1prompt-1story

https://arxiv.org/pdf/2501.13554*

February 16, 2025

1 问题背景

作者在论文中提出了一个名为 **One-Prompt-One-Story (1Prompt1Story)** 的方法，旨在解决文本到图像生成（Text-to-Image Generation, T2I）中的一致性问题。具体来说，现有的 T2I 模型在生成图像时，难以保持跨多个场景的 **主体一致性**（subject consistency），尤其是在需要生成一系列连贯的图像时（如动画、故事叙述等）。现有的方法通常需要大量的训练数据或对模型架构进行复杂的修改，限制了其在不同领域和不同扩散模型配置中的适用性。作者希望通过一种无需训练的方法，利用语言模型的上下文一致性（context consistency）来实现一致性的文本到图像生成。作者在论文中主要介绍了 **Naive Prompt Reweighting**、**Singular-Value Reweighting** 和 **Identity-Preserving Cross-Attention** 三种方法，接下来我们依次介绍一下。

2 Naive Prompt Reweighting

Naive Prompt Reweighting（朴素提示重加权）是一种简单的文本嵌入调整方法，用于在单提示生成设置下生成具有一致身份的图像。具体来说，这种方法通过对文本嵌入进行加权调整，来增强当前帧提示的表达，同时抑制其他帧提示的影响。

Naive Prompt Reweighting 的具体步骤：

1. **提示拼接 (Prompt Consolidation)**：
   - 首先，将所有的提示（身份提示和多个帧提示）拼接成一个长句子，形成一个单提示。例如，身份提示是 “一只可爱的水彩小猫”，帧提示是 “在花园里”、“穿着超级英雄斗篷” 等，拼接后的单提示为：“一只可爱的水彩小猫，在花园里，穿着超级英雄斗篷，戴着带铃铛的项圈，坐在篮子里，穿着可爱的毛衣”。
2. **文本嵌入生成**：
   - 使用文本编码器（如 CLIP）将拼接后的单提示转换为文本嵌入。假设生成的文本嵌入为 $C = [c^{SOT}, c^{P_0}, c^{P_1}, \ldots, c^{P_N}, c^{EOT}]$，其中 c^{SOT} 和 c^{EOT} 分别是开始和结束标记的嵌入，c^{P_0} 是身份提示的嵌入，$c^{P_1}, \ldots, c^{P_N}$ 是帧提示的嵌入。
3. **重加权操作**：

*https://github.com/xinli2008

- 为了生成第 i 帧的图像，作者对文本嵌入进行重加权操作：
  - 增强当前帧提示：将当前帧提示 c^{P_i} 的嵌入乘以一个放大因子（例如 2），以增强其在生成过程中的表达。
  - 抑制其他帧提示：将其他帧提示 c^{P_k} $(k \neq i)$ 的嵌入乘以一个缩小因子（例如 0.5），以抑制它们在生成过程中的影响。
  - 保持结束标记不变：结束标记 c^{EOT} 的嵌入保持不变。

4. **图像生成**：
   - 将重加权后的文本嵌入输入到文本到图像（T2I）扩散模型中，生成第 i 帧的图像。

3 Singular-Value Reweighting (SVR)

在论文的 **3.2 One-Prompt-One-Story** 章节中，**Singular-Value Reweighting (SVR)** 是一种更高级的文本嵌入调整技术，用于在单提示生成设置下进一步改进图像生成的质量。与 **Naive Prompt Reweighting** 相比，SVR 通过奇异值分解（SVD）来更精确地控制文本嵌入的表达和抑制，从而在生成图像时更好地保持身份一致性并减少背景混合问题。

3.1 Singular-Value Reweighting (SVR) 的具体步骤

3.1.1 提示拼接（Prompt Consolidation）

首先，将所有的提示（身份提示和多个帧提示）拼接成一个长句子，形成一个单提示。例如，身份提示是 “一幅可爱的水彩小猫”，帧提示是 “在花园里”、“穿着超级英雄斗篷” 等，拼接后的单提示为：

