

Projeto de Inteligência Computacional

Diffusion For Text To Image Generation : InstaFlow

João Cláudio Paco, M13709

Estagiário de Investigação em Universidade Kimpa-Vita em Angola

2024

Sumário

1	Introdução	3
2	Base de dados	3
3	Métricas de avaliação	4
3.1	Tabela 1 e 2: Comparação de pontuação de(a) FID e CLIP no MS COCO 2017 e (b) FID no MS COCO 2014	4
3.2	Legenda:	4
3.3	Descrição:	5
3.4	Resultados Experimentais:	5
4	Um dos dois métodos recentes publicados	5
4.1	Origem:	5
4.2	Funcionamento	5
4.3	Aspetos comparativos:	6
5	Avaliação comparativa	6
6	Implementação de algumas alterações aos códigos	9
6.1	Códigos originais:	9
6.1.1	Descrição:	9
6.1.2	Problema:	9
6.2	Códigos alterados:	12
6.2.1	Descrição:	12
6.2.2	Resultados:	12
7	Conclusão	12

1 Introdução

Na era digital em que vivemos, a interseção entre linguagem e imagem tem sido uma área de interesse crescente, impulsionada pela demanda por soluções inovadoras em áreas como design gráfico, arte digital, e até mesmo e, aplicações práticas como geração automática de conteúdo visual para acompanhar textos. Neste contexto, a geração de imagens a partir de textos tem emergido como uma área de pesquisa dinâmica e promissora.

Os modelos de difusão revolucionaram a **geração de texto para imagem** com sua qualidade e criatividade excepcionais. No entanto, sabe-se que seu processo de amostragem em múltiplas etapas é lento, muitas vezes exigindo dezenas de etapas de inferência para obter resultados satisfatórios. Tentativas anteriores de melhorar a velocidade de amostragem e reduzir custos computacionais através da destilação não tiveram sucesso na obtenção de um modelo funcional de *one-step*. Neste projeto, exploramos um método recente chamado *RectifiedFlow*, que, até agora, só foi aplicado a pequenos conjuntos de dados. O núcleo do *RectifiedFlow* está em seu procedimentoreflow. que endireita as trajetórias dos fluxos de probabilidade, refina o acoplamento entre ruídos e imagens e facilita o processo de destilação com *studentmodels*. Foi proposto um novo pipeline condicionado por texto para transformar a *StableDiffusion (SD)* em um modelo ultra rápido de *one-step*, no qual foi descoberto que o refluxo desempenha um papel crítico na melhoria da atribuição entre ruído e imagens.

A escolha deste tema se fundamenta na importância crescente da síntese de imagens baseada em texto, não apenas em termos de criatividade e expressão artística, mas também em sua utilidade prática em campos como *design* gráfico automatizado, criação de conteúdo para mídias sociais. Foi escolhido o método ***InstaFlow : One Step is Enough for High-Quality Diffusion-Based Text-to-Image Generation*** super rápido que a técnica de última geração anterior, destilação progressiva, por uma margem significativa (em FID), notavelmente, o treinamento do *InstaFlow* custa apenas 199 dias de GPU A 100. Códigos e modelos pré-treinados estão disponíveis em [url\(github.com/gnobitab/InstaFlow\)](https://github.com/gnobitab/InstaFlow).

Ao final deste estudo, esperamos não apenas oferecer *insights* valiosos para a comunidade acadêmica e profissional interessada neste campo, mais também destacar possíveis direções futuras para pesquisas adicionais, visando aprimorar ainda mais a capacidade de gerar imagens de alta qualidade a partir de descrições textuais.

2 Base de dados

Para descrever as bases de dados usadas para explorar o tema "*Diffusion for text to Image Generation*" com base no método *InstaFlow*, consideramos conjuntos de dados relevantes para treinar e avaliar modelos de geração de imagens a a partir de texto usando técnicas de difusão, e para diversas outras pesquisas. Aqui estão algumas bases de dados utilizadas para esse fim e seus *links*:

- *COCO (Common Objects in Context)*: o conjunto de dados COCO contém uma grande coleção de imagens naturais em diferentes contextos, com anotações detalhadas, como objetos presentes, segmentação e descrições de texto associa.

