

Part 6

Stable Diffusion 部署 及 AI 绘画解析



Stable Diffusion

"Stable Diffusion"是一种深度学习模型，主要用于将文本描述转化为详细的图像，也可以应用于其他图像处理任务。这个模型由创业公司 Stability AI 与学术研究者合作开发，使用了一种称为潜在扩散模型（LDM）的扩散模型。

1. 部署 Stable Diffusion
2. 给 Stable Diffusion 安装模型
3. Stable Diffusion 的原理

主流 AI 软件对比

与 Midjourney 相比，Stable Diffusion（以下简称 SD）的优点有：

① 免费开源

- Midjourney 需要科学上网使用，且最低 10 美元/月。而 Stable Diffusion 有已经整理好的整合安装包，可以免费下载一键安装。安装到本地的 Stable Diffusion 随开随用，生成的图片只有自己能看到，保密性更强。

② 拥有强大的外接模型

- 由于开源属性，SD有很多免费高质量的外接预训练模型（fine-tune）和插件，比如可以提取物体轮廓、人体姿势骨架、画面深度信息、进行语义分割的插件Controlnet，使用它可以让我们在绘画过程中精准控制人物的动作姿势、手势和画面构图等细节，这些都是目前 Midjourney 无法做到的。

③ 训练自己的模型

- 我们可以用 SD LORA 或者 Dreambooth 将自己喜欢的人物形象或者画风训练生成模型，打造自己的专属 AI 绘画工具，也有画师和设计师用自己的原创作品训练模型，之后有灵感了就之后用文字描述出来，让 SD 帮自己快速出各种概念草图。



推特网友 @Claude 利用带有 controlnet 和 mov2mov 功能的 SD webUI，将真人视频转换为动画风格。

主流 AI 软件对比

与 Midjourney 相比，Stable Diffusion（以下简称 SD）的缺点有：

- Stable Diffusion 的各种参数比 Midjourney 要复杂，而且很多功能都需要通过额外安装插件和模型来实现，因此我们需要了解各种参数的含义，不同的模型插件的使用方法，可能还需要了解背后的原理知识，因此入门学习的难度比 Midjourney 高。
- Stable Diffusion 对电脑的显卡要求较高，自行部署难度较大。

主流 AI 软件对比

Stable Diffusion 和 Midjourney 对比

软件对比

收费方式

商用版权

出图效果

优势评价

缺点评价

推荐指数

Stable Diffusion

开源免费

授权可免费商用

出图效果依赖选中的模型或lora

本地部署后无需联网，出图可控性强

入手难度高、电脑配置要求高



Midjourney

最低9美金/月，价格略贵

需要付费才可商用，网上存在争议

出图效果高、画面丰富

出图质量高、效果丰富，需要全程网络

出图可控性差，内容有限制，费用贵



1. 部署 Stable Diffusion

1.1 本地部署 Stable Diffusion

1.2 云端部署 Stable Diffusion

1.3 部署过程中出现问题的解决方案

Stable Diffusion 配置清单

配置清单并不绝对，低配也可以运行但体验效果不佳，下面的配置基于互联网上用户的推荐配置

最低配置：

操作系统：无硬性要求

CPU：无硬性要求

显卡：GTX 1660Ti 及同性能显卡

显存：6GB

内存：8GB

硬盘空间：20GB的可用硬盘空间

约 **1-2 分钟**出一张图，可绘制分辨率 **512*512 像素**

推荐配置：

操作系统：Windows 10 64位

CPU：支持64位的多核处理器

显卡：RTX 3060Ti 及同等性能显卡

显存：8GB

内存：16GB

硬盘空间：100~150GB的可用硬盘空间

约 **10-30 秒**出一张图，可绘制分辨率 **1024*1024 像素**

Stable Diffusion 的显卡要求

Stable Diffusion对显卡的要求较高，运行Stable Diffusion 优先选择NVIDIA，显卡需要为独立显卡，显卡的性能与显存会影响操作的体验。

显卡的另一个核心指标：**显存**

显存全称显示存储器，亦称帧缓存，它是用来存储显示芯片处理过或者即将读取的渲染数据。如同电脑的内存一样，显存是用来存储图形数据的硬件。在显示器上显示出的画面是由一个个的像素点构成的，而每个像素点都以4至64位的数据来控制它的亮度和色彩，这些点构成一帧的图形画面。为了保持画面流畅，要输出和要处理的多幅帧的像素数据必须通过显存来保存，达到缓冲效果，再交由显示芯片和中央处理器调配，最后把运算结果转化为图形输出到显示器上。



