A Recurrent Latent Variable Model for Sequential Data*

Junyoung Chung, Kyle Kastner, Laurent Dinh,
Kratarth Goel, Aaron Courville, Yoshua Bengio

2016

1 What

The authors including latent variables into RNN hidden state dynamics and empirically show that it might be beneficial via examples of speech modelling and handwriting generation.

The authors are interested in "highly structured" data which is characterised by two properties:

- high signal-to-noise ratio
- complex relationship between the underlying factors of variation and the observed data (e.g. in speech, the vocal characteristics of the speaker influence audio in a very complicated, but *consistent* manner.

The paper is not the first to integrate r.v. into RNN hidden states, but is the first to integrate the dependencies between the latent r.v. at neighbouring time steps.

2 Why

Learning generative models of sequences has been a challenge for a long time, and we're still working on it. Historically, dynamic Bayesian networks (DBNs), e.g. HMMs and Kalman filters, has dominated the solution space. Now, RNNs are running the show.

RNNs has two important parts: a transition function which determines the evolution of the hidden state, and a mapping from the state to the output. RNNs are powerful because they can have a rich internal state and model the transitions via non-linear functions. However, DBNs have something that RNN does not: randomness/variability in the hidden state: the transition function in

*Notes by Vitaly Kurin `https://yobibyte.github.io/`

RNNs is deterministic. This might lead to problems since because of that RNNs can't model complex dependencies using a unimodal distribution or a mixture of them.

3 How

What's RNN?

$$\mathbf{h}_t = f_\theta(\mathbf{x}_t, \mathbf{h}_{t-1}) \tag{1}$$

$$p(\mathbf{x}_t|\mathbf{x}_{<t}) = g_\tau(\mathbf{h}_{t-1}) \tag{2}$$

The latter can be decomposed into two parts. First, gets the parameters given the hidden state: $\phi_t = \varphi_\tau(\mathbf{h}_{t-1})$. Second, returns the density $p_{\phi_t}(\mathbf{x}_t|\mathbf{x}_{<t})$.

There is a really cool insight here about a potential issue with RNNs modelling variability. If our transition function is deterministic, p_{ϕ_t} is the only source of variability. Then there will be a compromise between "generation of a clean signal and encoding sufficient input variability to capture the high-level variability both within a single observed sequence and across data examples".

The paper makes use of VAEs which are trained by maximising ELBO:

$$\log p(\mathbf{x}) \geq -\mathrm{KL}(q(\mathbf{z} \mid \mathbf{x}) \| p(\mathbf{z})) + \mathbb{E}_{q(\mathbf{z}|\mathbf{x})}\left[\log p(\mathbf{x} \mid \mathbf{z})\right] \tag{3}$$

3.1 VRNN

3.1.1 Generation

To help VAE (VRNN has a VAE at every time step) to take into account the temporal structure, VAE is conditioned on the state $\mathbf{h}_{t-1}$.

Prior on latent vars is no longer standard Gaussian ($\boldsymbol{\mu}, \boldsymbol{\sigma}$ are conditional prior's params):

$$\mathbf{z}_t \sim \mathcal{N}(\boldsymbol{\mu}_{0,t}, \mathrm{diag}(\boldsymbol{\sigma}_{0,t}^2)) \text{ , where } [\boldsymbol{\mu}_{0,t}, \boldsymbol{\sigma}_{0,t}] = \varphi_\tau^{\mathrm{prior}}(\mathbf{h}_{t-1}) \tag{4}$$

$p(\mathbf{x}_t \mid \mathbf{z}_t)$ also depends on the state:

$$\mathbf{x}_t \mid \mathbf{z}_t \sim \mathcal{N}(\boldsymbol{\mu}_{x,t}, \mathrm{diag}(\boldsymbol{\sigma}_{x,t}^2)) \text{ , where } [\boldsymbol{\mu}_{x,t}, \boldsymbol{\sigma}_{x,t}] = \varphi_\tau^{\mathrm{dec}}(\varphi_\tau^{\mathbf{z}}(\mathbf{z}_t), \mathbf{h}_{t-1}) \tag{5}$$

