Distributional Reinforcement Learning for Energy-Based Sequential Models

Tetiana Parshakova, Jean-Marc Andreoli, Marc Dymetman
tetianap@stanford.edu; {jean-marc.andreoli,marc.dymetman}@naverlabs.com

NAVER LABS EUROPE

Stanford | Institute for Computational & Mathematical Engineering

Motivation

Global Autoregressive Models (GAMs) are a recent proposal [1] for exploiting global properties of sequences for **sample-efficient** learning of seq2seq models. In the first phase of training, an **Energy-Based model (EBM)** over sequences is derived. This EBM has higher representational power than a standard autoregressive model, but is unnormalized and cannot be directly exploited for sampling. To address this issue [1] proposes a distillation technique, which can only be applied under certain conditions. By relating this problem to Policy Gradient techniques in RL, but in a **distributional** rather than **optimization** perspective, **we propose an inference technique applicable to any sequential EBM**, beyond the initial GAM motivation.

Background: GAMs

A **GAM** [1] is an unnormalized potential (aka EBM) over sequences x:

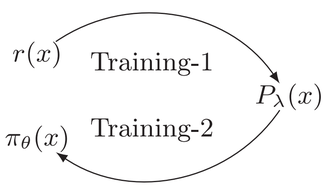
$$P_\eta(x|C) = r_{\eta_1}(x|C) \cdot e^{\langle \lambda_{\eta_2}(C),\ \phi(x;C)\rangle},\ \eta = \eta_1 \oplus \eta_2.$$

The factor $r_{\eta_1}(x|C)$ is an autoregressive model for generating x in the context C. The factor $e^{\langle \lambda_{\eta_2}(C),\ \phi(x;C)\rangle}$ is a log-linear potential. Normalized distribution: $p_\eta(x|C) = \frac{P_\eta(x|C)}{Z_\eta(C)}$, with $Z_\eta(C) = \sum_x P_\eta(x|C)$.
The features $\phi(x;C)$ provide prior knowledge to the model by drawing its attention to potentially useful global sequence properties that may be difficult for the autoregressive component to discover on its own.
Here, we focus on a simple unconditional (language modelling) version of GAMs:

$$P_\lambda(x) \doteq r(x) \cdot e^{\langle \lambda,\ \phi(x)\rangle} \quad (1)$$

where the autoregressive factor $r = r_{\eta_1}$ is first learnt on the training dataset of sequences D and then kept fixed, and where the parameter vector λ is then trained on top of r, also on D. In general, if P is an EBM, we denote by p the associated normalized distribution $p(x) = \frac{1}{Z}\ P(x)$, with $Z = \sum_x P(x)$. The normalized distribution associated with $P_\lambda(x)$ is denoted by $p_\lambda(x)$.

GAM training is done in two phases (see drawing). In [1], we applied a distillation technique for Training-2, but this supposed that we are able to sample from p_λ, a serious limitation. Here apply instead a distributional variant of RL for Training-2, which does not have this limitation, and is of general applicability to sampling with EBMs.



Distributional RL for EBMs

- **Standard RL Policy** Objective: $\max_\theta \mathbb{E}_{x\sim\pi_\theta(\cdot)}\ P(x)$
  (i.e. trying to find policy concentrated around **maximum reward**)
  $\Rightarrow$ SGD (PG): $\mathbb{E}_{x\sim\pi_\theta(\cdot)}\ P(x)\ \nabla_\theta \log \pi_{\theta(x)}$
  (Vanilla Policy Gradient - REINFORCE)

- **Distributional RL Policy**[a] Objective: $\max_\theta \mathbb{E}_{x\sim p(\cdot)} \log \pi_\theta(x)$
  (i.e. $\min_\theta\ CE(p, \pi_\theta)$: trying to find policy that **best approximates reward distribution**)
  $\Rightarrow$ SGD (Distillation): $\mathbb{E}_{x\sim p(\cdot)}\ \nabla_\theta \log \pi_\theta(x)$
  (What we did in [1]. Issue: we need to be able to sample from p !!!)
  $\Rightarrow$ SGD (DPG$_{\text{on}}$): $\mathbb{E}_{x\sim\pi_\theta(\cdot)}\ \frac{P(x)}{\pi_\theta(x)} \nabla_\theta \log \pi_\theta(x)$
  (A distributional variant of PG. We tried it, but unstable.)
  $\Rightarrow$ SGD (DPG$_{\text{off}}$): $\mathbb{E}_{x\sim q(\cdot)}\ \frac{P(x)}{q(x)} \nabla_\theta \log \pi_\theta(x)$
  (Importance sampling with proposal q. More stable than DPG$_{\text{on}}$.)

[a] For a different view of Distributional RL see [3].

References

[1] Tetiana Parshakova, Jean-Marc Andreoli, and Marc Dymetman. Global Autoregressive Models for Data-Efficient Sequence Learning (CoNLL 2019).

[2] Yann LeCun, Sumit Chopra, Raia Hadsell, Marc'Aurelio Ranzato, and Fu Jie Huang. A Tutorial on Energy-Based Learning. 2006.

