The Conditional Bernoulli and its Application to Speech Recognition

Sean Robertson

March 25, 2020

1 Motivations

A major challenge in speech recognition involves converting a variable number of speech frames $\{x_t\}_{t\in[0,T)}$ into a variable number of transcription tokens $\{c_\ell\}_{\ell\in[0,L)}$, where $L \ll T$. In hybrid architectures, c_ℓ are generated as a by-product of transitioning between states s_t in a weighted finite-state transducer. In end-to-end neural ASR, this process is commonly achieved either with Connectionist Temporal Classification (CTC) [5] or sequence-to-sequence (seq2seq) architectures [1]. The former introduces a special blank label; performs a one-to-many mapping $c_\ell \mapsto \tilde{c}_t^{(i)}$ by injecting blank tokens until the transcription matches length T in all possible configurations (i) during training; and removes all blank labels during testing. Seq2seq architectures first encode the speech frames x_t into some encoding h, then some separate recurrent neural network conditions on h to generate the token sequence c_t.

In 2017, Luo et al. developed a novel end-to-end speech recognizer. Given a prefix of acoustic feature frames including the current frame $\{x_{t'}\}_{t'\le t<T}$ and a prefix of Bernoulli samples excluding the current frame $\{b_{t'}\}_{t'<t<T}$, the recognizer produces a Bernoulli sample for the current frame $B_t \sim P_B(b_t|x_{\le t}, b_{<t})$, plus or minus some additional conditioned terms. Whenever $B_t = 1$, the model "emits" a token drawn from a class distribution conditioned on the same information $C_t \sim P_C(c_t|x_{\le t}, b_{<t})$. The paper had two primary motivations. First, though it resembles a decoder in a *seq2seq* architecture [1], it does not need to encode the entire input sequence x_t before it can start making decisions about what was said, making it suitable to online recognition. Second, we can interpret the emission points, or "highs," of the Bernoulli sequence $B_t = 1$ as a form of hard alignment: the token output according to C_t is unaffected by speech $x_{>t}$[1].

Because of the stochasticity introduced by sampling B_t discretely, the network cannot backpropagate through the Bernoulli parameterizations w. Thus,

[1] This is not necessarily a synchronous alignment. $B_t = 1$ may occur well after whatever caused the emission. The last high $\arg\max_{t'<t} B_{t'} = 1$ cannot be assumed to bound the event to times after t' for the same reason. Finite t and vanishing gradients will force some synchronicity, however.

the authors rely on a REINFORCE gradient estimate [9]:

$$\frac{\partial R}{\partial w_t} = \mathbb{E}_{b_t}\left[\left(\sum_{t' \geq t} R_{t'}\right) \frac{\partial \log P_B(b_t | x_{\leq t}, b_{<t})}{\partial w_t}\right] \tag{1}$$

Where

$$R_t = \begin{cases} \log P_C(C_t = c_{\sum_{t'<t} b_{t'}} | x_{\leq t}, b_{<t}) & \text{if } B_t = 1 \\ 0 & \text{if } B_t = 0 \end{cases} \tag{2}$$

The reward (eq. (2)) is the log probability of the k-th class label, where k is the number of high Bernoulli values up to and including time t whenever $B_t = 1$. The return for time step t accumulates the instantaneous rewards for all non-past time steps $t' \geq t$.

In practice, using eq. (1) is very slow to train and yields mixed results. The authors found it was necessary to add a baseline function and an entropy function in order to converge. In a later publication [6], a bidirectional model[2] used Variational Inference to speed up convergence, though this failed to improve the overall performance of the model on the TIMIT corpus. The mixed performance and convergence of these models was blamed on the high-variance gradient estimate of eq. (1) [6].

We believe that the performance and convergence issues of these models are not due, at least in whole, to a high-variance estimate. Instead, we propose that the training objective has two other critical issues.

First, under the current regime, there is no natural choice of reward for when $B_t = 0$. Equation (1) accumulates future rewards to mitigate this, but the choice to do so biases the system to emit as soon as possible to reduce the number of total frames accumulating negative rewards. This bias could explain why, without an additional "entropy penalty," the model would learn to emit entirely at the beginning of the utterance [7].

Second, in order to ensure the total number of high Bernoulli values matched the total number of labels L during training, i.e. $\sum_t b_t = L$, the authors would force later samples to some specific value. This implies that $B_t \not\sim P_B(b_t | \ldots)$, making the Monte Carlo estimate of eq. (1) biased. This bias could interact with the previous issue to produce a model that learns to immediately and repeatedly emit without stopping, since $b_{\geq L} = 0$, pushing $P_B(b_t | \ldots) \to 1$.