> “一幅可爱的水彩小猫，在花园里，穿着超级英雄斗篷，戴着带铃铛的项圈，坐在篮子里，穿着可爱的毛衣”。

3.1.2 文本嵌入生成

使用文本编码器（如 CLIP）将拼接后的单提示转换为文本嵌入。假设生成的文本嵌入为：

$$\mathcal{C} = [c^{SOT}, c^{\mathcal{P}_0}, c^{\mathcal{P}_1}, \ldots, c^{\mathcal{P}_N}, c^{EOT}],$$

其中：

- c^{SOT} 和 c^{EOT} 分别是开始和结束标记的嵌入。
- $c^{\mathcal{P}_0}$ 是身份提示的嵌入。
- $c^{\mathcal{P}_1}, \ldots, c^{\mathcal{P}_N}$ 是帧提示的嵌入。

3.1.3 奇异值分解（SVD）

为了生成第 j 帧的图像，作者将文本嵌入分为两个部分：

- **表达集（Express Set）**：包含当前帧提示 $\mathcal{P}_j$ 和结束标记 c^{EOT} 的嵌入，记为：

$$\mathcal{X}^{exp} = [c^{\mathcal{P}_j}, c^{EOT}].$$

- **抑制集（Suppress Set）**：包含其他帧提示 $\mathcal{P}_k$（$k \neq j$）和结束标记 c^{EOT} 的嵌入，记为：

$$\mathcal{X}^{sup} = [c^{\mathcal{P}_1}, \ldots, c^{\mathcal{P}_{j-1}}, c^{\mathcal{P}_{j+1}}, \ldots, c^{\mathcal{P}_N}, c^{EOT}].$$

3.1.4 增强表达集（SVR+）

对表达集 $\mathcal{X}^{exp}$ 进行奇异值分解（SVD），得到：

$$\mathcal{X}^{exp} = U\Sigma V^T,$$

其中 $\Sigma = \text{diag}(\sigma_0, \sigma_1, \ldots, \sigma_{n_j})$ 是奇异值矩阵。

通过以下公式增强奇异值：

$$\hat{\sigma} = \beta e^{\alpha\sigma} * \sigma,$$

其中 α 和 β 是正数参数，σ 是原始奇异值，$\hat{\sigma}$ 是增强后的奇异值。

使用增强后的奇异值矩阵 $\hat{\Sigma} = \text{diag}(\hat{\sigma}_0, \hat{\sigma}_1, \ldots, \hat{\sigma}_{n_j})$ 重构表达集的嵌入：

$$\hat{\mathcal{X}}^{exp} = U\hat{\Sigma}V^T.$$

3.1.5 抑制抑制集（SVR-）

对抑制集 $\mathcal{X}^{sup}$ 中的每个帧提示 $\mathcal{P}_k$（$k \neq j$）单独进行奇异值分解，得到：

$$\mathcal{X}_k^{sup} = [c^{\mathcal{P}_k}, c^{EOT}].$$

通过以下公式抑制奇异值：

$$\tilde{\sigma} = \beta' e^{-\alpha'\sigma} * \sigma,$$

其中 α' 和 β' 是正数参数，σ 是原始奇异值，$\tilde{\sigma}$ 是抑制后的奇异值。

使用抑制后的奇异值矩阵重构抑制集的嵌入：

$$\hat{\mathcal{X}}_k^{sup} = [\hat{c}^{\mathcal{P}_k}, \hat{c}^{EOT}].$$

3.1.6 更新文本嵌入

将增强后的表达集嵌入和抑制后的抑制集嵌入组合成新的文本嵌入：

$$\tilde{\mathcal{C}} = [c^{SOT}, c^{\mathcal{P}_0}, \hat{c}^{\mathcal{P}_1}, \ldots, \hat{c}^{\mathcal{P}_j}, \ldots, \hat{c}^{\mathcal{P}_N}, \hat{c}^{EOT}].$$

