Teaching Artificial Agents to Understand Language by Modelling Reward*

Edward Grefenstette†
DeepMind and UCL
etg@google.com

ABSTRACT

Recent progress in Deep Reinforcement Learning has shown that agents can be taught complex behaviour and solve difficult tasks, such as playing video games from pixel observations, or mastering the game of Go without observing human games, with relatively little prior information. Building on these successes, researchers such as Hermann and colleagues have sought to apply these methods to teach—in simulation—agents to complete a variety of tasks specified by combinatorially rich instruction languages. In this talk, we discuss some of these highlights and some of the limitations which inhibit scalability of such approaches to more complex instruction languages (including natural language). Following this, we introduce a new approach, inspired by recent work in adversarial reward modelling, which constitutes a first step towards scaling instruction-conditional agent training to "real world" language, unlocking the possibility of applying these techniques within a wide range of industrial applications.

KEYWORDS

Reinforcement Learning; Reward Modelling; Grounded Language

ACM Reference Format:
Edward Grefenstette. 2018. Teaching Artificial Agents to Understand Language by Modelling Reward. In *The 27th ACM International Conference on Information and Knowledge Management (CIKM '18), October 22–26, 2018, Torino, Italy*. ACM, New York, NY, USA, **??** pages. https://doi.org/10.1145/3269206.3272922

TALK SUMMARY

Developing agents that can learn to follow user instructions pertaining to an environment is a longstanding goal of AI research [**?**]. This challenge is complicated by the large degree of vagueness (under-specification) and ambiguity inherent in natural language. Recent work has shown deep reinforcement learning (RL) to be a promising paradigm for learning to follow language-like instructions in both 2D and 3D worlds (e.g. [**? ?**]). However, in each of these cases, a reward function instantiated in the environment is programmed to evaluate whether an instruction—such as *find the red tomato*—has been successfully executed. This approach is viable if the environment can unambiguously report whether a red tomato has been found. However, many language instructions in complex environments (or, indeed, the real world) could not plausibly be checked in this way. For instance, it is hard to imagine a hard-coded reward function for the everyday chores *fold the towels*, *arrange the flowers* or *set the table*, even though human users would have little problem judging whether these tasks had been carried out correctly.

In this talk, I discuss a step towards learning to execute a much wider class of underspecified and partially ambiguous instructions, initially presented in work done with colleagues [**?**]. We focus on the case of declarative commands that implicitly characterize a set of goal-states (e.g. "arrange the red blocks in a circle."). Given a dataset of instructions and a subset of the (multiple) viable goal-states for each instruction, provided by an expert, we jointly train a discriminator network and a policy network, which focus on the "what to do" and "how to do it" aspects of the tasks, respectively. The discriminator predicts whether a given state is the goal-state for an instruction or not. Meanwhile, the policy network maximizes the frequency with which it confuses the discriminator. We call our approach Adversarial Goal-Induced Learning from Examples (AGILE). AGILE is strongly inspired by Inverse Reinforcement Learning (IRL; **? ?**) methods in general, and Generative Adversarial Imitation Learning [**?**] in particular. However, it develops these methods to enable language learning; the policy and the discriminator are conditioned on an instruction, and that the training data contains goal-states—rather than complete trajectories.

This new method opens up new possibilities for training language-aware agents, because in the real world, and even in rich simulated environments [**? ?**], acquiring such data via human annotation would often be much more viable than defining and implementing reward functions programmatically. Indeed, programming rewards to teach robust and general instruction-following may ultimately be as challenging as writing a program to interpret language directly, an endeavour that is notoriously laborious [**?**], and some say, ultimately futile [**?**].

BIOGRAPHY

Edward Grefenstette is a Staff Research Scientist at DeepMind, and Honorary Associate Professor at UCL. He completed his DPhil (PhD) at the University of Oxford in 2013 under the supervision of

*This talk and the content of this abstract are substantially based on [**?**], which should be cited in lieu of the present document to reference this work appropriately.
†Work produced with co-authors Dzmitry Bahdanau, Felix Hill, Jan Leike, Edward Hughes, and Pushmeet Kohli. The accompanying talk references work completed by colleagues Karl Moritz Hermann, Simon Green, Fumin Wang, Ryan Faulkner, Hubert Soyer, David Szepesvari, Wojciech Marian Czarnecki, Max Jaderberg, Denis Teplyashin, Marcus Wainwright, Chris Apps, Demis Hassabis, and Phil Blunsom.

CIKM '18, October 22–26, 2018, Torino, Italy

ACM ISBN 978-1-4503-6014-2/18/10.
https://doi.org/10.1145/3269206.3272922

Profs Coecke and Pulman, and Dr Sadrzadeh, working on applying category-theoretic tools—initially developed to model quantum information flow—to model compositionality of distributed representations in natural language semantics. His recent research has covered topics at the intersection of deep learning and machine reasoning, addressing questions such as how neural networks can model or understand logic and mathematics, infer implicit or human-readable programs, or learn to understand instructions from simulation.