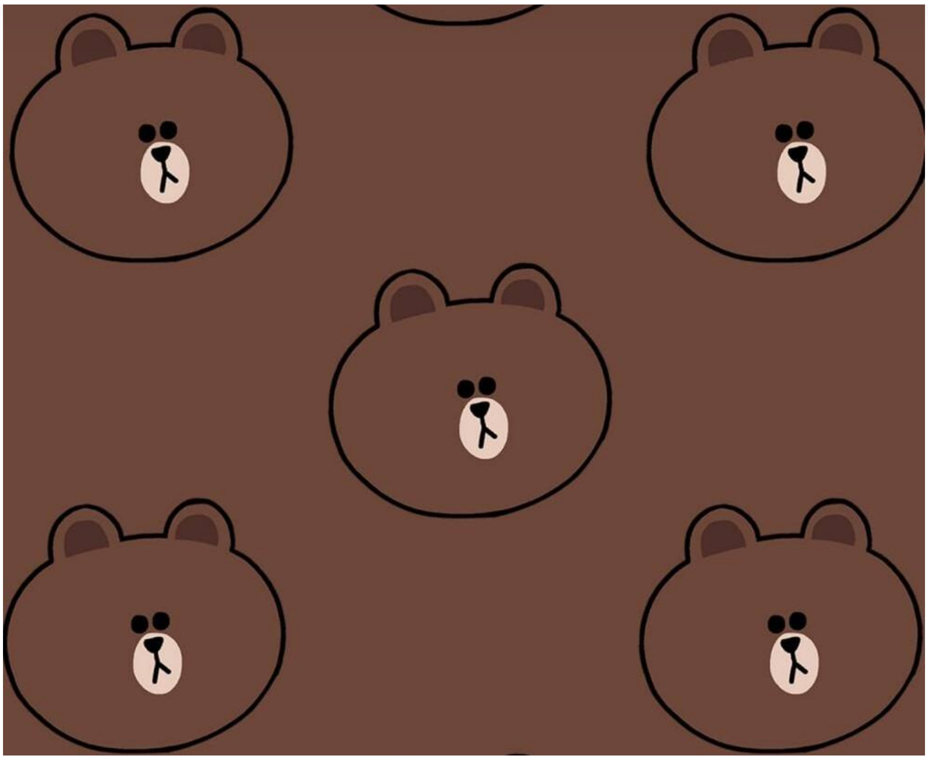

Diffusion学习笔记（五）——Conditional Control（Classifier-Guidance and Classifier-Free）

Hammour Yue
AI新人，菜鸡学生

前几篇文章都是讨论无条件生成式的Diffusion模型，只能随机采样，无法控制模型的输出。但很多时候，我们要求得到与指定文本信息或者与图像信息对应的输出（即文生图或图生图），这就需要用到条件控制生成技术了。而真正让Diffusion出圈的也正是条件控制生成技术，例如Stable Diffusion，Dall ·E 2。若将指定的条件（或标签）设为 y ，相比无条件的采样过程 $p(x_{t-1} \mid x_t)$ ，条件生成的最终目的是为了得到条件采样过程 $p(x_{t-1} \mid x_t, y)$ 。关于条件控制一般分为**Classifier-Guidance**和**Classifier-Free**，下面对两种方法进行简要介绍。

1 Classifier-Guidance

Classifier-Guidance也叫“**事后修改**”方案，即给定了一个训练好的无条件Diffusion模型，再进行条件控制输出，最早出现在《Diffusion Models Beat GANs on Image Synthesis》中。作者主要对 $p(x_{t-1} \mid x_t, y)$ 进行了一些变化：

$$\begin{aligned} p(x_{t-1} \mid x_t, y) &= \frac{p(x_{t-1}, x_t, y)}{p(x_t, y)} \\ &= \frac{p(y \mid x_{t-1}, x_t) p(x_{t-1} \mid x_t) p(x_t)}{p(y \mid x_t) p(x_t)} \\ &= \frac{p(y \mid x_{t-1}, x_t) p(x_{t-1} \mid x_t)}{p(y \mid x_t)} \end{aligned} \tag{1.1}$$

