Evaluating Predictive Knowledge

Alex Kearney, Anna Koop, Craig Sherstan, Johannes Günther
Richard S. Sutton, Patrick M. Pilarski, Matthew E. Taylor
Department of Computing Science, University of Alberta, Edmonton, Alberta, Canada;
kearney@ualberta.ca

Abstract

Predictive Knowledge (PK) is a group of approaches to machine perception and knowledgability using large collections of predictions made online in real-time through interaction with the environment. Determining how well a collection of predictions captures the relevant dynamics of the environment remains an open challenge. In this paper, we introduce specifications for sensorimotor baselines and robustness-to-transfer metrics for evaluation of PK. We illustrate the use of these metrics by comparing variant architectures of General Value Function (GVF) networks.

Predictive Knowledge

A key challenge for machine intelligence is that of representation: a system's performance is tied to its ability to perceive and represent its environment. Predictive knowledge representations use large collections of predictions to model the environment. An agent continually anticipates its sensation from its environment by making many predictions about the dynamics of its environment with respect to its behaviour (Modayil, White, and Sutton 2014). These predictions about expected sensation can then be used to inform an agent's internal representation of its environment (Littman and Sutton 2002). Other proposals describe inter-relations of predictions, similar to TD Networks (Tanner and Sutton 2005; Makino and Takagi 2008) to enable abstract, conceptual representations by making predictions of predictions (Schapire and Rivest 1988).

In this paper we discuss the subtleties of evaluation predictive representation and propose two complimentary techniques. We specifically consider PK methods that 1) are able to expand their representations by proposing new predictions, 2) are able to self-verify their predictions through interaction with their environment, and 3) are able continually learn their predictions on-line.

To examine these evaluation metrics we use the General Value Function framework for predictive representations (White 2015). GVFs estimate the expected discounted return of a signal C defined as $G_t = \sum_{k=0}^{\infty}(\prod_{j=1}^{k}(\gamma_{t+j}))C_{t+k+1}$. Value is estimated with respect to a specific policy π, discount function γ, and cumulant c: $v(s;\pi,\gamma,c) = \mathbb{E}_\pi[G_t|S_t = s]$.

The parameters c, π, and γ are the *question parameters* which specify what a GVF is about; the *answer parameters*—such as the step size α and eligibility decay λ—describe how a learning method learns to answer the GVF question. GVFs can be learnt online, incrementally through methods such as Temporal-difference (TD) learning (Sutton 1988). The representational power of a given GVF network depends not just on the quality of the answers, but also in the architecture of the network, as illustrated in Figure 1.

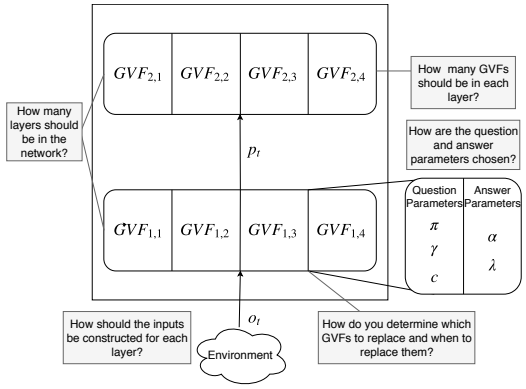

Figure 1: Many of the decisions which specify a PK architecture.

PK systems have been shown to be a scalable way to update and verify an agent's representation of the world, with examples of real-world robotic prediction tasks making thousands or tens of thousands of predictions in real-time on consumer-grade devices (Sutton et al. 2011; White 2015; Pilarski and Sherstan 2016).

Evaluating PK Architectures

Existing evaluation metrics for PK fall into two categories: 1) reporting the **average** error over all predictions within the PK system and 2) reporting errors on a known, challenging **subset** of the predictions within the system. Reporting the average error penalizes the accuracy of every prediction equally, when some predictions may have high error (such as for inherently random signals) but still provide representational power. Conversely, a representation that makes irrelevant but constant predictions will perform well according to average error, while providing no useful signals. Reporting errors on a subset of predictions requires identification of said subset across all architectures and lends itself to overfitting for those particular questions. It is difficult to iden-

tify predictions of interest without biasing towards particular architectures or network structures. Identifying predictions that require more complex representations in real-world settings requires extensive domain knowledge. In addition, it forces the inclusion of those pre-defined predictions when a goal of PK is to independently construct a useful representation. Neither of these are entirely satisfactory proxies for the real question: What is the representational power of a given PK system?

