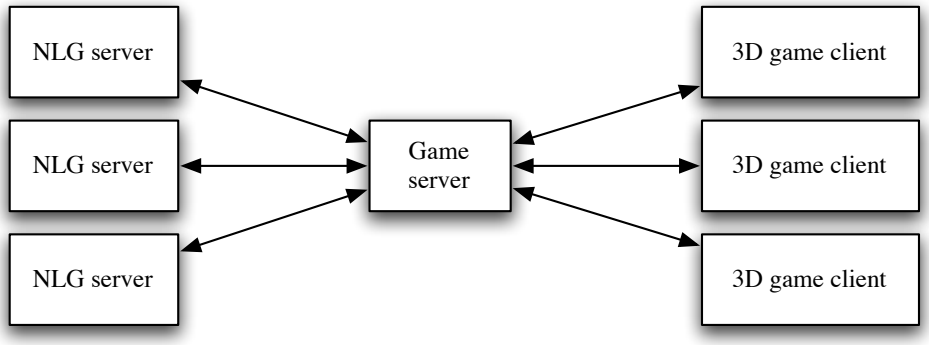


Figure 2: The system architecture. Note that no two subsystems need to run on the same machine.

itself can be defined by the task designer, using existing tools for designing maps for 3D computer games. Different 3D engines support different views of the scene; for example, Fig. 1a is a first-person view, whereas Fig. 1b uses a view over the avatar's shoulder. In the challenge, we will focus on a first-person view.

The NLG system is initialized with the properties of all objects in the virtual world. It is then notified every time the virtual world changes, e.g. in response to a user action. Furthermore, it receives periodic updates about the user's position and orientation, as well as about the objects in the world that the user can see. It can then decide for itself at which times it should take an action to communicate an instruction to the user, or to guide the user back into its plan, and send the instruction to the user at any time, to be displayed to the user as written text or spoken using a TTS system. The information that the system receives about the world is symbolic: All objects in the virtual world have names and properties (such as the object type, color, etc.) and three-dimensional positions. The task makes no assumptions about the linguistic formalisms or resources that the NLG system uses to generate the NL instructions.

In addition to instruction giving, virtual worlds can also be used for other concrete tasks. For instance, one could imagine an implementation of a referring expressions task in which the potential referents are all realized as objects in the virtual environment. The system could generate an RE, and the user's job would be to click on what they think is the intended referent. On the other hand, the instruction-giving task could also be scaled up in difficulty, extended to a dialogue task, or modified into a pure navigation

task (such as the Map Task [1]). Such tasks would still benefit from the network-based architecture.

0.3 Evaluation

One of the main strengths of the proposed task is that it can be evaluated very well. The central game server can automatically determine the task completion rate of an NLG system and the typical task completion times. In addition, because it is informed about every single mouse click of the user, it can also determine the rate of REs generated by the NLG system that were correctly resolved by the users. All these data can be collected without requiring any user intervention beyond their playing the game. The system can also collect subjective data via questionnaires presented to the user after each game round. These subjective and objective criteria could then be analyzed using a PARADISE-style framework [10].

Technically, all NLG systems participating in the shared task could be evaluated simultaneously. Each participating research group would run their system on a server at their own institution, and register it with the central game server provided by the task organizers. The game server would then accept connections from game clients (running on the machines of each experimental subject) and connect each client to a random NLG server; this run of the client would then count towards the evaluation data for this NLG system. After a certain period of time, the central game server would be stopped and the collected data aggregated and compared.

If the user is made to interact with the virtual world in a lab environment rather than over the Internet, it is also possible to collect further data through eyetracking studies. This sacrifices the size of the subject pool in favor of a more controlled experiment that allows us to collect more detailed data. Such a study of users instructed by avatars in a virtual environment is currently being piloted in Edinburgh [5].