其中：

$$\begin{aligned} p(y \mid x_{t-1}, x_t) &= \frac{p(y, x_{t-1}, x_t)}{p(x_{t-1}, x_t)} \\ &= \frac{p(x_t \mid x_{t-1}, y) p(y \mid x_{t-1}) p(x_{t-1})}{p(x_t \mid x_{t-1}) p(x_{t-1})} \\ &= \frac{p(x_t \mid x_{t-1}, y) p(y \mid x_{t-1})}{p(x_t \mid x_{t-1})} \end{aligned} \tag{1.2}$$

显然 $p(x_t \mid x_{t-1}, y) = p(x_t \mid x_{t-1})$ ，这是因为前向过程是固定的与条件 y 无关，因此（1.2）变成：

$$p(y \mid x_{t-1}, x_t) = p(y \mid x_{t-1}) \tag{1.3}$$

这一步与DDPM中变换KL散度的其中一步很相似，也即文章[1]的（3.12）式第5个等号。关于

价于 $p(y \mid x_{t-1})$ 。最后将关系式带入回（1.1）有：

$$\begin{aligned} p(x_{t-1} \mid x_t, y) &= \frac{p(y \mid x_{t-1}, x_t)p(x_{t-1} \mid x_t)}{p(y \mid x_t)} \\ &= \frac{p(y \mid x_{t-1})p(x_{t-1} \mid x_t)}{p(y \mid x_t)} \\ &= p(x_{t-1} \mid x_t)e^{\log p(y|x_{t-1}) - \log p(y|x_t)} \end{aligned} \tag{1.4}$$

为了处理这样的表达式，我们可以在理论分析的时候将前向和后向过程当作其对应的连续SDE，也就是此时 x_{t-1} 相当于 x_{t-dt} （类似上篇文章（17）式的处理方式），因此 $\log p(y \mid x_{t-1})$ 关于 $(x_{t-1}, t-1)$ 就可以当作一个连续二元函数，这就是研究SDE理论带来的好处。与[2]中第4部分的引用部分的操作一样，对函数 $\log p(y \mid x_{t-1})$ 在点 (x_t, t) 处二元泰勒展开：

$$\begin{aligned} \log p(y \mid x_{t-1}) &\approx \log p(y \mid x_{t-1})|_{x_{t-1}=x_t} + (x_{t-1} - x_t)[\nabla_{x_{t-1}} \log p(y \mid x_{t-1})]|_{x_{t-1}=x_t} + \frac{\partial}{\partial t}\log p(y \mid x_{t-1})|_{x_{t-1}=x_t} \\ &= \log p(y \mid x_t) + (x_{t-1} - x_t)\nabla_{x_t} \log p(y \mid x_t) + \frac{\partial}{\partial t}\log p(y \mid x_t) \end{aligned} \tag{1.5}$$

接着就很简单了，直接将（1.5）式子带入（1.4）可以得到：

$$\begin{aligned} p(x_{t-1} \mid x_t, y) &= p(x_{t-1} \mid x_t)e^{\log p(y|x_{t-1}) - \log p(y|x_t)} \\ &\approx p(x_{t-1} \mid x_t)e^{(x_{t-1}-x_t)\nabla_{x_t} \log p(y|x_t) + \frac{\partial}{\partial t}\log p(y|x_t)} \\ &\approx p_\theta(x_{t-1} \mid x_t)e^{(x_{t-1}-x_t)\nabla_{x_t} \log p(y|x_t) + \frac{\partial}{\partial t}\log p(y|x_t)} \\ &\propto e^{-\frac{1}{2}(x_{t-1}-\mu_\theta(x_t))^T\Sigma_\theta^{-1}(x_{t-1}-\mu_\theta(x_t)) + x_{t-1}\nabla_{x_t} \log p(y|x_t)} \end{aligned} \tag{1.6}$$