As a result of this evaluation bottleneck, examples of PK on real-world problems are largely proof-of-concept applications which serve to highlight the type, quantity, and diversity of predictions which can be made (Pilarski and Sherstan 2016; Modayil, White, and Sutton 2014; Sutton et al. 2011). Where evaluation exists, it focuses on prediction error as a means of evaluating the quality of a collection of predictions. This is insufficient, as *the reliability of predictions does not necessarily equate to the quality of a learned representation*. While low prediction error describes the quality of a single predictor, low average prediction error is not necessarily indicative of the best collection of predictions for constructing representations of the environment.

For example, one could maintain a diverse collection of GVFs for different time-scales γ and policies π that exclusively anticipate the voltage of servos on a robotic limb—a signal that is often constant. These trite predictions would likely have a lower error than a collection of predictions which represent the environment more completely. Moreover, comparing the average error between two collections of predictions with different question parameters is inappropriate, as the errors are with respect to different signals. When we compare the average error of different sets of predictions in PK architectures, we are unable to meaningfully quantify how changes in the architectural proposal impact the knowledgability of a system.

We propose sensorimotor predictions as a baseline which balances our ability to meaningfully assess the representational capacity of a collection of predictions in a meaningful way, while being general enough to be extensible to real-world prediction problems.

Evaluation by Sensorimotor Baselines

A scalable alternative to comparison by hand-crafted predictions is to maintain a collection of baseline sensorimotor predictions common between each architecture being evaluated. Instead of hand-crafting predictions based on the idiosyncrasies of a particular domain, a sensorimotor baseline uses the observations from the environment as prediction targets. The identification of good features is integral to being able to make reliable predictions; in evaluating the ability of a system to predict its raw stimuli, we are in fact evaluating the ability of the system to perform representation learning for the simplest predictions we could want to make.

By comparing architectures based on how well they can represent their stimuli, we are prioritizing architectures that are able to find better representations for learning low-level sensory input, rather than better representations of the environment in general. While a limitation, it is a natural approach to evaluation: approaches to PK have been motivated by being able to anticipate their environment (Modayil, White, and Sutton 2014), and low-level anticipatory predictions are useful as inputs in applications of PK (Sherstan, Modayil, and Pilarski 2015).

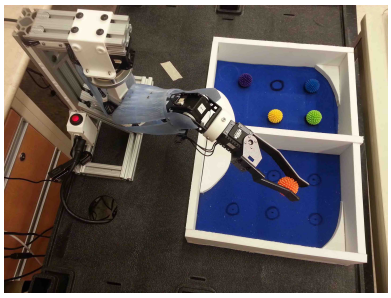

Figure 2: The data source for the experiments in this work: The Bento Arm, controlled by a human participant, generating a stream of multimodal sensory data from participants' interactions with a modified Box and Blocks task.

Sensorimotor baselines are a balance between the two aforementioned methods of evlauation: Baseline predictions enable us to assess the representation generated by our PK system with no designer intervention, making them a general scalable alternative for evaluation of real-world systems. By assessing representation quality, we can begin to precisely quantify the impact of different construction methods in real-world domains. Using sensorimotor baselines is a fair first step in bridging the evaluation gap between toy domains and real-world problems.

Evaluation by Transfer

Perhaps one of the most natural qualities of an effective PK system is generality. PK systems are intended for use in life-long, continual learning methods—methods that are expected to learn for the duration of their deployment. In such a setting it is imperative that the predictions being made are resilient to changes in their environment. A method of evaluating the ability of a continual learning system to produce general representations is through transfer-learning (Taylor and Stone 2009). We can evaluate the generality of PK by constructing GVFs in one setting and testing their generality on experience in a transfer environment that shares some traits with the source setting. An architecture that is able to propose and interrelate GVFs such that they are more robust to such transfers is an architecture that produces more general representations.

Experiment: Prosthetic Prediction Task

We explore sensorimotor baselines and transfer using data from a human control task on the Bento Arm (Dawson et al. 2014), an open-source robot arm intended for use as a research prosthesis. Human control of a robotic prosthesis is an area with active interest in PK (Pilarski and Sherstan 2016), and GVFs have been previously used to improve the control in this domain (Pilarski et al. 2013).

Data for this experiment was sourced from the previous experiments of (Edwards et al. 2016). Four users performed a common manipulation challenge where they used the robot arm to move objects over a barrier (Figure 2). Each user