Generating Instructions in Virtual Environments (GIVE): A Challenge and an Evaluation Testbed for NLG

Donna Byron[⋆] **Alexander Koller**[†] **Jon Oberlander**[‡] **Laura Stoia**[⋆] **Kristina Striegnitz**[◦]
[⋆] The Ohio State University [†] Columbia University [‡] University of Edinburgh [◦] Northwestern University
{dbyron|stoia}@cse.ohio-state.edu koller@cs.columbia.edu jon@inf.ed.ac.uk kris@northwestern.edu

Would it be helpful or detrimental for the field of NLG to have a generally accepted competition? Competitions have definitely advanced the state of the art in some fields of NLP, but the benefits sometimes come at the price of over-competitiveness, and there is a danger of overfitting systems to the concrete evaluation metrics. Moreover, it has been argued that there are intrinsic difficulties in NLG that make it harder to evaluate than other NLP tasks (Scott and Moore, 2006).

We agree that NLG is too diverse for a single "competition", and there are no mutually accepted evaluation metrics. Instead, we suggest that all the positive aspects, and only a few of the negative ones, can be achieved by putting forth a *challenge* to the community. Research teams would implement systems that address various aspects of the challenge. These systems would then be evaluated regularly, and the results compared at a workshop. There would be no "winner" in the sense of a competition; rather, the focus should be on learning what works and what doesn't, building upon the best ideas, and perhaps reusing the best modules for next year's round. As a side effect, the exercise should result in a growing body of shareable tools and modules.

The Challenge The challenge we would like to put forth is instruction giving in a virtual environment (GIVE). In this scenario, a human user must solve a task in a simulated 3D space (Fig. 1). The generation module's job is to guide the human player, using natural language instructions. Only the human user can effect any changes in the world, by moving around, manipulating objects, etc.

We envision a system architecture in which a central game server keeps track of the state of the world. The user connects to this server using a graphical client, and the generation system also connects to the server. Thus the implementation details of the virtual world are hidden from the generation system, which gets access to a symbolic representation of the world and a description of the task goal, and receives regular updates on the user's position, objects in his field of vision and their properties, etc. A sequence of actions that will achieve the goal is provided by an off-the-shelf planner.

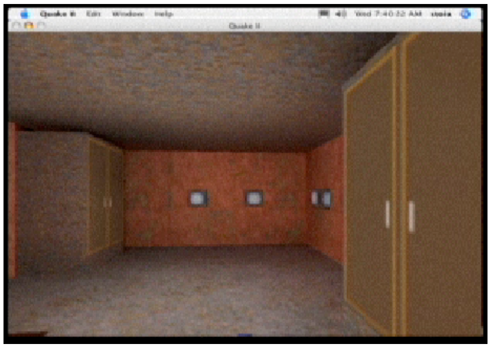

Figure 1: A sample virtual environment

There are numerous ways in which such a system could be evaluated. Quantitative measures can be collected automatically (completion time, success rate, percentage of generated referring expressions that the user resolved correctly), and subjective ones can be gathered from user satisfaction surveys. Since some 3D game engines, such as the open-source Quake II engine, support network play, it is technically possible to collect data cheaply from participants over the Internet.

Why this is a good challenge The proposed challenge spans a wide range of sub-problems of NLG, such as referring expression generation, aggregation, grounding, realization, and user modeling. On the other hand, the challenge can be scaled up and down along a number of different dimensions, both on the level of the challenge as a whole and on the level of individual systems. The output modality could be either text or speech; the system may or may not accept and process language input from the user; the user's position can be made discrete or even simplified to a text-adventure-like "room" concept (Koller et al., 2004); and the system might choose to present all instructions in one block and expect the user to follow them without any further intervention. Furthermore, most tasks require only a simple ontology and a limited vocabulary, and the challenge is completely theory-neutral in that it makes no assumptions about the representations that a system uses internally. All this means is that many NLG researchers could find something interesting in the challenge, and even small research teams could participate, focusing on one module and implementing all others with simple template-based systems.

We are aware that generalized instruction-giving is beyond the capabilities of the current state of the art. That's what makes it a challenge. Comparable events, such as the Textual Entailment challenge (Dagan et al., 2005), have been very successful in revitalizing a research field and attracting outside interest. Furthermore, like the highly successful Robocup challenge and its more resource-light variants, GIVE has the benefit of addressing hard research issues in the context of a "fun" game-based scenario. Such scenarios can bring visibility to a field and encourage the entry of young researchers.

Finally, the GIVE challenge has the potential to lead to the development of practically relevant technologies. It is closely related to the problem of pedestrian navigation assistance (termed the "Black Hawk Down problem" in military circles; Losiewicz, p.c.), object manipulation tasks (the "Apollo 13" or "Baufix" problem), and training systems (Rickel and Johnson, 1998). On a more theoretical level, the GIVE problem has already been found to shed new light on standard NLG tasks. For example, Stoia et al. (2006) observed that human instruction givers avoid the generation of complex referring expressions; instead, they guide the user into a position where a simple RE is available.

Logistics Assuming that we decided to organize such a challenge, we would provide the computational infrastructure. We would distribute a software package to interested participants, including the 3D engine (perhaps based on the modified version of Quake created by Byron's research group), a framework for the generation system servers, a planner, and example maps.

During the challenge itself, the participating research teams would run their generation servers on machines at their own institutions. These would communicate with the central game server we provide. Experimental subjects would be made available by the challenge organizers. While we hope to be able to let subjects interact with the systems online, such a setup makes it difficult to ensure that the sample of subjects is representative. Thus we would probably run a dual evaluation for the first challenge, at which we have both online and controlled subjects, to verify the comparability of the results.

Finally, we would communicate the evaluation results to the participants and invite them to present system descriptions at a workshop. This would also serve as a forum for participants to evaluate the challenge, modify it for the future, and identify interesting subchallenges. To encourage cooperation and ensure a benefit for the community as a whole, we are considering to require participants to make their code available to the public. However, we recognize that this suggestion may discourage some from participating and needs to be discussed within the NLG community along with the other details of how to implement the proposed GIVE challenge.

References

I. Dagan, O. Glickman, and B. Magnini. 2005. The PASCAL recognising textual entailment challenge. In *Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment*.

A. Koller, R. Debusmann, M. Gabsdil, and K. Striegnitz. 2004. Put my galakmid coin into the dispenser and kick it: Computational linguistics and theorem proving in a computer game. *Journal of Logic, Language, and Information*, 13(2):187–206.

J. Rickel and W. L. Johnson. 1998. Steve: A pedagogical agent for virtual reality. In *Proceedings of the Second International Conference on Autonomous Agents*.

D. Scott and J. Moore. 2006. An NLG evaluation competition? Eight reasons to be cautious. Technical Report 2006/09, Department of Computing, The Open University. `http://mcs.open.ac.uk/ds5473/publications/TR2006_09.pdf`.

L. Stoia, D. Byron, D. Shockley, and E. Fosler-Lussier. 2006. Sentence planning for realtime navigational instruction. In *Companion Volume to Proceedings of HLT-NAACL 2006*, pages 157–160.