其中 $p_\theta(x_{t-1} \mid x_t) = N(\mu_\theta(x_t), \Sigma_\theta(t))$ 。最后一步写成这样的目的是为了接下来可以配方，得到正态分布的均值和方差，其它与 x_{t-1} 无关的常数项并不关心。而对于形如 $x^TAx + x^Tb$ 的二次型，希望配方成 $(x+u)^TA(x+u) + C$ ，展开可知：

$$\begin{aligned} (x+u)^TA(x+u) + C &= x^TAx + x^TAu + u^TAx + u^TAu + C \\ &= x^TAx + 2x^TAu + u^TAu + C \end{aligned} \tag{1.7}$$

对比可知 $u = \frac{1}{2}A^{-1}b$ ， $C = -\frac{1}{4}b^TA^{-1}b$ ，因此：

$$\begin{aligned} \log p(x_{t-1} \mid x_t, y) &\propto -\frac{1}{2}(x_{t-1} - \mu_\theta(x_t))^T\Sigma_\theta^{-1}(x_{t-1} - \mu_\theta(x_t)) + x_{t-1}\nabla_{x_t} \log p(y \mid x_t) \\ &\propto -\frac{1}{2}(x_{t-1} - \mu_\theta(x_t))^T\Sigma_\theta^{-1}(x_{t-1} - \mu_\theta(x_t)) + (x_{t-1} - \mu_\theta(x_t))\nabla_{x_t} \log p(y \mid x_t) \\ &= -\frac{1}{2}[(x_{t-1} - \mu_\theta(x_t))^T\Sigma_\theta^{-1}(x_{t-1} - \mu_\theta(x_t)) - 2(x_{t-1} - \mu_\theta(x_t))\nabla_{x_t} \log p(y \mid x_t)] \\ &\propto -\frac{1}{2}(x_{t-1} - \mu_\theta(x_t) - \Sigma_\theta\nabla_{x_t} \log p(y \mid x_t))^T\Sigma_\theta^{-1}(x_{t-1} - \mu_\theta(x_t) - \Sigma_\theta\nabla_{x_t} \log p(y \mid x_t)) \end{aligned} \tag{1.8}$$

即：

$$p_\theta(x_{t-1} \mid x_t, y) = N(\mu_\theta(x_t) + \Sigma_\theta\nabla_{x_t} \log p(y \mid x_t), \Sigma_\theta) \tag{1.9}$$

因此如果已知了无条件Diffusion模型 $p_\theta(x_{t-1} \mid x_t)$ ，只需要再训练一个分类器 $p_\phi(y \mid x_t)$ ，最终用：

$$p_{\theta,\phi}(x_{t-1} \mid x_t, y) = N(\mu_\theta(x_t) + \Sigma_\theta\nabla_{x_t} \log p_\phi(y \mid x_t), \Sigma_\theta) \tag{1.10}$$

对其进行采样，就能达到条件控制生成的目的，所以叫“事后修改”。可以看到条件控制的 $p_{\theta,\phi}(x_{t-1} \mid x_t, y)$ 实际就是在无条件的 $p_\theta(x_{t-1} \mid x_t)$ 采样的基础上，均值多了 $\Sigma_\theta\nabla_{x_t} \log p_\phi(y \mid x_t)$ 的漂移。

另外，回到（1.5）式，我们只将其展开到一次项，是为了处理简单，因为这样还能保持 $p(x_{t-1} \mid x_t, y)$ 为正态分布，所以只需要处理与 x_{t-1} 有关的项即可，因此原文中作者将（1.4）式中改写成了：

$$\begin{aligned} p(x_{t-1} \mid x_t, y) &= Zp(y \mid x_{t-1})p(x_{t-1} \mid x_t) \\ &\propto p(x_{t-1} \mid x_t)e^{\log p(y|x_{t-1})} \end{aligned} \tag{1.11}$$