<https://cocodataset.org>

- *Joint the Hugging face Community* :

<https://github.com/CompVis/stable-diffusion>

<https://huggingface.co/docs/diffusers/training/text2image>

- ArXiv:
<https://arxiv.org/abs/2211.15388>
- *deepai.org*:
AI Chat (deepai.org)

3 Métricas de avaliação

A tabela abaixo mostra como são avaliados os resultados dum dos testes experimentais realizados sobre o *Diffusion for Text to Image Generation* considerando o método *InstaFlow*:

3.1 Tabela 1 e 2: Comparação de pontuação de(a) FID e CLIP no MS COCO 2017 e (b) FID no MS COCO 2014

1. Tabela 1 pontuação de(a) FID e CLIP no MS COCO 2017

Method	Inf. Time	FID-5k	CLIP
SD 1.4(25 step)[70]	0.88s	22.8	0.315
(1)(Pre)2-RF(25 step)	0.88s	22.1	0.313
PD(1 step)[58]	0.09s	37.2	0.275
SD 1.4+Distill	0.09s	40.9	0.255
(Pre)2-RF(1 step)	0.09s	68.3	0.252
(2)(Pre)2-RF+Distill	0.09s	31.0	0.285

Tabela 1: (a) MS COCO 2017

2. Tabela 2 pontuação de (b) FID no MS COCO 2014

Method	Inf. Time	FID-30k
SD*[70]	0.2.9s	9.62
(3)(Pre)2-RF(25 step)	0.88s	13.4
SD 1.4+Distill	0.09s	34.6
((4)Pre)2-RF+Distill	0.09s	20.0

Tabela 2: (b) MS COCO 2014

3.2 Legenda:

1. FID: refere-se a "*FrechetInceptionDistance*", uma métrica de avaliação de qualidade de imagens em comparação com um conjunto de dados de referência. Valores menores de FID indicam uma melhor qualidade das imagens geradas.
2. CLIP: refere-se a "*ContrastiveLanguage-ImagePretrainig*", uma técnica de aprendizado de máquina que associa imagens e texto.
3. SD: como "*StableDiffusion*" é um método ou modelo no contexto de geração de imagens a partir do texto.

4. Inf. Time: "*Inference Time*", que se refere ao tempo necessário para realizar uma inferência ou uma previsão com o modelo.
5. PD: refere-se a "*ProgressiveDistillation*"

3.3 Descrição:

Comparação de(a) pontuação FID e CLIP no MS COCO 2017 com 5.000 imagens seguindo a configuração de avaliação em [58] e (b) FID no MS COCO 2014 com 30.000 imagens seguindo a configuração de avaliação em [2], o tempo de inferência é medido na GPU NVIDIA A 100 com um tamanho de lote de 1. "**Pré**" é adicionado para distinguir os modelos da Tabela 2. "**RF**" refere-se ao Fluxo Reticado; "PD" refere-se à destilação progressiva[1] * denota que os números são médios.

3.4 Resultados Experimentais:

(1) O **(Pre) 2-RectifiedFlow** pode gerar(5k-5000) imagens realistas que produzem FID semelhante de (22,1- 22.3) com SD 1.4 usando 25 step dentro de uma *Inf. Time* de 0.88s.

(2) O **(Pre) 2-Rectified Flow+Distill** obtém um FID de 31.0, com um SD 1.4+Distill, superando o melhor modelo SD de uma etapa anterior(FID=37.2) da **Progressive Distillation** com muito menos custo de treinamento(1 step), dentro de um *Inf. Time* de 0.9s.

(3) **(Pre) 2 Rectified Flow+Distill** tem vantagem notável(FID = 20.0) em comparação com *directed distillation* SD 1.4 + Distill(FID=34.6)

(4) **(Pre) "Rectified Flpaw** tem pior desempenho(FID=13.4) do que o SD original(FID=9.62) devido a insuficiência de treinamento, indicando a eficácia da operação de reFlow.

4 Um dos dois métodos recentes publicados

Corgi (*Compositional Relevance Guided Image Generation*)

4.1 Origem:

O nome "Corgi" vem da raça de cachorro, mas na verdade é um acrônimo para "*Compositional Relevance-Guided Image Generation*". O método *Corgi* é um algoritmo de geração de imagens a partir de texto desenvolvido pela *OpenAI*. O método CORGI foi publicado em 2022. A pesquisa sobre método, intitulada "*Shifthead Diffusion for Text-to-Image Generation*", foi disponibilizada no *arXiv* em novembro de 2022.