临时存储显卡的图形数据，提供图形显示和运算的“缓冲地带”

显存对生成图片的影响

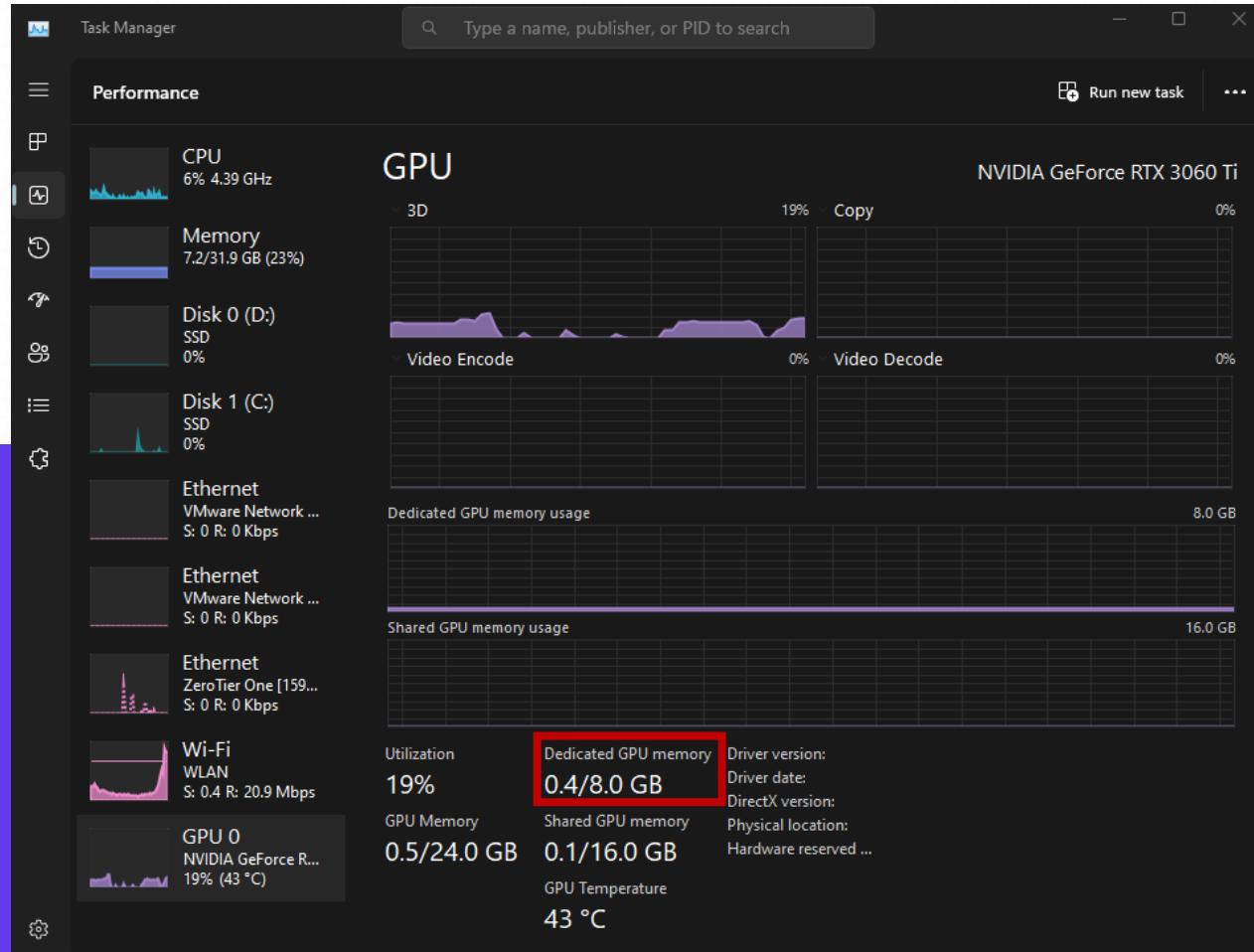
1. 图片绘制分辨率：

显存过低的显卡，通常只能绘制尺寸较小的图片，且无法同时加载过多模型；

2. 模型训练规模（进阶）：