并且原论文实际上是在 $(\mu_\theta(x_{t+1}), t)$ 处展开，而 $\mu_\theta(x_{t+1})$ 实际是对 x_t 的一阶矩估计（期望），所以最后原论文的结果和（1.10）是近似等价的，因此原论文的结果其实是（1.4）式的一

接添加漂移 $\Sigma_\theta \nabla_{x_t} \log p_\phi(y \mid x_t)$ 效果一般，但作者又加以改进，添加了条件放缩因子 $\gamma > 1$ ，最终的条件采样分布为：

$$p_\theta(x_{t-1} \mid x_t, y) = N(\mu_\theta(x_t) + \gamma \Sigma_\theta \nabla_{x_t} \log p(y \mid x_t), \Sigma_\theta) \tag{1.12}$$

> 这是作者通过实验给出的经验公式，他对此解释为：
>
> $$\gamma \nabla_{x_t} \log p(y \mid x_t) = \nabla_{x_t} \log \frac{1}{Z} p(y \mid x_t)^\gamma \tag{1.13}$$
>
> 其中 Z 是使得 $\frac{1}{Z} p(y \mid x_t)^\gamma$ 为概率分布的归一化常数。因此，实际上采样的分类器就变为了 $\frac{1}{Z} p(y \mid x_t)^\gamma$ ，这显然比 $p(y \mid x_t)$ 的分类效果要好，因为其分布会更加“尖锐”，对应的梯度就会越大，进而漂移项条件的影响就会更大。

但是 Z 是和 x_t 相关的， $Z = \int p(y \mid x_t)^\gamma dy$ ，离散标签的话 $Z = \sum_y p(y \mid x_t)^\gamma$ ，所以（1.13）式实际上**不成立**。

因为采样公式（1.10）只是它的一个一阶近似，所以为了分析改进后公式的有效性，我们需要重新回到公式（1.4）。原文说明了（1.4）也能写成（1.11）的形式，即条件生成分布 $p(x_{t-1} \mid x_t, y)$ 正比于无条件的采样分布 $p(x_{t-1} \mid x_t)$ 乘以 $e^{\log p(y \mid x_{t-1})}$ ，其中，分类器 $p(y \mid x_{t-1})$ 的含义是若 x_{t-1} 能很好的对应给定的条件 y ，则值为很大的数，若 x_{t-1} 不能很好的对应给定的 y ，则值会接近于0；所以 $\log p(y \mid x_{t-1})$ 会分别趋向大正数和负无穷。因此为了逼近这个分类器，可以直接定义一个相似函数 $F(x_{t-1}, y)$ 来代替 $\log p(y \mid x_{t-1})$ ，若 x_{t-1} 与 y 相似，则值会很大，反之则很小。并且也提到过，如果使这样的分布更加“尖锐”，那么效果应该会更好，所以使用放缩因子 $\gamma > 1$ 来控制，因此最终可以直接定义：

$$\begin{aligned} p(x_{t-1} \mid x_t, y) &\propto p(x_{t-1} \mid x_t) e^{\gamma F(x_{t-1}, y)} \\ &= \frac{p(x_{t-1} \mid x_t) e^{\gamma F(x_{t-1}, y)}}{Z(x_t, y)} \end{aligned} \tag{1.14}$$

其中 $Z(x_t, y) = \int p(x_{t-1} \mid x_t) e^{\gamma F(x_{t-1}, y)} dx_{t-1}$ 。最后仿照之前的推导，将 $F(x_{t-1}, y)$ 在点 (x_t, t) 处泰勒展开，最后就能得到：

$$p_\theta(x_{t-1} \mid x_t, y) = N(\mu_\theta(x_t) + \gamma \Sigma_\theta \nabla_{x_t} F(x_t, y), \Sigma_\theta) \tag{1.15}$$

一般的，可以令：

$$F(x_{t-1}, y) = \tilde{E}(x_{t-1}) \cdot E(y) \tag{1.16}$$

其中 E 是CLIP[3]编码器（能够提取多模态特征）， $\tilde{E}$ 是对含噪声信息微调后的CLIP编码器，“·”为内积，具体的细节在论文中有详细[4]说明。