[3] Bellemare, Dabney and Munos. A Distributional Perspective on Reinforcement Learning. 2017.

[4] Anonymous. Residual Energy-Based Models for Text Generation. Submitted to ICLR-2020.

Algorithm DPG$_{\text{off}}$

Algorithm 1 DPG$_{\text{off}}$

Input: P, initial policy q ▷ in GAMs: $P = P_\lambda$ and $\pi_{\theta_0} = r$

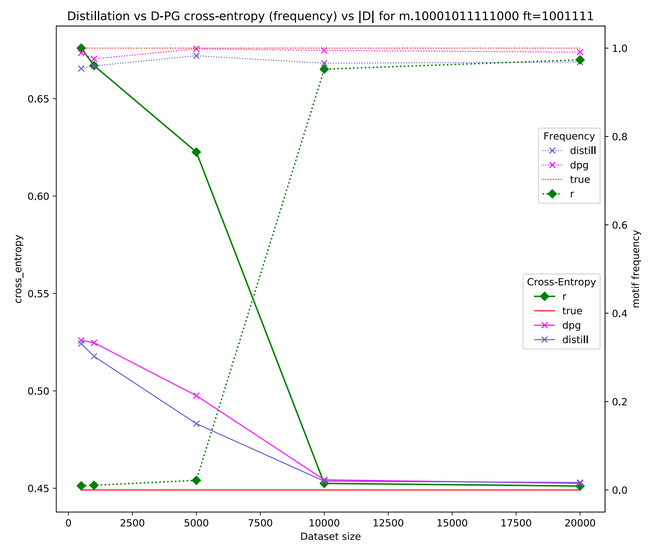
$\leftarrow q$
for each iteration **do**
  for each episode **do**
    sample x from $q(\cdot)$
    $\theta \leftarrow \theta + \alpha^{(\theta)} \frac{P(x)}{q(x)}\ \nabla_\theta \log \pi_\theta(x)$ ▷ $\alpha^{(\theta)}$: learning rate
  if π_θ is superior to q **then** ▷ in terms of validation perplexity
    $q \leftarrow \pi_\theta$

Output: π_θ

Case study: Motifs in Strings

true	101**10001011111000**1000001001001
r	011111000010111111000111000101 1
π_θ	111010**10001011111000**0111111100
features	$[m, \_, \_, d_0, d_1, d_2, d_3]$
λ's	$[-10.1, \_, \_, -0.15, -0.06, 0.0, -0.14]$
moments *true*	$[0.0, \_, \_, 0.47, 0.99, 1.0, 0.91]$
moments r	$[0.95, \_, \_, 0.53, 0.99, 1.0, 0.91]$
moments π_θ	$[0.0006, \_, \_, 0.43, 0.99, 0.99, 0.91]$
cross entropy (CE)	*true*: 0.45, r: 0.56, π_θ: 0.47
motif freqs	*true*: 1.0, r: 0.045, π_θ: 0.959

Distillation vs. DPG



- Here, D consists of random binary strings (of length 30), filtered by the condition that they contain a fixed motif 10001011111000
- Features: m (binary feature $m = 0$ means motif is present), d_0, d_1, d_2, d_3 are "distractor" binary features, with small correlation with presence of motif (0 means feature is present).
- We vary D and observe CE (i.e. perplexity) on test data (solid lines), and also motif frequency in samples from different models (dotted lines).
- We compare results with Training-2 done by Distillation vs. DPG$_{\text{off}}$.

Results

$\lvert D\rvert$	$\frac{\text{CE}(T,\pi_\theta^{dpg})}{\text{CE}(T,\pi_\theta^{dis})}$	$\frac{\text{mtf\_frq}(\pi_\theta^{dpg})}{\text{mtf\_frq}(\pi_\theta^{dis})}$	$\frac{\text{CE}(T,\pi_\theta^{dpg})}{\text{CE}(T,r)}$	$\frac{\text{CE}(T,\pi_\theta^{dpg})}{H(p_{true})}$	$\frac{\text{mtf\_frq}(\pi_\theta^{dpg})}{\text{mtf\_frq}(r)}$
500	1.008	1.252	0.76	1.18	281.51
1000	1.014	1.102	0.762	1.178	240.40
5000	1.019	1.21	0.865	1.059	34.73
10000	1.014	1.067	0.968	1.023	2.17
20000	1.004	1.023	1.0	1.006	1.03

Discussion

Several recent works [1,4] have been stressing the representational power and sample efficiency of Energy-Based Models in the context of sequence generation. One delicate aspect of these models is the difficulty of efficiently sampling from the EBM representation.

We have addressed this general problem here by connecting it to a distributional variant of RL. Looking back at the specific situation of GAMs, Training-2 can thus be viewed as a form of RL; it is interesting to note that Training-1, by contrast, might be seen as a form of *Inverse* RL: determining a reward (i.e. fitting an EBM) from the data itself, rather than by external prescription.

In terms of importing standard RL approaches to the problem of sampling from EBMs, our simple DPG$_{\text{off}}$ technique only scratches the surface, and further work could consider the adaptation of more sophisticated RL techniques (e.g. actor-critic) in this context.