4.2 Funcionamento

Esse método busca criar imagens que sejam semanticamente relevantes ao texto de entrada, ou seja, que capturem de forma precisa e relevante os elementos descritos no texto. Ele utiliza técnicas avançadas de inteligência artificial, especialmente modelos de linguagem como GPT, juntamente com redes neurais convolucionais para traduzir descrições textuais

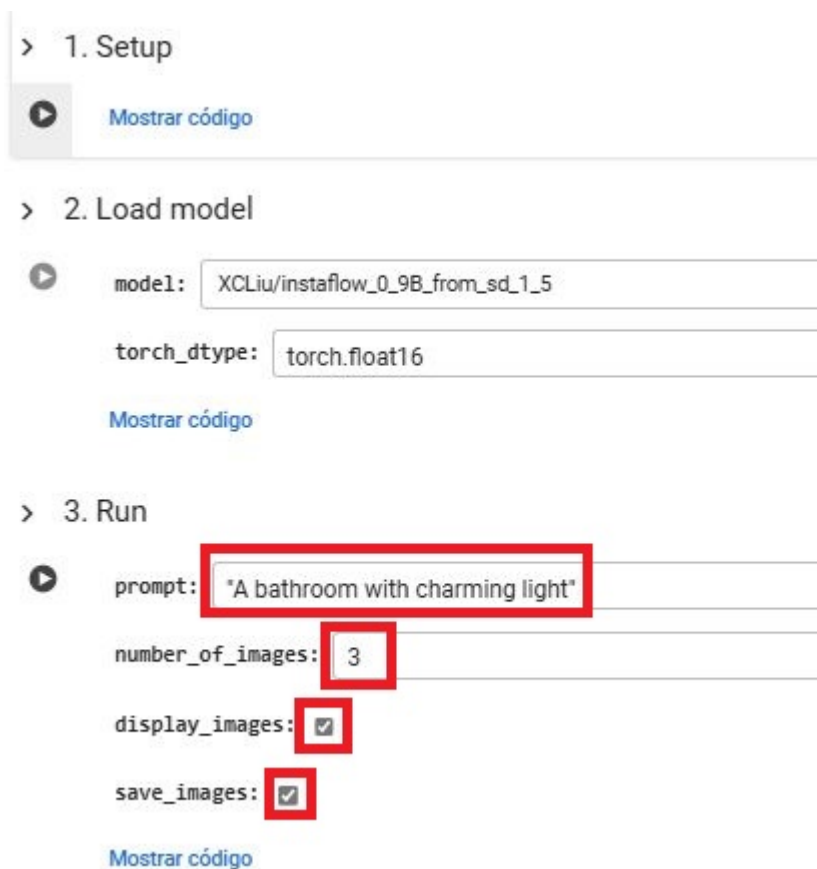
em imagens correspondentes. O Corgi, um novo método para geração de imagem a partir de texto como avança *arXiv2211.15388* é baseado em modelo, o qual alcança uma melhor geração de incorporação de imagem a partir do texto de entrada.

4.3 Aspectos comparativos:

Ao contrário do modelo de difusão de referência utilizado no DALL-E 2, o Corgi codifica de forma transparente o conhecimento prévio do modelo CLIP pré-treinado em seu processo de difusão, projetando uma nova distribuição de inicialização e um novo passo de transição da difusão. Comparado com o forte modelo de referência do DALL-E 2, o Corgi apresenta melhor desempenho na geração de incorporação de imagem a partir do texto em termos de eficiência e eficácia, resultando em uma melhor geração de imagem a partir de texto. experimentos extensivo em larga escala são realizados e avaliados em termos de medidas quantitativas e avaliação humana, indicando uma capacidade de geração mais forte deste método em comparação com os exsistentes, além disso, este modelo possibilita o treinamento semi-supervisionado e de linguagem livre para geração de imagem a aprtir de texto, onde apenas parte ou nenhuma das imagens no *dataset* de treinamaento possui uma legenda associada. treinado com apenas 1,7

5 Avaliação comparativa

1. Interface do InstaFlow-0.98:



The screenshot displays the InstaFlow-0.98 interface with three main sections: 1. Setup, 2. Load model, and 3. Run. Each section has a play button icon and a 'Mostrar código' (Show code) link. In the '3. Run' section, the 'prompt' field contains the text 'A bathroom with charming light', the 'number_of_images' field is set to 3, and both the 'display_images' and 'save_images' checkboxes are checked. Red rectangular boxes highlight the prompt text, the number 3, and the checked checkboxes.

```
> 1. Setup
▶ Mostrar código

> 2. Load model
▶ model: XCLiu/instaflow_0_9B_from_sd_1_5
  torch_dtype: torch.float16
  ▶ Mostrar código

> 3. Run
▶ prompt: "A bathroom with charming light"
  number_of_images: 3
  display_images: [x]
  save_images: [x]
  ▶ Mostrar código
```