Classifier-Guidance的这两种方法，本质上都是对分类器 $p(y \mid x_{t-1})$ 的近似，从而估计出 $p(x_{t-1} \mid x_t, y)$ 。所以如果能更好的对 $p(y \mid x_{t-1})$ 或者直接对 $p(x_{t-1} \mid x_t, y)$ 进行理论上更精确的逼近，例如高次展开或者变分法等等，最后可能会得到更好的结果。

但是，如果对于DDIM或者概率流ODE的情况，此时采样方差为0，那么式（1.10）和（1.15）就失效了。因此有必要使用SDE的工具重新进行分析。

根据[2]，前向扩散过程可以用以下SDE进行描述：

$$dx = f(x_t, t)dt + g(t)dw \tag{1.17}$$

又根据[5]，更一般的逆向生成过程为：

$$dx = \left(f(x_t, t) - \frac{1}{2} \left(g^2(t) + \sigma^2(t) \right) \nabla_{x_t} \log p(x_t) \right) dt + \sigma(t) dw \tag{1.18}$$

其中 $\sigma(t)$ 是自由变量，我们可以自由的选取合适的 $\sigma(t)$ 使得DDPM、DDIM都是它的特例。我们需要将条件 y 加入到采样过程（1.18）中去，因此只需要将 $\nabla_{x_t} \log p(x_t)$ 替换为 $\nabla_{x_t} \log p(x_t \mid y)$ 即可：

$$dx = \left(f(x_t, t) - \frac{1}{2} \left(g^2(t) + \sigma^2(t) \right) \nabla_{x_t} \log p(x_t \mid y) \right) dt + \sigma(t) dw \tag{1.19}$$

简单来说，若（1.17）描述条件前向过程 $p(x_t \mid x_{t-1}, y)$ ，其对应的无条件逆向过程为（1.19），而其又与无条件前向过程 $p(x_t \mid x_{t-1})$ 是等价的，所以（1.17）也对应了无条件的前向过程，因此（1.17）无条件前向过程对应的条件逆向过程则为（1.19），论文[6]附录的 I 部分

$$\nabla_{x_t} \log p(x_t \mid y) = \nabla_{x_t} \log p(x_t) + \nabla_{x_t} \log p(y \mid x_t) \tag{1.20}$$

而根据[7]的（3）到（5）式可知，$\nabla_{x_t} \log p(x_t)$ 又等价 $\nabla_{x_t} \log p(x_t \mid x_0)$，而：

$$\begin{aligned}\nabla_{x_t} \log p(x_t \mid x_0) &= -\frac{x_t - \sqrt{\bar{\alpha}_t} x_0}{1 - \bar{\alpha}_t} \\ &= -\frac{\varepsilon(x_t, t)}{\sqrt{1 - \bar{\alpha}_t}}\end{aligned} \tag{1.21}$$

> 此结论在得分匹配算法中会经常用到，常令：
>
> $$\nabla_{x_t} \log p_\theta(x_t) = \nabla_{x_t} \log p_\theta(x_t \mid x_0) = -\frac{\varepsilon_\theta(x_t, t)}{\sqrt{1 - \bar{\alpha}_t}} \tag{1.22}$$

所以（1.20）：

$$\begin{aligned}\nabla_{x_t} \log p(x_t \mid y) &= -\frac{\varepsilon(x_t, t)}{\sqrt{1 - \bar{\alpha}_t}} + \nabla_{x_t} \log p(y \mid x_t) \\ &= -\frac{\varepsilon(x_t, t) - \sqrt{1 - \bar{\alpha}_t} \nabla_{x_t} \log p(y \mid x_t)}{\sqrt{1 - \bar{\alpha}_t}}\end{aligned} \tag{1.22}$$

