KL vanishing 问题的一般解

本文主要针对阿里最近发表于 ICASSP2019 上的一篇论文“Improve Diverse Text Generation by Self Labeling Conditional Variational Auto Encoder”[1]，通过博弈论的角度分析这篇论文存在的一些问题，并给出了 KL vanishing 问题的一般解。因此，本文的贡献主要包含了两方面：

1. 尝试从博弈论的角度分析 KL vanishing problem 的纳什均衡解，并从这个角度得出阿里的这篇论文并不能很好地解决 KL vanishing problem
2. 从原始的 VAE 优化函数出发，通过引入互信息的限制条件来解决 KL vanishing problem，并从博弈论角度给出文本提出的优化算法的纳什均衡解不再会遇到 KL vanishing 问题

KL vanishing problem

在 VAE 里面，KL vanishing problem 主要是指当 KL(q(z|x) || p(z))消失时，并且当隐变量 z 和 x 独立时，此时的 ELBO=log(p(x))【推导过程见 Supplemental Materials】。由 KL(q(z|x) || p(z|x)) = log(p(x)) - ELBO，得到此时 KL(q(z|x) || p(z|x))=0，因此得出 q(z|x)==p(z|x)，但这个解并不是我们需要的理想解，因为此时的互信息 I(x, z)=0。

从博弈论观点看待 KL vanishing problem

通过对 KL vanishing problem 的分析，假设当 KL(q(z|x) || p(z))=0 时，存在着两种情况，一个是 z 与 x 独立， 另一个是 z 和 x 不独立。我们想得到的解是当 KL(q(z|x) || p(z))=0 时，z 和 x 不独立。现在我们引入博弈论的思想，假设模型的两个 modules 之间进行博弈，它们分别是决定 z 和 x 独立与否的 module A，以及决定 q(z|x)于 p(z)是否相等的 module B。而它们的收益函数都是 ELBO 或 ELBO 的相关变种。
首先对传统 VAE 进行分析，得到 module A 和 module B 的收益矩阵【计算过程见 Supplemental Materials】：

Module A \ Module B	q(z|x) == p(z)	q(z|x) ≠ p(z)
z 与 x独立	(logp(x), logp(x))	(logp(x)-KL(q(z),p(z)), logp(x)-KL(q(z),p(z))
z 与 x不独立	($E_{p(z)}$[logp(x|z)], $E_{p(z)}$[logp(x|z)])	(ELBO, ELBO)

由于 $E_{p(z)}$[logp(x|z)] <= logp(x)，当且仅当 x 和 z 独立时等式成立，因此可以得到纳什均衡解是(z 与 x 独立，q(z|x) == p(z))，此时就会造成所谓的 KL vanishing problem。

而“Improve Diverse Text Generation by Self Labeling Conditional Variational Auto Encoder”[1]这篇论文提出通过引入新的损失函数：

$$\mathcal{L}_{\text{SLVAE}} = -\mathbb{E}_{q(z|x)}[\log p(x|z)] + KL(q(z|x)||p(z)) \\ +\lambda\mathbb{E}_{q(z|x)}[||z - g(x)||^2]$$

可以避免出现 KL vanishing problem。他们的基本思想是，通过使隐变量 z 包含 x 的信息，从而避免模型收敛到(z 与 x 独立，q(z|x) == p(z))这个解。但这种方法并不能完全避免模型收敛到(z 与 x 独立, q(z|x) == p(z))。我们先一般化这个思路，即定义新的$ELBO' = ELBO + f(x, z; \pi)$，具体的，这里的