如要自行训练各类模型，最好用高配置的台式机，显卡选 40 系列及以上，显存最低 12G，CPU 选 Intel i7 或者以上，内存越大越好。

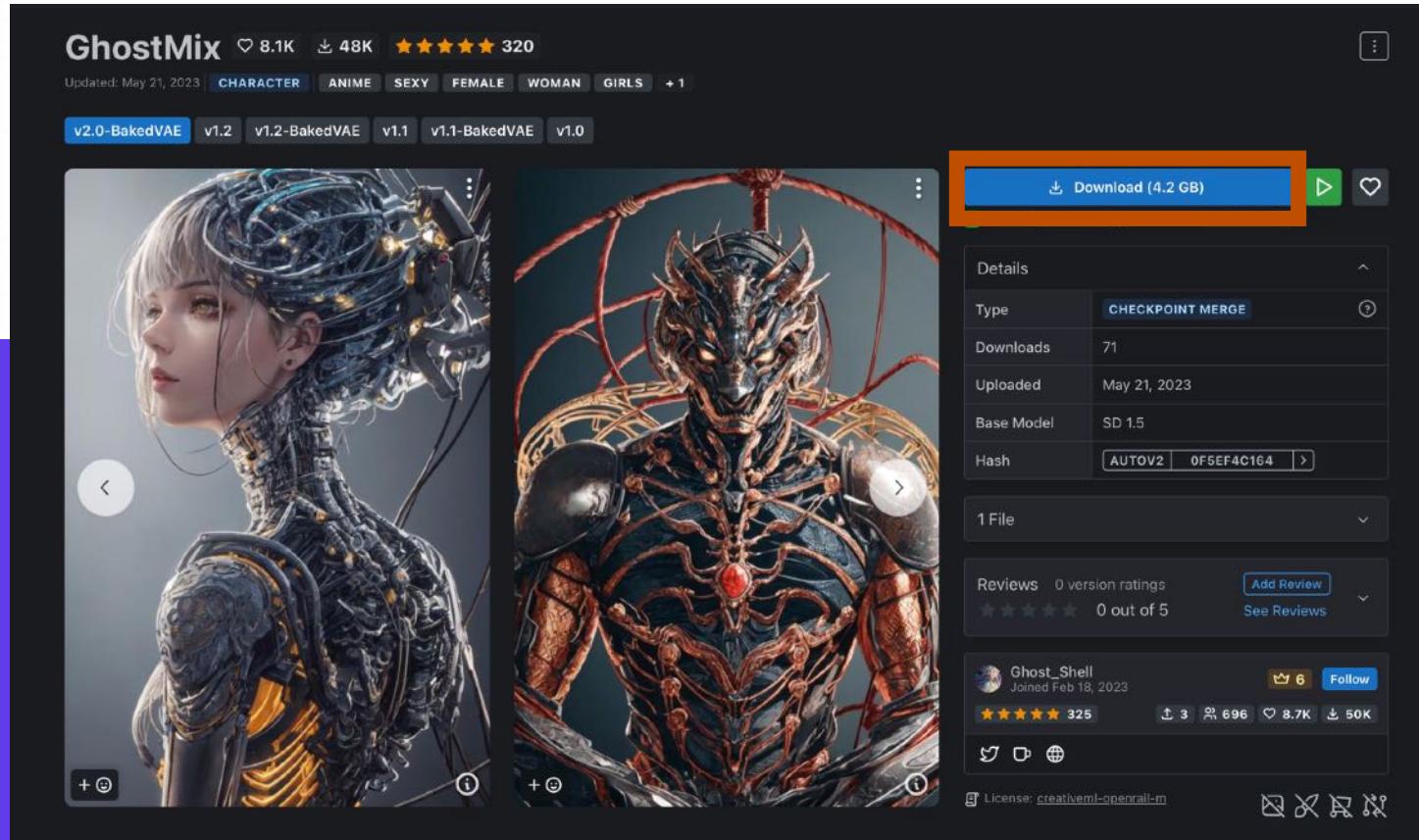
Stable Diffusion 的显卡要求



检查电脑显存的方法 (windows) :