2. Teste experimentais:

Correndo o código dos métodos estudados *InstaFlow* e *Text2im* usando os mesmos dados em ambos aplicativos, para gerar as imagens partindo do texto fornecido, com os parâmetros seguintes :

- *Prompt* : "A big dog"
- *Número of image* : 2
- *Display images* : **ativado**
- *Save images* : **ativado**

E obtivemos os resultados seguintes:



3. Os detalhes sobre os resultados experimentais:

(GPU: T4) de back-end do Google Colab Compute Engine em Python 3 RAM do sistema: 3.4 / 12.7 GB RAM da GPU : 4.6 / 15.0 GB Disco : 32.8 / 78.2 GB	(GPU: T4) de back-end do Google Colab Compute Engine em Python 3 RAM do sistema: 3.4 / 12.7 GB RAM da GPU : 4.6 / 15.0 GB Disco : 32.8 / 78.2 GB
Aplicativo : InstaFlow	Aplicativo : Text2im
Imagem nº1 :	Imagem nº1 e 2 :
Indicador de progresso: 00:00;00:00 Iterações : 10.84 it por segundo Tempo : 0.42400693893432617 s	Indicador de progresso: 00:09;00:00 Iterações : 3.03 it por segundo Tempo :
Imagem nº2 :	Imagem nº :
Indicador de progresso: 00:00;00:00 Iterações : 10.11 it por segundo Tempo : 0.38594754219055176 s	

4. Discussão dos resultados obtidos:

- O ***InstaFlow***, gera as duas imagens uma por uma, e cada uma com as suas métricas enquanto o ***Text2im*** gera as duas imagens simultaneamente;
- Em ***InstaFlow*** o indicador de progresso para as ambas imagens marca 00:00;00:00, não atinge 1 segundo: 0.42400693893432617 s para a 1ª imagem e 0.38594754219055176 s para a 2ª imagem enquanto o ***Text2im*** que indica 00:09;00:00, significa fez 9 segundos para gerar as duas imagens;
- o ***InstaFlow*** gera a 1ª e 2ª imagem em 10.84 e 10.11 iterações enquanto ***Text2im*** gera as duas imagens simultaneamente em 3.03 iterações

5. Conclusão sobre discussão dos resultados:

- Por gerar imagens separadamente e cada uma com as suas métrica, o ***InstaFlow*** tem mais performance em relação ao ***Text2im***;
- Em termos da **latência**, o ***Text2im*** gera simultaneamente as duas imagens em 00:09;00:00 enquanto o ***InstaFlow*** gera duas imagens uma por uma em 00:00;00:00, **tem assim mais performance**;
- Em termos de número de **iteraões**, o ***InstaFlow*** gera a 1ª e 2ª imagem em 10.84 e 10.11 iterações enquanto ***Text2im*** gera as duas imagens simultaneamente em 3.03 iterações, **tem assim mais performance**.

6 Implementação de algumas alterações aos códigos

6.1 Códigos originais:

```
2. Load model

#@title 2. Load model
import torch

model = 'XCLiu/instafLOW_0_9B_from_sd_1_5' # @param ["XCLiu/instafLOW_0_9B_from_sd_1_5", "XCLiu/instafLOW_0_9B_from_sd_1_5"]
torch_dtype = torch.float16 # @param [torch.float16, torch.float32]{type:"r

from pipeline_rf import RectifiedFlowPipeline

pipe = RectifiedFlowPipeline.from_pretrained("XCLiu/instafLOW_0_9B_from_sd_1_5")
### switch to torch.float32 for higher quality
pipe.requires_safety_checker = False
pipe.safety_checker = None
pipe.to("cuda") ### if GPU is not available, comment this line

clear_output()
```

6.1.1 Descrição:

Disponibiliza unicamente o dispositivo GPU como o tipo de tempo de execução.

6.1.2 Problema:

Assim sendo caso tiver CPU configurado como o tipo de tempo de execução :

Alterar tipo de tempo de execução

Tipo de tempo de execução

Python 3 ▼

Acelerador de hardware ?