$f(x,z;\pi) = -\lambda E_{q(z|x)}[||z-g(x)||^2]$，此处的 π 就是

module A 和 module B 采取的策略，即 module A 选择 z 是否与 x 独立，module

B 选择 q(z|x)是否等于 p(z)。由此，我们得到新的收益矩阵：

Module A \ Module B	q(z|x) == p(z)	q(z|x) ≠ p(z)
z 与 x 独立	($logp(x) + f(x,z;\pi_{0,0})$, $logp(x) + f(x,z;\pi_{0,0})$)	($logp(x) - KL(q(z),p(z)) + f(x,z;\pi_{0,1})$, $logp(x) - KL(q(z),p(z)) + f(x,z;\pi_{0,1})$)
z 与 x 不独立	($E_{p(z)}[logp(x|z)] + f(x,z;\pi_{1,0})$, $E_{p(z)}[logp(x|z)] + f(x,z;\pi_{1,0})$)	($ELBO + f(x,z;\pi_{1,1})$, $ELBO + f(x,z;\pi_{1,1})$)

因为要想让模型收敛到纳什均衡解(z 与 x 不独立，q(z|x) == p(z))，必须满足：

$$\begin{cases} E_{p(z)}[logp(x|z)] + f(x,z;\pi_{1,0}) > logp(x) + f(x,z;\pi_{0,0}) & (1) \\ E_{p(z)}[logp(x|z)] + f(x,z;\pi_{1,0}) > ELBO + f(x,z;\pi_{1,1}) & (2) \end{cases}$$

但是当 $f(x,z;\pi) = -\lambda E_{q(z|x)}[||z-g(x)||^2]$ 时，由于

$f(x,z;\pi_{0,0}) = f(x,z;\pi_{1,0}) = -\lambda E_{p(z)}[||z-g(x)||^2]$，而当 z 与 x 不独立时，$E_{p(z)}[logp(x|z)] < logp(x)$，故条件(1)不成立。因此此时的纳什均衡解依然是(z 与 x 独立，q(z|x) == p(z))。所以论文里提出的算法并不能解决 KL vanishing problem。

新方法

我们发现如果尝试去通过修改中间优化函数，比如 ELBO，往往会导致最终得到的新的 objective function 不再和最原始的 objective function：min KL(q(z|x), p(z|x)) 等价，这样的 objective function 往往是一个四不像，虽然从局部分析往往对模型有利，但如果从全局进行分析就会发现这样做会顾此失彼。因此，能否在原始的 objective function 上做修改，比如引入一个限制条件，使得隐变量 z 和 x 的互信息大于某个常数，这样我们能够保证求解得

到的解与求解原始 objective function 等价。

定义新的 objective function

由于 VAE 最初的目标是优化 KL(q(z|x), p(z|x)), 而我们只要在优化过程中引入一些限制条件，比如使得 x 和 z 的互信息 I(x,z) >= c，因此得到下面新的 objective function：

$$\begin{aligned} min \qquad & KL(q(z|x), p(z|x)) \\ & s.t. \qquad I(x,z) >= c \end{aligned}$$

通过引入拉格朗日乘子，得到不带约束的优化问题：

$$\begin{aligned} min \quad \max_{\lambda} \qquad & KL(q(z|x), p(z|x)) + \lambda(c - I(x,z)) \qquad \lambda >= 0 \\ & \approx KL(q(z|x), p(z|x)) + \lambda max(c - I(x,z), 0) \end{aligned}$$

我们可以看出，这个优化目标函数可以使得当 I(x, z) < c 时，将得到一个惩罚项，只有当 I(x, z) >= c 时，惩罚项才为 0。

由

$$KL(q(z|x), p(z|x)) = logp(x) - ELBO$$

因此，若 I(x, z) >= c，则

$$\begin{aligned} min \qquad & KL(q(z|x), p(z|x)) + \lambda max(c - I(x,z), 0) \\ & \iff KL(q(z|x), p(z|x)) \\ \iff min \qquad & -ELBO + logp(x) \qquad (3) \end{aligned}$$

而当 I(x, z) < c 时，由

$$\begin{aligned} I(x,z) &= KL(q(x,z), p(x)p(z)) \\ &= E_{p(x)}[KL(q(z|x), p(z))] \end{aligned}$$