鼠标放在工具栏，单击右键打开“任务管理器”，选择顶部的“性能”，进入后下拉查看 GPU 的部分，找到“专用 GPU 内存”，下面对应的数字就是电脑的显存。下图显示的电脑显存是 2G，生成一张最低配置的 512*512px 的图像需要 8 分钟，而 32G 显存的生成速度只需几秒，所以显存低的电脑不适合运行 Stable Diffusion。

Stable Diffusion 的硬盘空间



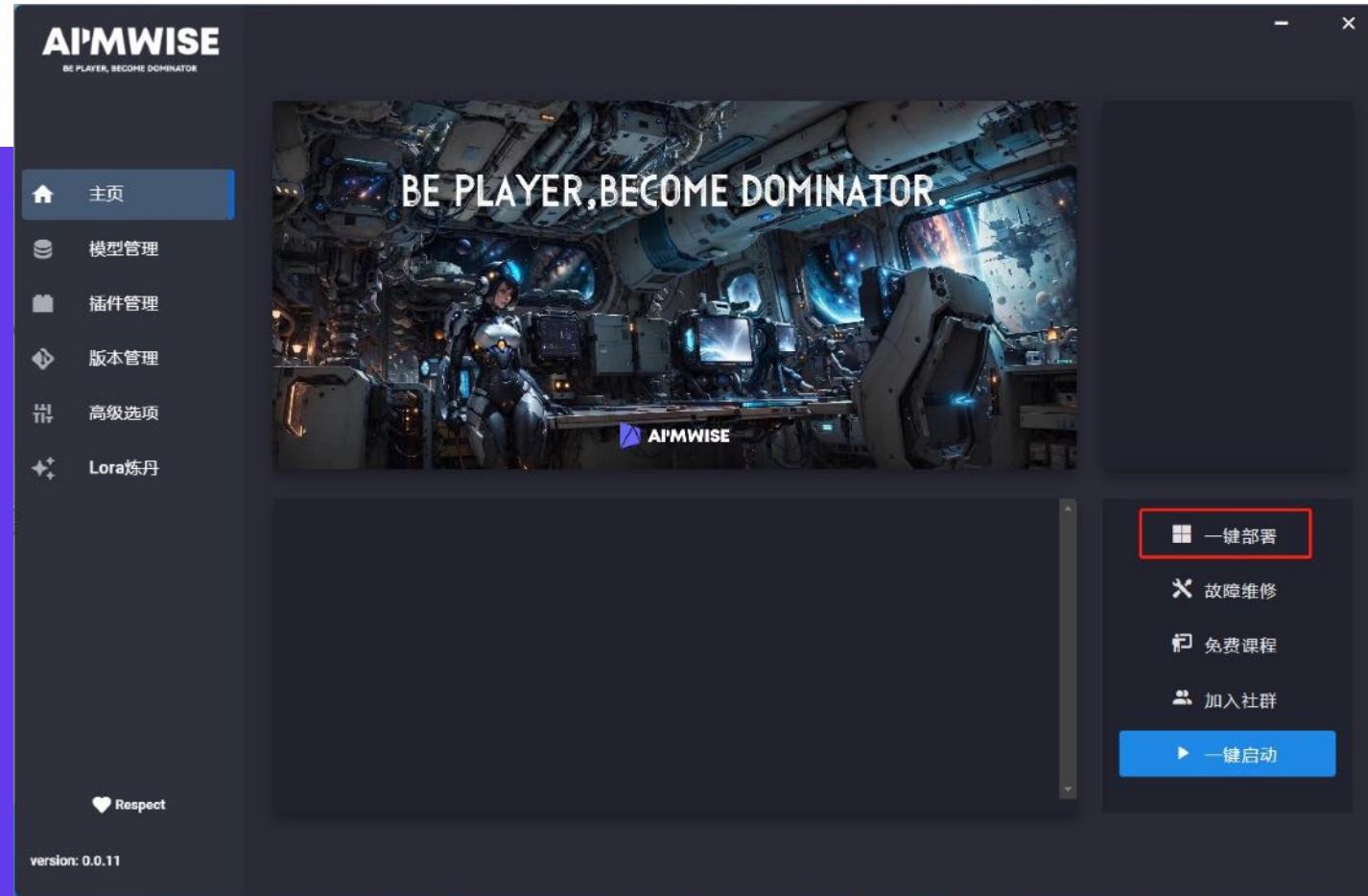
需要较大的硬盘空间

因为 SD 所有文件都是安装在本地电脑上，而且生成不同的风格图像需要下载不同的模型，一个大模型至少 2G，假设你想要尝试 10 种风格，那 10 个大模型可能就要 30G 的空间。一般要求装 SD 的那个盘要有 100G 的剩余空间，最低不小于 40G，且最好不要安装在 C 盘。