0.4 Why this task is interesting

The primary aim of the proposed scenario is to provide a new framework for evaluating NLG systems. By making it possible to collect experimental data over the Internet, we tap into a huge pool of potential experimental subjects: For instance, the ESP game [9] has collected over 10 million labels for online images in the past three years, and the MIT Restaurant Game [6], which received far less media attention and requires users to download and install

a client to their own computers, still ran about 5,600 games, with an average length of ten minutes, within its first half year. This means that different systems, and different versions of the same system, can be compared in the context of a task-based human evaluation. This has advantages both over (expensive) evaluations using paid subjects, and over gold-standard based comparisons, which are problematic for NLG. These advantages apply to any task that can be evaluated in the virtual environments setting.

In addition, the instruction-giving task in virtual worlds emphasizes the role of generating referring expressions in a situated setting, and thus opens up new research perspectives. This is a very different problem than the classical non-situated Dale & Reiter style RE generation task: For example, experiments have shown that human instruction givers make the instruction follower move to a different location in order to use a simpler RE [8]. The task also involves such issues as aggregation and the generation of discourse cues and prosody. Overall, the virtual world setting can improve our understanding of situated communication – with potential applications to human-robot interaction, but without the need to deal with the difficulties of real robots, such as image recognition or navigation.

Because the virtual environments scenario is so open-ended, it – and specifically the instruction-giving task – can potentially be of interest to a wide range of NLG researchers. This is most obvious for research in sentence planning (GRE, aggregation, lexical choice) and realization (the real-time nature of the task imposes high demands on the system's efficiency). But as we have argued above, the task can also involve issues of prosody generation (i.e., research on text/concept-to-speech generation), discourse generation, and human-robot interaction. In addition, it touches upon a variety of neighboring research fields: In particular, the task constitutes a new application area for planning and plan recognition.

Furthermore, the virtual worlds setting could be relevant for researchers interested in dialogue systems. The instruction-giving NLG task can be extended to an instruction-giving dialogue task by allowing the user to talk back to the system, e.g. to ask clarification questions, making the virtual worlds scenario a platform for the evaluation of dialogue systems. The virtual worlds platform could also be used directly to connect two human users and observe their dialogue while solving a problem. Judicious variation of parameters (such as the familiarity of users or the visibility of an instruction giving avatar) would allow the construction of new dialogue corpora along such lines.

It is clear that no single system participating in the proposed shared task will involve ground-breaking progress in all of these areas. However, we be-

lieve that each research team could implement a simple baseline system with limited effort, and then improve those modules they find most interesting. We hope that the teams would then make their systems (or the modules into which they put the most research effort) available to the public. These systems could then be used by other teams in the next iteration of the shared task, which would lower the barrier to entry for new NLG researchers and could lead to the development of de facto standards for such modules in the long run.

0.5 Making it happen

0.5.1 Required resources

The most expensive resource that is required for the proposed shared task is the computing infrastructure for the network-based evaluation. It will be necessary to develop the central game server, the 3D game client running on the experimental subjects' machines, and an API or protocol for the NLG servers. Such components don't exist today in this exact form, but there is a wealth of open-source software that can be adapted and libraries that can be used to facilitate the development. For example, Byron's research group successfully adapted the Quake 2 game engine for their human-human experiments [4].

In addition, it will be necessary to develop virtual worlds and concrete tasks that the user needs to perform in these worlds. Again, there are open-source tools that support this, but of course substantial effort will be needed to define worlds that (a) people will want to actually play in, and (b) are challenging for the NLG systems we want to evaluate. One source of inspiration for the development of these worlds could be the Edinburgh Map Task [1]. In addition, experiments with human instruction givers, as started in [8], would contribute to an understanding of the NLG-relevant phenomena in this task.

Running the evaluation itself requires a game server that has a fast network connection and is capable of keeping track of multiple instances of the virtual world simultaneously. Finally, it will be necessary to make the experiment visible to potential experimental subjects, e.g. by posting about it in online gaming forums or listing it in a directory of psycholinguistic web experiments.

0.5.2 Plan of execution

The task of giving instructions in virtual worlds is, at this point, not yet sufficiently well-defined and the research challenges involved in it not yet sufficiently well-understood to be used as a shared task. This is why we propose to proceed in two steps, as follows.