The state is updated via:

$$\mathbf{h}_t = f_\theta\left(\varphi_\tau^{\mathbf{x}}(\mathbf{x}_t), \varphi_\tau^{\mathbf{z}}(\mathbf{z}_t), \mathbf{h}_{t-1}\right) \tag{6}$$

The generative model is factorised as:

$$p(\mathbf{x}_{\leq T}, \mathbf{z}_{\leq T}) = \prod_{t=1}^{T} p(\mathbf{x}_t \mid \mathbf{z}_{\leq t}, \mathbf{x}_{<t}) p(\mathbf{z}_t \mid \mathbf{x}_{<t}, \mathbf{z}_{<t}). \tag{7}$$

3.1.2 Inference

Approximate posterior also depends on $\mathbf{h}_{t-1}$ ($\boldsymbol{\mu}, \boldsymbol{\sigma}$ are q's params):

$$\mathbf{z}_t \mid \mathbf{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_{z,t}, \mathrm{diag}(\boldsymbol{\sigma}^2_{z,t})) \text{ , where } [\boldsymbol{\mu}_{z,t}, \boldsymbol{\sigma}_{z,t}] = \varphi^{\mathrm{enc}}_\tau(\varphi^{\mathbf{x}}_\tau(\mathbf{x}_t), \mathbf{h}_{t-1}), \tag{8}$$

$$q(\mathbf{z}_{\leq T} \mid \mathbf{x}_{\leq T}) = \prod_{t=1}^{T} q(\mathbf{z}_t \mid \mathbf{x}_{\leq t}, \mathbf{z}_{<t}) \tag{9}$$

3.1.3 Learning objective

$$\mathbb{E}_{q(\mathbf{z}_{\leq T} | \mathbf{x}_{\leq T})} \left[\sum_{t=1}^{T} \left(-\mathrm{KL}(q(\mathbf{z}_t \mid \mathbf{x}_{\leq t}, \mathbf{z}_{<t}) \| p(\mathbf{z}_t \mid \mathbf{x}_{<t}, \mathbf{z}_{<t})) + \log p(\mathbf{x}_t \mid \mathbf{z}_{\leq t}, \mathbf{x}_{<t}) \right) \right]. \tag{10}$$

4 Evaluation

I'm distant from the field, so, I have no idea about datasets etc. However, the authors evaluate the models on tons of datasets: Blizzard, TIMIT, Onomatopoeia and Accent, and this is quite impressive.

Again, evaluation of generative models is hard [Theis et al., 2015] and I will not touch it here. Apart from comparing log-likelihoods, the authors do latent space analysis: plotting deltas in the mean for different time steps and comparing them visually. They also plot KL between the approximate posterior and the conditional prior.

I really like the way authors do some suggestions after observing particular behaviour without claiming that something happens 100% sure because of this or that:

"*We suggest that the large amount of noise apparent in the waveforms from the RNN-GMM model is a consequence of the compromise these models must make between representing a clean signal consistent with the training data and encoding sufficient input variability to capture the variations across data examples. The latent random variable models can avoid this compromise by adding variability in the latent space, which can always be mapped to a point close to a relatively clean sample.*"

5 Comments

- Didn't get the following from 2.1: *"...it's important to note that any approach based on having stochastic latent state is orthogonal to having a structured output function,..."*
- I have little RNN experience and this decomposition of an RNN into generating distribution parameters and then outputting the probabilities given these params looks very exciting to me. Instead of thinking of an RNN

as some function composition without any semantics, this insight adds semantics to the model structure.

References

[Theis et al., 2015] Theis, L., Oord, A. v. d., and Bethge, M. (2015). A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844*.