1.1 Stable Diffusion 本地部署

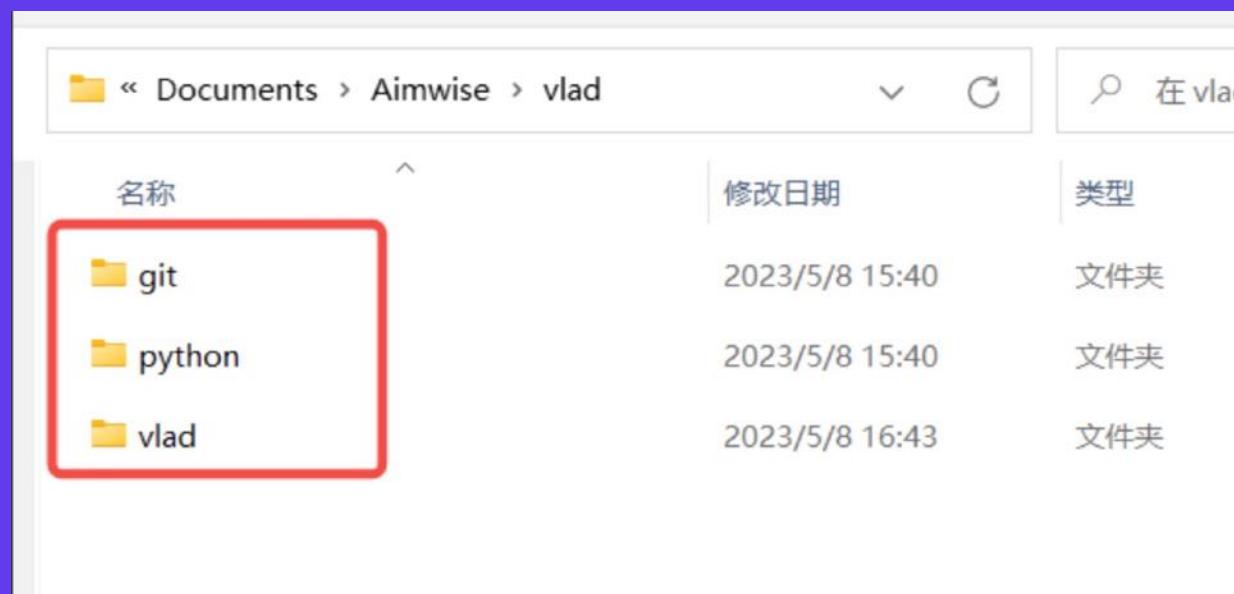
1. Aimwise启动器

Aimwise启动器基于 vladmandic 版本的 stable diffusion 源码开发跨平台一键部署启动器，支持全系列N卡， A卡， M1芯片， vladmandic版本中文汉化，适配kitchen主题，内置命令行。永久免费使用。



1.1 Stable Diffusion 本地部署

2. 安装 Stable diffusion



Step 3:

安装后，我们选择的目录里面就可以看到，三个文件夹了

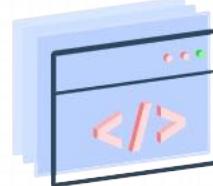
- **git**: 用于存放git程序，git的作用是能够对软件进行更新以及安装软件所需要的依赖
- **python**: 软件是基于python进行开发的，在python环境下才能运行，这里将python内置到了我们的目录，不再需要手动配置和下载了。
- **vlad**: 程序的主要运行单元，vladmandic版本的SD源码，模型、插件的主要存放地方