To solve both of these problems, we propose replacing the T i.i.d. Bernoulli random variables B_t sampled during training with a single sample B from the Conditional Bernoulli (CB) distribution, discussed in section 2. We will show that the switch elegantly supports the same inference procedure. Further, since the CB uses parameters generated at all time steps, rather than just the current time step, the rewards from when $B_t = 1$ will induce an error signal in the parameters for $B_{t'} = 0$. In addition to modifying eq. (1), we will also show how the CB can be applied to other gradient estimation methods, such as Straight-Through Estimators (STEs) [2] or RELAX-like estimators [8, 4].

[2] Forgoing the motivation for online speech recognition.

2 The Conditional Bernoulli

The Conditional Bernoulli, discussed initially by Chen et al. [3], is defined as

$$P\left(b \middle| \sum_i b_i = k; w\right) = \frac{\prod_i w_i^{b_i}}{\sum_{\{b':\sum_\ell b'_\ell = k\}} \prod_j w_i^{b'_j}} \tag{3}$$

Where $w_i = p_i/(1-p_i)$ are the odds of a Bernoulli random variable $B_i \sim P(b_i; w_i) = p_i^{b_i}(1-p_i)^{(1-b_i)} = w_i^{b_i}(1-p_i)$. Equation (3) reads as "what is the probability that Bernoulli random variables $\{B_i\}_i$ have values $\{b_i\}_i$, given that exactly k of them are high ($\sum_i b_i = k$)?" Letting $K = \sum_i B_i$, K is a random variable that counts the total number of "highs" in a series of Bernoulli trials. K is distributed according to the Poisson-Binomial distribution, a generalization of the Binomial distribution for when $p_i \neq p_j$. It is defined as

$$\begin{aligned} P(K = k; w) &= \sum_{\{b:\sum_\ell b_\ell = k\}} P(b; w) \\ &= \left(\prod_i (1-p_i)\right) \sum_{\{b:\sum_\ell b_\ell = k\}} \prod_j w_j^{b_j} \end{aligned} \tag{4}$$

If we use eq. (4) to marginalize out K from eq. (3), we recover the independent Bernoulli probabilities:

$$\begin{aligned} P(b; w) &= \sum_k P(b|k)P(k) \\ &= P(b|k')P(k') \text{ for some } k' = \sum_i b_i \\ &= \left(\prod_i (1-p_i)\right) \frac{\prod_i w_i^{b_i}}{\sum_{\{b':\sum_\ell b'_\ell = k'\}} \prod_j w_i^{b'_j}} \left(\sum_{\{b':\sum_\ell b'_\ell = k'\}} \prod_j w_j^{b'_j}\right) \\ &= \prod_i (1-p_i) w_i^{b_i} \end{aligned} \tag{5}$$

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations*, ICLR '15, San Diego, USA, 2015. URL `http://arxiv.org/abs/1409.0473`.

[2] Yoshua Bengio, Nicholas Léonard, and Aaron C. Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *CoRR*, abs/1308.3432, 2013. URL `http://arxiv.org/abs/1308.3432`.

[3] Xiang-Hui Chen, Arthur P. Dempster, and Jun S. Liu. Weighted finite population sampling to maximize entropy. *Biometrika*, 81(3):457–469, 1994. ISSN 00063444. doi: 10.2307/2337119.

[4] Will Grathwohl, Dami Choi, Yuhuai Wu, Geoffrey Roeder, and David Duvenaud. Backpropagation through the void: Optimizing control variates for black-box gradient estimation. In *6th International Conference on Learning Representations*, ICLR '18, Vancouver, Canada, 2018. URL `https://openreview.net/forum?id=SyzKd1bCW`.

[5] Alex Graves, Santiago Fernández, Faustino Gomez, and Jürgen Schmidhuber. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural networks. In *International Conference on Machine Learning*, ICML '06, pages 369–376, New York, NY, USA, 2006. ACM. ISBN 1-59593-383-2. doi: 10.1145/1143844.1143891.

[6] Dieterich Lawson, Chung-Cheng Chiu, George Tucker, Colin Raffel, Kevin Swersky, and Navdeep Jaitly. Learning hard alignments with variational inference. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, ICASSP '18, pages 5799–5803, April 2018. ISBN 2379-190X. doi: 10.1109/ICASSP.2018.8461977.

[7] Yuo Luo, Chung-Cheng Chiu, Navdeep Jaitly, and Ilya Sutskever. Learning online alignments with continuous rewards policy gradient. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, ICASSP '17, pages 2801–2805, March 2017. doi: 10.1109/ICASSP.2017.7952667.

[8] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The Concrete Distribution: A continuous relaxation of discrete random variables. In *5th International Conference on Learning Representations*, ICLR '17, Toloun, France, 2017. URL `https://openreview.net/forum?id=S1jE5L5gl`.

[9] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3):229–256, May 1992. ISSN 1573-0565. doi: 10.1007/BF00992696.