3.1.7 图像生成

将更新后的文本嵌入输入到文本到图像（T2I）扩散模型中，生成第 j 帧的图像。

4 Identity-Preserving Cross-Attention (IPCA)

在文本到图像 (T2I) 扩散模型中，交叉注意力机制 (Cross-Attention) 是连接文本嵌入和图像生成过程的关键组件。交叉注意力机制通过计算文本嵌入和图像特征之间的注意力权重，来决定图像生成过程中哪些文本信息应该重点关注。

然而，在单提示生成设置下，由于多个帧提示被拼接成一个长句子，交叉注意力机制可能会受到其他帧提示的干扰，导致生成图像的身份一致性下降。为了解决这个问题，作者提出了 Identity-Preserving Cross-Attention (IPCA)，通过调整交叉注意力机制来增强身份提示的表达，同时抑制其他帧提示的影响。

1. **输入准备**：
   - 在生成第 j 帧图像时，假设已经通过 Singular-Value Reweighting (SVR) 更新了文本嵌入 $\tilde{\mathcal{C}} = [c^{SOT}, c^{P_0}, \ldots, c^{P_N}, c^{EOT}]$。
   - 在扩散模型的交叉注意力层中，文本嵌入 $\tilde{\mathcal{C}}$ 被拆分为键矩阵 $\tilde{\mathcal{K}}$ 和值矩阵 $\tilde{\mathcal{V}}$。
2. **过滤提示的键和值**：
   - 为了增强身份提示的表达并抑制其他帧提示的影响，作者将键矩阵 $\tilde{\mathcal{K}}$ 和值矩阵 $\tilde{\mathcal{V}}$ 中与帧提示 P_i ($i \in [1, N]$) 对应的部分设为零向量。具体来说：
     - 对于键矩阵 $\tilde{\mathcal{K}}$，将与帧提示 P_i 对应的列设为零向量，得到过滤后的键矩阵 $\mathcal{K}$。
     - 对于值矩阵 $\tilde{\mathcal{V}}$，同样将与帧提示 P_i 对应的列设为零向量，得到过滤后的值矩阵 $\mathcal{V}$。
   - 这样的目的是确保在交叉注意力计算中，只有身份提示 P_0 的语义信息被保留，而其他帧提示的语义信息被抑制。
3. **构建新的键和值矩阵**：
   - 将过滤后的键矩阵 $\mathcal{K}$ 和初始的键矩阵 $\tilde{\mathcal{K}}$ 进行拼接，得到新的键矩阵：
   $$\hat{\mathcal{K}} = [\mathcal{K}; \tilde{\mathcal{K}}]$$
   - 同样，将过滤后的值矩阵 $\mathcal{V}$ 和初始的值矩阵 $\tilde{\mathcal{V}}$ 进行拼接，得到新的值矩阵：
   $$\hat{\mathcal{V}} = [\mathcal{V}; \tilde{\mathcal{V}}]$$
4. **计算新的交叉注意力图**：
   - 使用新的键矩阵 $\hat{\mathcal{K}}$ 和查询矩阵 $\tilde{\mathcal{Q}}$ 计算新的交叉注意力图：
   $$\hat{\mathcal{A}} = \text{softmax}\left(\frac{\tilde{\mathcal{Q}}\hat{\mathcal{K}}^\top}{\sqrt{d}}\right)$$
   其中 d 是查询和键矩阵的维度。
5. **生成输出特征**：
   - 使用新的交叉注意力图 $\hat{\mathcal{A}}$ 和新的值矩阵 $\hat{\mathcal{V}}$ 计算输出特征：
   $$\text{Output} = \hat{\mathcal{A}} \times \hat{\mathcal{V}}$$
   这个输出特征是经过重新加权的，只包含身份提示的语义信息，从而增强了生成图像的身份一致性。