进一步得到：

$$\begin{aligned} min \qquad & KL(q(z|x), p(z|x)) + \lambda max(c - I(x,z), 0) \\ & \Rightarrow KL(q(z|x), p(z|x)) + \lambda(c - I(x,z)) \\ & \Rightarrow logp(x) - ELBO + \lambda(c - E_{p(x)}[KL(q(z|x), p(z))]) \\ & \Rightarrow -ELBO - \lambda E_{p(x)}[KL(q(z|x), p(z))] + \lambda c + logp(x) \\ & \Rightarrow -ELBO - \lambda E_{p(x)}[KL(q(z|x), p(z))] + const \qquad (4) \\ & where \qquad const = \lambda c + logp(x) \end{aligned}$$

在实际训练的时候，我们让：

$$E_{p(x)}[KL(q(z|x), p(z))] \approx \frac{\sum_{i=1}^{n} KL(q(z|x_i), p(z))}{n}$$

因此新的求解算法具体过程如下：

1. Choose hyper-parameters: λ, c , and initialize model parameters θ randomly
2. While not stopped:
   Sample a mini-batch data: {x_1, x_2, ..., x_n}
   Calculate mini-batch mean KL(q(z|x), p(z)):

$$\overline{KL(q(z|x), p(z))} = \frac{\sum KL(q(z|x_i), p(z))}{n}$$

$$if \qquad \overline{KL(q(z|x), p(z))} >= c$$

Calculate gradients:

$$\nabla_\theta - ELBO$$

$$if \qquad \overline{KL(q(z|x), p(z))} < c$$

Calculate gradients:

$$\nabla_\theta - ELBO - \lambda \overline{KL(q(z|x), p(z))}$$

Update model parameters θ

算法分析

文本提出的优化算法的收益矩阵为【计算过程见 Supplemental Materials】：

Module A \ Module B	q(z|x) == p(z)	q(z|x) ≠ p(z)
I(x, z) < c	($E_{p(z)}[logp(x\|z)] - \lambda c - logp(x)$, $E_{p(z)}[logp(x\|z)] - \lambda c - logp(x)$)	($ELBO + \lambda E_{p(x)}[KL(q(z\|x), p(z))] - \lambda c - logp(x)$, $ELBO + \lambda E_{p(x)}[KL(q(z\|x), p(z))] - \lambda c - logp(x)$)
I(x, z) >= c	($E_{p(z)}[logp(x\|z)] - logp(x)$, $E_{p(z)}[logp(x\|z)] - logp(x)$)	($ELBO - logp(x)$, $ELBO - logp(x)$)

特别的，当 I(x, z)=0，并且 q(z|x) == p(z)时 $E_{p(z)}[logp(x|z)] - \lambda c - logp(x)$ 的值取最大：

$$max \quad E_{p(z)}[logp(x|z)] - \lambda c - logp(x) \\ = -\lambda c$$

根据收益矩阵可以得出，我们只要满足条件：

$$\begin{cases} E_{p(z)}[logp(x|z)] - logp(x) > -\lambda c \qquad (5) \\ \lambda E_{p(x)}[KL(q(z|x), p(z))] - \lambda c < 0 \quad (where \quad I(x,z) < c, \quad q(z|x) \neq p(z)) \qquad (6) \end{cases}$$

而这两项条件只要我们设置合理的 λ, c 值就可以满足。此时的纳什均衡解为（I(x, z) >= c, q(z|x) == p(z)），而这正是我们希望得到的解，因为此时在保证 q(z|x) == p(z)的同时，x 与 z 不独立。

