Seminário 1 - Diffusion Priors In Variational Autoencoders

Leonardo Belo

Abril 2025

Contexto

Dentre as abordagens baseadas em verossimilhança para modelagem generativa, *Variational Autoencoders* (VAEs) oferecem de modo escalável a inferência da posteriori amortizada.

Contudo, acabaram sendo muito superados por modelos generativos como normalização de fluxo. Como um dos motivos destaca-se que *Variational Autoencoders* (VAEs) são modelos generativos que utilizam uma distribuição prior simples, geralmente Gaussiana, para modelar as variáveis latentes.

No entanto, essa simplicidade pode limitar a capacidade do VAE em capturar distribuições de dados mais complexas.

Proposta do artigo

O artigo propõe substituir o prior Gaussiano tradicional por um Diffusion Prior, utilizando Denoising diffusion probabilistic models (DDPMs) para modelar a distribuição das variáveis latentes. Essa abordagem visa aumentar a expressividade do prior, permitindo que o VAE capture melhor a complexidade dos dados.

Contribuições:

- **Integração de DDPMs como Prior**: O DDPM captura distribuições complexas (multimodais, com dependências não-lineares), enquanto priors Gaussianos são limitados.
- Espaço latente mais denso e estruturado
- **Resultados Competitivos**
- **Potencial em VAEs Hierárquicos**: Com muitas camadas latentes, cada uma pode ter uma prior DDPM.

Por que precisa de uma "boa" prior?

A qualidade das amostras geradas depende fortemente da prior definida no espaço latente, pois determina quais representações latentes são mais prováveis ou naturais antes mesmo do modelo observar qualquer dado específico.

É interessante buscar uma prior informativa que consegue modelar representações latentes mais ricas, que capturam estruturas complexas nos dados originais.

VAEs clássicos utilizam um prior relativamente simples, normalmente uma distribuição Gaussiana padrão. Prior simples (Gaussiano) pode gerar representações latentes ”vazias” ou pouco expressivas, reduzindo o realismo das amostras geradas.

Introdução

1) **Variational Autoencoders (VAEs)**

Os Variational Autoencoders (VAEs) são modelos generativos designados para capturar a distribuição de probabilidade subjacente a um determinado conjunto de dados, permitindo a geração de novas amostras.

Sua arquitetura consiste em uma estrutura **encoder-decoder**. O **encoder** transforma os dados de entrada em uma *representação latente* (comprimida), enquanto o **decoder** busca reconstruir os dados originais a partir dessa representação latente.

O VAE é treinado para minimizar a diferença entre os dados originais e os reconstruídos.

Introdução

2) Denoising diffusion probabilistic models (DDPM)

Geram dados por meio de um processo gradual, revertendo ruído em dados estruturados ao longo de vários passos. Ele envolve dois processos:

1. **Processo de difusão (forward diffusion)**
   - Recebe dado real e, progressivamente, adiciona-se ruído (aleatoriedade) aos dados originais em pequenas etapas, até que esses dados se tornem praticamente indistinguíveis de um ruído completamente aleatório.
   - No final, obtém-se algo semelhante a uma distribuição de puro ruído.
2. **Processo inverso (reverse diffusion ou denoising)**
   - Partindo de um ruído completamente aleatório, o modelo aprende a inverter esse processo.
   - O modelo tenta gradualmente remover ruído passo-a-passo, recuperando a estrutura e as características dos dados originais.
   - Ao final, obtém-se uma amostra totalmente nova que é semelhante aos dados que o modelo foi treinado para gerar.

Conceitos matemáticos - VAE

1) Formulação VAEs

No VAE assumimos que x são gerados a partir de um espaço latente z, onde há uma prior ($p(z)$) e um decoder ($p(x|z)$) que reconstroi x a partir de z. Dessa forma, temos o desafio de aprender $p(z|x)$ que pode ser difícil, sendo mais fácil aprender um encoder aproximado $q(z|x)$.

Para isso é necessário aprender $p(x)$, pois queremos aprender um modelo que gere amostras que sejam estatisticamente semelhantes aos dados reais. Para isso, busca-se maximizar $logp(x) = log \int p(x|z)p(z)dz$. Devido a dificuldade do problema, aplica-se *inferência variacional* para maximizar o limite inferior ELBO da log-verossimilhança.

Conceitos matemáticos - VAE

2) ELBO

Usamos um encoder $q(z|x)$ para aproximar $p(z|x)$, assim

$$logp(x) = E_q[logp(z|x)] - KL(q||p(z)) + KL(q||p(z|x))$$

, sendo $KL(Q||P) = \int Q(x) log\frac{Q(x)}{P(x)} dx$.

Como $KL(q||p(z|x)) \geq 0$, obtemos o **Evidence Lower Bound (ELBO)**

$$ELBO = E_q[logp(z|x)] - KL(q||p(z))$$

.