因此用训练好的 $\varepsilon_\theta(x_t, t)$ 替换 $\varepsilon(x_t, t)$，只需要额外训练分类器 $p_\phi(y \mid x_t)$ 替换 $p(y \mid x_t)$ 即可得到（1.22）式的估计，这就代表着可以用：

$$\hat{\varepsilon}_{\theta,\phi}(x_t, t) = \varepsilon_\theta(x_t, t) - \sqrt{1 - \bar{\alpha}_t} \nabla_{x_t} \log p_\phi(y \mid x_t) \tag{1.23}$$

来代替原来的 $\varepsilon_\theta(x_t, t)$，那么更一般的采样过程就变为了：

$$p(x_{t-1} \mid x_t, y) = N\left(\sqrt{\frac{\bar{\alpha}_{t-1}}{\bar{\alpha}_t}} x_t + \left(\sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2} - \sqrt{\frac{\bar{\alpha}_{t-1}(1 - \bar{\alpha}_t)}{\bar{\alpha}_t}}\right) \hat{\varepsilon}_{\theta,\phi}(x_t, t), \sigma_t^2\right) \tag{1.24}$$

此时令 $\sigma_t = 0$ 就得到DDIM的采样结果了。利用SDE的工具，得到了更加一般化的Classifier-Guidance。

2 Classifier-Free

Classifier-Free也叫“**事前修改**”方案，即直接将条件 y 加入到训练过程，因此模型需要重新训练，最早出现在《Classifier-Free Diffusion Guidance》。对于Classifier-Free，我们依然关心条件采样过程 $p(x_{t-1} \mid x_t, y)$，只不过这里直接定义为：

$$p_\theta(x_{t-1} \mid x_t, y) = N(\mu_\theta(x_t, y), \sigma_t^2) \tag{2.1}$$

因此，此时的损失函数：

$$\mathcal{L} \propto \|\varepsilon(x_t, t) - \varepsilon_\theta(x_t, t, y)\|^2 \tag{2.2}$$

这个过程是非常好理解的，也就是给定样本对 (x_0, y)，经过DDPM的方式训练后，这时采样的模型 ε_θ 中就蕴含了 y 的“指引”，这样就能引导噪声向着 y 的方向去噪。

结论

无论是Classifier Guidance还是Classifier Free，本质都是让条件采样分布尽量向正态分布靠拢，Classifier Guidance利用泰勒展开，而Classifier Free直接定义。不过利用SDE的工具，似乎绕开了这一行为？相信对研究Conditional Control的扩散模型有更大的价值！

参考

1. ^ 概率视角下的生成模型 https://zhuanlan.zhihu.com/p/611466195
2. ^ a b Diffusion学习笔记（三）——随机微分方程（SDE） https://zhuanlan.zhihu.com/p/619188621
3. ^ Learning Transferable Visual Models From Natural Language Supervision https://arxiv.org/abs/2103.00020
4. ^ More Control for Free! Image Synthesis with Semantic Diffusion Guidance https://arxiv.org/abs/2112.05744
5. ^ Diffusion学习笔记（四）——概率流ODE（Probability flow ODE） https://zhuanlan.zhihu.com/p/622771940
6. ^ Score-based generative modeling through stochastic differential equations https://arxiv.org/abs/2011.13456

编辑于 2023-06-08 10:04 · IP 属地广东

Stable Diffusion　深度学习生成模型　扩散模型

▲ 赞同 26　▼　4 条评论　分享　喜欢　收藏　申请转载　···

写下你的评论...

4 条评论　默认　最新

蓝冰
😘😘😘
05-16 · IP 属地山东　回复　赞

麦哲
Classifier-Guidance看下来还是很需要耐心的😳，大佬数学功底真扎实啊，比如"二元泰勒展开"这种如果能放上一般性公式作为对比就好很多了哈哈哈😄
05-12 · IP 属地北京　回复　赞

cz.y
泰库辣
04-28 · IP 属地广东　回复　赞

Hammour Yue 作者
如果你也能像我一样
04-28 · IP 属地广东　回复　赞