与β-VAE 对比

我们知道β-VAE 可以学到比较好的解耦表示，即使得每个维度的隐变量 z_i 之间的相关度很小，但同时它会减小隐变量 z 与 x 之间的互信息[2]。因此会导致 reconstruction 网络性能的下降。由于β-VAE 只是简单地增大对 KL(q(z|x), p(z))的惩罚，随着训练的进行，有可能会 push 隐变量的分布朝着跟 x 独立的分布的方向去优化。而本文提出的算法，可以动态去调整 KL(q(z|x), p(z))的惩罚项，从而避免减小隐变量 z 和 x 之间的互信息。同时我们发现，在优化目标函数(3)和(4)里引入β(β>1)后，收益矩阵变为：

Module A \ Module B	q(z|x) == p(z)	q(z|x) ≠ p(z)
I(x, z) < c	($E_{p(z)}[logp(x\|z)] - \lambda c - logp(x)$, $E_{p(z)}[logp(x\|z)] - \lambda c - logp(x)$)	($E_{q(z\|x)}[logp(x\|z)] - \beta KL(q(z\|x), p(z)) + \lambda E_{p(x)}[KL(q(z\|x), p(z))] - \lambda c - logp(x)$, $E_{q(z\|x)}[logp(x\|z)] - \beta KL(q(z\|x), p(z)) + \lambda E_{p(x)}[KL(q(z\|x), p(z))] - \lambda c - logp(x)$)
I(x, z) >= c	($E_{p(z)}[logp(x\|z)] - logp(x)$, $E_{p(z)}[logp(x\|z)] - logp(x)$)	($E_{q(z\|x)}[logp(x\|z)] - \beta KL(q(z\|x), p(z)) - logp(x)$, $E_{q(z\|x)}[logp(x\|z)] - \beta KL(q(z\|x), p(z)) - logp(x)$)

此时依然只要满足条件：

$$\begin{cases} E_{p(z)}[logp(x|z)] - logp(x) > -\lambda c \qquad (5) \\ \lambda E_{p(x)}[KL(q(z|x), p(z))] - \lambda c < 0 \quad (where \quad I(x,z) < c, \quad q(z|x) \neq p(z)) \qquad (6) \end{cases}$$

就可以得到纳什均衡解：（I(x, z) >= c, q(z|x) == p(z)），并且此时也可以学到比较好的解耦表示。

References

1. Zhang Y , Wang Y , Zhang L , et al. Improve Diverse Text Generation by Self Labeling Conditional Variational Auto Encoder[J]. 2019.

2. Chen R T Q, Li X, Grosse R, et al. Isolating Sources of Disentanglement in Variational Autoencoders[J]. 2018.

Supplemental Materials

1. 当 KL(q(z|x) || p(z))=0，并且隐变量 z 和 x 独立时，ELBO=logp(x)

Proof:

$$
\begin{aligned}
ELBO = E_{q(z|x)}[logp(x|z)] - KL(q(z|x), p(z)) \\
= E_{q(z|x)}[logp(x|z)](\because KL(q(z|x), p(z)) = 0) \\
= E_{p(z)}[logp(x|z)] \\
\leq log(E_{p(z)}[p(x|z)]) \quad (Jensen's \quad inequality) \\
= logp(x)
\end{aligned}
$$

上面不等式等号成立的条件是隐变量 z 和 x 独立。因此 ELBO=logp(x)。
证毕。

2. 传统 VAE module A 和 module B 的收益矩阵(此处假设 module A 和 module B 的收益都是优化目标函数值，下同):

Module A \ Module B	q(z|x) == p(z)	q(z|x) ≠ p(z)
z 与 x 独立	(logp(x), logp(x))	(logp(x)-KL(q(z),p(z)), logp(x)-KL(q(z),p(z))
z 与 x 不独立	($E_{p(z)}$[logp(x|z)], $E_{p(z)}$[logp(x|z)])	(ELBO, ELBO)

$$
ELBO = E_{q(z|x)}[logp(x|z)] - KL(q(z|x), p(z))
$$

当隐变量 z 和 x 独立并 q(z|x) == p(z)时，由 1 得到此时 ELBO=logp(x)