Conceitos matemáticos - VAE

3) Maximização ELBO

Para o termo $E_q[logp(z|x)]$, precisamos medir o quão bem $p(x|z)$ pode reconstruir os dados a partir da representação latente z. Geralmente, é modelado como $X|Z \sim N(x; NN_\theta(z), \sigma^2 I)$, onde $NN_\theta(z)$ é a rede neural parametrizada por θ, assim recebe z e retorna a reconstrução $\hat{x}$

Para o termo KL, o qual mede a diferença aproximada de $q(z|x)$ em relação a $p(z)$, muitas vezes é assumido *prior Gaussiana*. Por exemplo, se assumirmos normal padrão, temos $KL(q||p(z)) = \frac{1}{2}\sum_{i=1}^{d}(1 + log\sigma_i^2 - \mu_i - \sigma_i^2)$, sendo o conjunto $\{(\mu_i, \sigma_i^2)\}_{i=1}^{d}$ oriundos da distribuição de X|Z.

Conceitos matemáticos - VAE

4) Reparametrização

Em vez de amostrar diretamente $q(z|x)$ como $N(\mu, \sigma^2)$, reescrevemos $z = \mu + \sigma \odot \epsilon, \epsilon \sim N(0, I)$.

Isso transforma a amostragem em uma operação diferenciável, permitindo backpropagation.

Conceitos matemáticos - DDPM

DDPMs atuam como a operação de difusão reversa. Formalmente, o processo de reversão é um modelo de variável latente da forma $p_\phi(x_0) = \int p_\phi(x_{0:T})dx_{1:T}$, onde $x_0 = x$ denota a observação e $x_1, .., x_T$ denota variáveis latentes de mesma dimensão.

A distribuição conjunta $p_\phi(x_{0:T})$ é modelada por uma cadeia de Markov de primeira ordem com transições gaussianas, ou seja

$$p_\phi(x_{0:T}) = p_\phi(x_T)\Pi_{t=1}^{T} p_\phi(x_{t-1}|x_t),$$

$$p_\phi(x_T) = N(0, I),$$

$$p_\phi(x_{t-1}|x_t) = N(\mu_\phi(x_t, t), \sigma_t^2 I).$$

Conceitos matemáticos - DDPM

A posterior aproximada é fixada para um processo de difusão que é também uma cadeia de Markov de primeira ordem com transições Gaussianas

$$q(x_{1:T}|x_0) = \Pi_{t=1}^{T} q(x_t|x_{t-1})$$

$$q(x_t|x_{t-1}) = N(\sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

, onde $\beta_1, ..., \beta_T$ são os esquemas de variância que podem ser fixados como hiperparâmetros de treinamento ou aprendidos.

O ELBO é então dado por

$$ELBO = E_q[log\frac{p_\phi(x_{0:T})}{q(x_{1:T}|x_0)}] \leq logp_\phi(x_0)$$

Conceitos matemáticos - DDPM

Assumindo a variância dos termos β_t é pequena e T é grande o suficiente, as hipóteses Gaussiana no processo generativo p_ϕ são razoáveis. Usando as transições Gaussianas podemos expressar ELBO como:

$$E_q[KL[q(x_T|x_0)||p(x_T)] - logp_\phi(x_0|x_1) + \sum_{t=2}^{T} KL[q(x_{t-1}|x_t, x_0)||p_\phi(x_{t-1}|x_t)]].$$

Com isso, a posterior condicional de avanço $q(x_{t-1}|x_t, x_0)$ pode ser expressa de forma fecahada como Gaussianas $N(\tilde{\mu}_t(x_0, x_t), \tilde{\beta}_t I)$, onde $\tilde{\beta}_t$ são as funções do esquema de variação.
Então, o KL pode ser calculado de forma fechada e a função objetivo final é:

$$L_{DDPM}(x_0, \phi) := E_{t,x_0,x_t}[\frac{1}{2\sigma_t^2}||\mu_\phi(x_t, t) - \tilde{\mu}_t(x_0, x_t)||^2],$$

Modelando prior com DDPM

Formulação do modelo generativo:
$z_T \sim N(0, I), z_{t-1|t} \sim p_\phi(z_{t-1}|z_t), \forall t \in [1, 2, ..., T]$ e $x \sim p_\theta(x|z_0)$, onde ϕ denota os parâmetros do modelo de difusão reversa que codifica a distribuição prior.