☒ CPU ☐ T4 GPU ☐ A100 GPU ☐ L4 GPU

☐ V100 GPU (deprecated) ☐ TPU (deprecated)

☐ TPU v2

Quer aceder a GPUs premium? [Compre unidades de computação adicionais](#)

Cancelar Guardar

O *InstaFlow* gera o erro :

> 3. Run



prompt: "A big dog"

number_of_images: 2

display_images: ☒

save_images: ☒

[Mostrar código](#)



```
-----  
NameError                                Traceback (most recent call last)  
<ipython-input-1-de378b3181a1> in <cell line: 13>()  
    13 for i in range(0,number_of_images):  
    14     time_0=time.time()  
--> 15     images = pipe(prompt=prompt,  
    16                     num_inference_steps=1,  
    17                     guidance_scale=0.0).images
```

NameError: name 'pipe' is not defined

6.2 Códigos alterados:

2. Load model

```
#@title 2. Load model
import torch

model = 'XCLiu/instafLOW_0_9B_from_sd_1_5' # @param ["XCLiu/instafLOW_0_9B_from_sd_1_5", "XCLiu/instafLOW_0_9B_from_sd_1_5"]
torch_dtype = torch.float16 # @param [torch.float16, torch.float32]{type:"r"}

from pipeline_rf import RectifiedFlowPipeline

pipe = RectifiedFlowPipeline.from_pretrained("XCLiu/instafLOW_0_9B_from_sd_1_5")
### switch to torch.float32 for higher quality
pipe.requires_safety_checker = False
pipe.safety_checker = None
#pipe.to("cuda") ### if GPU is not available, comment this line
#Verificar se o GPU está disponível senão usar o CPU
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
# Mover o pipeline para dispositivo adequado
pipe.to(device)

clear_output()
```

6.2.1 Descrição:

- O código verifica se há uma GPU disponível
- Se houver, define o dispositivo como GPU(cuda)
- Senão houver, define o dispositivo como CPU
- Move o pipeline do modelo para o dispositivo adequado, garantindo que as operações subsequentes sejam executadas no dispositivo correto (GPU ou CPU).

6.2.2 Resultados:

Aproveita-se o poder computacionais da GPU (se disponível). Se uma GPU não estiver disponível, o modelo será movido para CPU (embora tem tempo de execução mais longo comparado ao GPU), garantindo que o código funcione em qualquer máquina, independentemente da presença de uma GPU, **com isso erradica-se o erro gerado anteriormente devido a indisponibilidade da GPU**. Isso é uma prática comum para aproveitar o hardware disponível da melhor forma possível, otimizando a performance do aplicativo.

7 Conclusão

Neste projeto, exploramos a geração de imagens a partir de texto utilizando o método de descrição textuais no contexto de aplicações criativas e práticas. Em nossa análise,

apresentamos dois (2) métodos recentes e inovadores (InstaFlow e Text2im) na área de difusão para geração de imagens. Realizamos uma avaliação comparativa desses métodos, considerando as suas abordagens, vantagens e limitações.

Adicionalmente, implementamos uma alteração ao código existente do *InstaFlow*, baseada nas percepções obtidas durante a análise dos métodos recentes. Essa modificação, teve como objetivo aprimorar os tipos de tempo de execução entre GPU e CPU de modo a se aplicar alternativamente.

Os resultados experimentais e comparativos obtidos entre os dois modelos distintos *InstaFlow* e *Text2im* mostraram que as técnicas de difusão de imagens a partir de descrições textuais são ambas **eficazes, mas com eficiência relativa** em termos de **(1) latência em geração de imagens, (2) o número de iterações** para tal geração e **(3) a técnica utilizada**.

Concluimos que o uso de métodos de difusão no *InstaFlow* oferece um avanço significativo em relação às abordagens tradicionais. Este projeto abre caminho para futuras pesquisas e desenvolvimento de novas técnicas que possam aprimorar ainda mais a geração de imagens a partir de texto, proporcionando ferramentas cada vez mais poderosas para criadores de conteúdo e desenvolvedores.

Referências

- [1] Ruiqi Gao Diederik Kingma Stefano Ermon Jonathan Ho] Chenlin Meng, Robin Rombach and Tim Salimans. *On distillation of guided diffusion models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 14297–14306.* Paris, 2023.
- [2] Richard Zhang Jaesik Park Eli Shechtman Sylvain Minguk Kang, Jun-Yan Zhu. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 10124–10134.* Paris, and Taesung Park. Scaling up gans for text-to-image synthesis, Paris, 2023.