In a first step, we propose to publicize the instruction-giving task as a challenge for teams of students. We will implement the necessary software infrastructure and some sample worlds and tasks, as well as a clear API for NLG systems. We hope to complete this step around Spring 2008. We will then publish a call for participation to student teams anywhere (which will hopefully be supported by the readers of this document), and run a first evaluation using the students' submissions late in 2008. We believe that it is feasible for a (reasonably well supervised) student team to come up with a system that can participate in the challenge within a few months, although such a system will typically not have a very high task completion or user satisfaction rate. As a side effect, we believe that the challenge, with its 3D and game-playing aspects, would attract smart students to spend time on NLG.

We will then organize a workshop to present the students' systems, compare notes, learn from the experiences in this first round, and refine the task definition into a concrete shared task to be organized in 2009. This first "real" instance of the shared task would then also be an opportunity to iron out bugs in the software infrastructure and come up with improved, more interesting, or more challenging virtual worlds and tasks. From this point on, we could then organize the shared task annually or every other year. In doing so, we will emphasize the non-competitive character of the challenge, and review our experiences from each year's challenge to make sure we are still working towards interesting research goals, rather than pursuing a local maximum, and modify or extend the shared task as needed.

0.6 Conclusion

In this document, we have presented our proposal for a shared task of generating instructions in a virtual world. This proposal has two aspects: It is simultaneously a concrete shared task proposal and a proposal for a novel framework for evaluating NLG systems.

After an initial preparation phase in which we will develop the software infrastructure necessary for carrying out this task, we will first carry out a simple version of the proposed task, targeted at student teams. We will

then evaluate our experiences from this step and use them to define a more advanced version of the shared task, which we will publicize as an actual research challenge in 2009.

One interesting topic to explore will be the relationship between the shared task we propose and the GRE shared task. Our task properly subsumes the GRE task: As a tiny special case, we can position the user in front of a number of possible referents and then generate a RE without allowing the user to move. Thus our system could be used as an internet-based evaluation platform for the GRE task, but whether this is reasonable or overkill remains to be seen.

Bibliography

[1] A. Anderson, M. Bader, E. Bard, E. Boyle, G. M. Doherty, S. Garrod, S. Isard, J. Kowtko, J. McAllister, J. Miller, C. Sotillo, H. S. Thompson, and R. Weinert. The HCRC Map Task corpus. *Language and Speech*, 34:351–366, 1991.

[2] Idaho Bioterrorism Awareness and Preparedness Program. Play2train website. `http://play2train.hopto.org/`.

[3] Donna Byron, Alexander Koller, Jon Oberlander, Laura Stoia, and Kristina Striegnitz. Generating instructions in virtual environments (GIVE): A challenge and an evaluation testbed for NLG. In Robert Dale and Mike White, editors, *Workshop for Shared Tasks and Comparative Evaluation in NLG*, Arlington, VA, 2007.

[4] Donna K. Byron. The OSU Quake 2004 corpus of two-party situated problem-solving dialogs. Technical Report OSU-CISRC-805-TR57, The Ohio State University Computer Science and Engineering Department, 2005. `ftp://ftp.cse.ohio-state.edu/pub/tech-report/2005/TR57.pdf`.

[5] Sara Dalzel-Job. A comparison of eye tracking and self-report measures of engagement with an eca. Master's thesis, University of Edinburgh, 2007.

[6] Jeff Orkin. Learning plan networks in conversational video games. Master's thesis, Massachusetts Institute of Technology, 2007.

[7] Ulf-Dietrich Reips. Standards for Internet-based experimenting. *Experimental Psychology*, 49(4):243–256, 2002.

[8] L. Stoia, D. Byron, D. Shockley, and E. Fosler-Lussier. Sentence planning for realtime navigational instruction. In *Companion Volume to*

Proceedings of HLT-NAACL 2006, pages 157–160, New York City, USA, 2006. Association for Computational Linguistics.

[9] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the ACM CHI Conference*, 2004.

[10] Marilyn Walker, Diane Litman, Candace Kamm, and Alicia Abella. PARADISE: A framework for evaluating spoken dialogue agents. In *Proceedings of the 35th ACL*, 1997.