Infelizmente, não dá para treinar VAE com prior difusão diretamente no ELBO, pois $p_\phi(z)$ não pode ser avaliada. Contudo, podemos reescrever ELBO como:

$$E_{q_\psi}[logp_\theta(x|z_0)] - E_{q_\psi}[logq(z_0|x)] + E_{q_\psi}[logp_\phi(z_0)]$$

Finalmente, temos a expressão

$$E_{q_\psi}[logp_\theta(x|z_0) - logq(z_0|x) + E_q[log\frac{p_\phi(z_{0:T})}{q(z_{1:T}|z_0)}]]$$

$$\leq E_{q_\psi}[logp_\theta(x|z_0)] - logq(z_0|x) + logP_\phi(z_0)] \leq logp_\theta(x),$$

Modelando prior com DDPM

Finalmente, a prior de difusão p_ϕ é treinada conjuntamente com a aproximação da posterior q_ψ e os modelos de verossimilhança p_θ, os quais são otimizados por VAE clássico.

A função de perda (loss function) é:

$$\mathcal{L}(x;\phi,\theta,\psi) = E_{q_\psi}[log\frac{p_\theta(x|z)}{q_\psi(z|x)}] + E_{q_\psi}[L_{DDPM}(z_0;\phi)]$$

Arquitetura

O artigo usa arquitetura DCGAN - (Deep Convolutional Generative Adversarial Network (Radford et al., 2015), a qual possui as seguintes características:

- Uso de camadas convolucionais e transpostas
- Normalização por Lote (BatchNorm) - garante estabilidade no treinamento
- Em relação as funções de ativação, são usadas no Encoder LeakyReLU e no Decoder ReLU (exceto no final que é usado Tanh).

Além disso, são implementados:

- Time step: Sinusoidal
- Cosine Beta Schedule - Similar ao Improved DDPM (Nichol Dhariwal, 2021).
- KL Annealing no Peso Latente: regularizar de forma gradual o impacto do termo de divergência na função de perda - Bowman et al. (2015).

Arquitetura DCGAN original

Architecture guidelines for stable Deep Convolutional GANs

- Replace any pooling layers with strided convolutions (discriminator) and fractional-strided convolutions (generator).
- Use batchnorm in both the generator and the discriminator.
- Remove fully connected hidden layers for deeper architectures.
- Use ReLU activation in generator for all layers except for the output, which uses Tanh.
- Use LeakyReLU activation in the discriminator for all layers.

Figure: Diferenças do DCGAN para o original

Código

Github

Experimentos - artigo

O artigo comparou os desempenhos de VAEs com diferentes prior: Gaussiana, NF e Difusão.

Os conjuntos de dados usados foram CIFAR10 e CelebA com três variáveis latentes diferentes dimensionalmente (40, 100, 200)

Condições de treinamento:

- Optimization: Adam
- epochs = 250
- learning rate = 5e-4
- Treinamento do DDPM e VAE é conjunto
- Método de avaliação: FID scores

Percepção e resultados segundo os autores

- Os 3 modelos performaram bem em CIFAR10, provavelmente, devido a simplicidade.
- Apesar do FID scores sugerir que prior Gaussiana supera prior de difusão no CIFAR10, as imagens usando prior de difusão parecem mais realísticas.
- Prior NF obteve o melhor FID score, mas tal diferença de superioridade não é refletido nas imagens.
- Por esse motivo, os autores acreditam que os valores do FID não sejam totalmente informativos sobre a qualidade das imagens sintetizadas pelos modelos e devem ser interpretados com cautela.
- Prior de difusão supera prior Gaussiana no conjunto CelebA

Percepção e resultados segundo os autores

Table 1. FID scores for different models for prior modelling in VAEs and for different latent size. *Diffusion priors outperform classical VAE on CelebA but are slightly worse than NFs. FID scores do not reveal the superiority of any method on CIFAR10.*

Dataset	**CelebA**			**CIFAR10**		
Latent Size	40	100	200	40	100	200
Gaussian	154.3	149.4	139.1	176.0	126.2	123.9
NF	72.9	59.49	54.7	167.6	129.1	129.6
Diffusion	114.8	67.95	88.3	177.9	160.5	153.1

Figure: Table 1. FID scores for different models for prior modelling in VAEs and for different latent size. Diffusion priors outperform classical VAE on CelebA but are slightly worse than NFs. FID scores do not reveal the superiority of any method on CIFAR10.

Referências

1. Diffusion Priors In Variational Autoencoders - Antoine Wehenkel & Gilles Louppe
2. Ho, J., Jain, A., and Abbeel, P. Denoising diffusion proba bilistic models. arXiv preprint arXiv:2006.11239, 2020
3. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in Neural Information Processing Systems, pages 5998–6008, 2017.
4. Improved Denoising Diffusion Probabilistic Models - Alex Nichol and Prafulla Dhariwal (2021)
5. UNSUPERVISED REPRESENTATION LEARNING WITH DEEP CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS - Alec Radford, Luke Metz & Soumith Chintala (2016)

Referências

6. Generating Sentences from a Continuous Space - Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Jozefowicz & Samy Bengio (2016)
7. Understanding Posterior Collapse in Generative Latent Variable Models - James Lucas, George Tucker, Roger Grosse & Mohammad Norouzi (2019)

Perguntas?