当隐变量 z 和 x 不独立并 q(z|x) ≠ p(z)时，即原始的 ELBO 值

当隐变量 z 和 x 独立并 q(z|x) ≠ p(z)时：

$$\begin{aligned} ELBO &= E_{q(z|x)}[logp(x|z)] - KL(q(z|x), p(z)) \\ &= E_{q(z)}[logp(x)] - KL(q(z), p(z)) \\ &= logp(x) - KL(q(z), p(z)) \end{aligned}$$

当隐变量 z 和 x 不独立并 q(z|x) == p(z)时：

$$\begin{aligned} ELBO &= E_{q(z|x)}[logp(x|z)] - KL(q(z|x), p(z)) \\ &= E_{p(z)}[logp(x|z)] - KL(q(z|x), p(z)) \\ &= E_{p(z)}[logp(x|z)] \end{aligned}$$

3. 文本提出的优化算法的收益矩阵为：

Module A \ Module B	q(z|x) == p(z)	q(z|x) ≠ p(z)
I(x, z) < c	($E_{p(z)}[logp(x\|z)] - \lambda c - logp(x)$, $E_{p(z)}[logp(x\|z)] - \lambda c - logp(x)$)	($ELBO + \lambda E_{p(x)}[KL(q(z\|x), p(z))] - \lambda c - logp(x)$, $ELBO + \lambda E_{p(x)}[KL(q(z\|x), p(z))] - \lambda c - logp(x)$)
I(x, z) >= c	($E_{p(z)}[logp(x\|z)] - logp(x)$, $E_{p(z)}[logp(x\|z)] - logp(x)$)	($ELBO - logp(x)$, $ELBO - logp(x)$)

当 I(x, z) >= c 时，收益函数为：

$$\begin{aligned} & min \qquad - ELBO + logp(x) \\ \iff & max \qquad ELBO - logp(x) \quad (3) \end{aligned}$$

当 I(x, z) < c 时，收益函数为：

$$\begin{aligned} & min \quad - ELBO - \lambda E_{p(x)}[KL(q(z|x), p(z))] + const \\ & \qquad\qquad where \qquad const = \lambda c + logp(x) \\ \iff & max \quad ELBO + \lambda E_{p(x)}[KL(q(z|x), p(z))] - const \quad (4) \end{aligned}$$

当 I(x, z) < c 并 q(z|x) == p(z)时，由(4)得：

$$
\begin{aligned}
&ELBO + \lambda E_{p(x)}[KL(q(z|x), p(z))] - const \\
&\quad = E_{p(z)}[logp(x|z)] - \lambda c - logp(x)
\end{aligned}
$$

特别的，当 I(x, z)=0 时值取最大，

$$
\begin{aligned}
max \quad & E_{p(z)}[logp(x|z)] - \lambda c - logp(x) \\
&= logp(x) - \lambda c - logp(x) \\
&= -\lambda c
\end{aligned}
$$

当 I(x, z) < c 并 q(z|x) ≠ p(z)时，由(4)得：

$$
\begin{aligned}
&ELBO + \lambda E_{p(x)}[KL(q(z|x), p(z))] - const \\
&= E_{q(z|x)}[logp(x|z)] - KL(q(z|x), p(z)) + \lambda E_{p(x)}[KL(q(z|x), p(z))] - \lambda c - logp(x)
\end{aligned}
$$

当 I(x, z) >= c 并 q(z|x) == p(z)时，由(3)得：

$$
\begin{aligned}
ELBO - logp(x) &= E_{q(z|x)}[logp(x|z)] - KL(q(z|x), p(z)) - logp(x) \\
&= E_{p(z)}[logp(x|z)] - logp(x)
\end{aligned}
$$

当 I(x, z) >= c 并 q(z|x) ≠ p(z)时，由(3)得：

$$
ELBO - logp(x) = E_{q(z|x)}[logp(x|z)] - KL(q(z|x), p(z)) - logp(x)
$$