A Machine-Learned Framework for Automatic Content Generation, Evaluation, and Critique

Adam James Summerville
University of California, Santa Cruz
1156 High Street
Santa Cruz, CA 95066
(+1) 831-459-0111,
asummerv@ucsc.edu

ABSTRACT

The majority of procedural level research has relied on human authored rules and heuristics. I intend to develop an end-to-end system capable of examining play, understanding levels and generating new content of similar style and playability, and finally offering analysis and critique of levels. The current roadmap utilizes computer vision, causal modeling, and neural network systems. The system should allow a human to step in at any point and make whatever changes they wish and get all downstream benefits.

INTRODUCTION

The procedural generation of game content has existed since the late 1970s, but has only become a research topic over the last decade. The vast majority of work, both academic and from practice has relied on human authored content (*Spelunky*) , heuristics (Sorenson and Pasquier n.d.), and rules (G. Smith et al. 2010),(A. M. Smith et al. n.d.). Humans are notoriously bad at introspection, either failing to capture why they performed a certain action or misattributing the reasoning, especially in creative domains (Nisbett and Wilson 1977). Given this, it would seem that trying to proceduralize design is going to be fraught with challenges, at least for a lay designer. While careful reasoning and analysis can circumvent the introspection problem it is a difficult task for a designer to sufficiently formalize their work and might require a different set of skills (e.g. understanding of constraint solving or logic programming).

However, we know (and if we do not then we have larger issues) that humans are capable of creative acts even if the process is difficult, or even impossible for humans to understand. I believe that a better way forward for procedural content generation is to learn how to create artifacts from the existing artifacts of these creative processes. This has multiple benefits:

- A lay designer need only curate positive and negative examples as opposed to trying to formalize their design constraints, choices, and intents
- The ability for cross-domain generalization opens new avenues of creativity that would be lost with islands of hand-authored design choices
- Incorporating playtraces into generation circumventing post-hoc player modeling
- Invert the process to supply critique of existing levels by highlighting sections that are possibly either too bland (high likelihood) or incoherent (low likelihood).
- Given a level, probabilistically generate player paths for designer consumption

METHOD AND CURRENT STATUS

My thesis work is intended to go from end-to-end and consists of 3 major pieces:

Proceedings of 1st International Joint Conference of DiGRA and FDG

1. Take in existing 2D game levels, from video, and learn the properties of the entities in the game. From having learned these properties, annotate the levels for things such as enemy, powerup, tile placement, and player path through the level.
2. From the annotated game levels train a machine learning system to generate levels that match the style of the input levels. As part of this generation also learn player paths through the level.
3. Given a level, or partial piece of a level, make suggestions of edits to the level, provide analysis of the level, and show likely player paths through the level.

My work so far has been focused on (2) with promising results in both the *Legend of Zelda* and *Super Mario Bros.* domains (Summerville et al. 2015) (Summerville and Mateas 2016). This is not to say that (2) is completed, as there still remains a lot of work to be done to generalize across games, and to better handle games with complex level topologies. I have performed some work on (1), mostly focused on the annotation task, but not starting from video; results are promising but require additional work to reach human annotation quality. I also intend to leverage work from the field of computer vision to be able to go directly from video. I have used (3) as an evaluation method for my generation work. The learned generators are just probability distributions that we sample and as such can provide likelihoods, but this is just initial work and requires further investigation. I am also currently working on providing player path analysis from video playtraces gathered from YouTube.

FUTURE WORK

Beyond completing the aforementioned work for my PhD, I am generally interested in applying machine learning techniques to ease and speed up the design process. I want to lower the barrier to make games and to essentially provide a QA department/design partner in a box to those would typically be unable to afford such things (casual and indie developers). On the technical side, neural machine learning has gone from using scalar values, to images, to sequences as first class objects, but I believe that graphs still remain a crucial missing piece and hope to develop methods better able to handle graphs.

BIBLIOGRAPHY

Mossmouth. 2008. *Spelunky.* [PC] Mossmouth, Pasadena, USA.

Nisbett, Richard E., and Timothy D. Wilson. 1977. "Telling more than we can know: Verbal reports on mental processes." *Psychological Review.*

Smith, Adam M, Eric Butler, and Zoran Popović. n.d. "Quantifying over Play: Constraining Undesirable Solutions in Puzzle Design." *FDG 2013.*

Smith, Gillian, Jim Whitehead, and Michael Mateas. 2010. "Tanagra: a mixed-initiative level design tool." In *FDG,* 209–216. ACM.

Sorenson, Nathan, and Philippe Pasquier. n.d. "Towards a Generic Framework for Automated Video Game Level Creation." EvoApplicatons'10.

Summerville, Adam, Morteza Behrooz, Michael Mateas, and Arnav Jhala. 2015. "The Learning of Zelda: Data-Driven Learning of Level Topology." In *Proceedings of FDG'15.*

Summerville, Adam, and Michael Mateas. 2016. *Super Mario as a String: Platformer Level Generation Via LSTMs.* eprint: `arXiv:1603.00930`.