1.2 云端部署 - 前期准备



能上网的电脑

支持任意系统



稳定的浏览器

Safari, chrome, edge



能扫码的手机

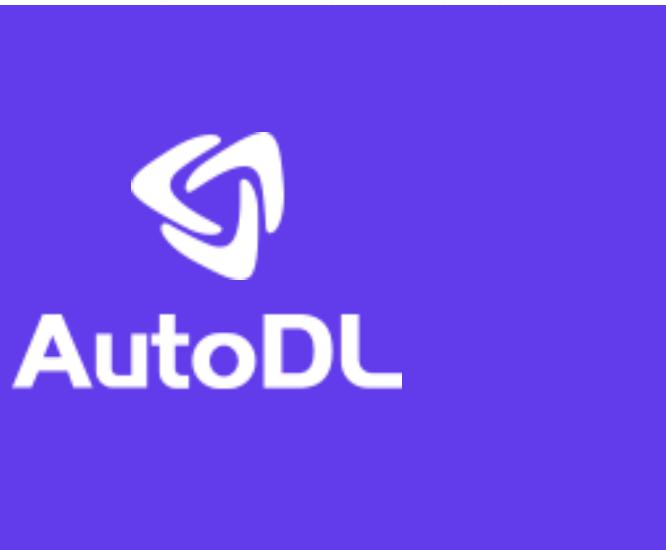
用于登录注册、支付会员



云部署平台

www.autodl.com

1.2 云端部署 - 创建实例



AutoDL 是一个GPU租用平台，有各种型号的服务器可以选择。网络无要求，部署简单，比较稳定，价格也还好。

对电脑系统没有要求，无论是Mac电脑还是的windows电脑，都可以进行云端部署，租用服务器后使用浏览器打开即可。全部在云端完成，有很多服务器可以选择，可以根据自己需求租相应配置的服务器。

1.2 云端部署 - 创建实例

Step1 . 登录注册云部署平台

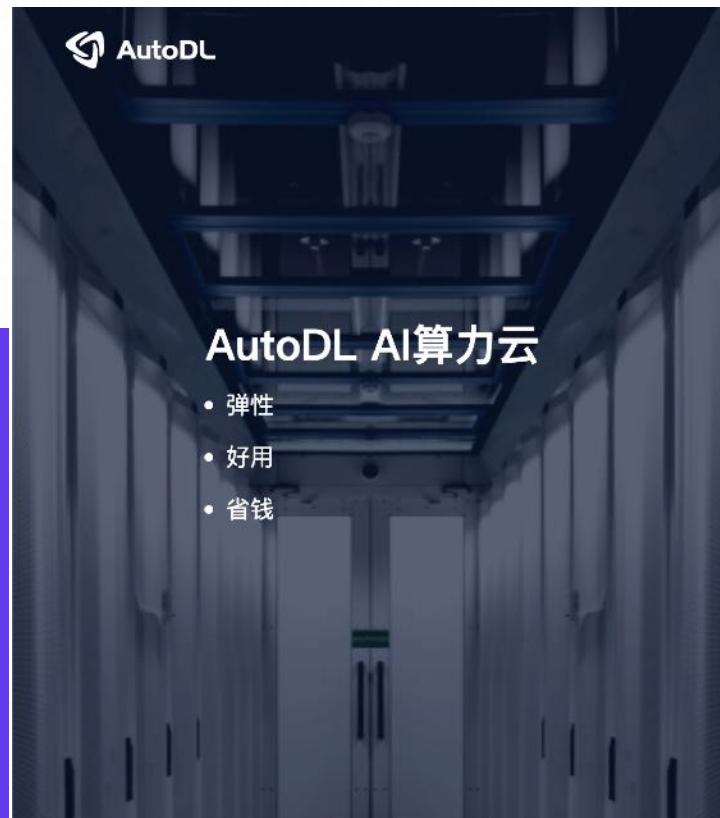


AutoDL官网: <https://www.autodl.com/home>

1.2 云端部署 - 创建实例

Step1 . 登录注册云部署平台

点击右上角的注册或登录，或者点击“微信登录”，扫码并关注公众号，即可成功登录。



1

AutoDL AI 算力云

• 弹性
• 好用
• 省钱

登录 **注册**

+86 ▾ 请输入手机号

请输入验证码 **发送验证码**

请输入8~16位包含数字和字母组合

我已阅读并同意 [《AutoDL服务协议》](#)

注册

已有账号, [点击登录](#)

2

AutoDL AI 算力云

• 弹性
• 好用
• 省钱

登录 **注册**

+86 ▾ 请输入手机号

请输入密码

登录

[微信登录](#)

[短信登录](#) [子账号登录](#) [忘记密码?](#)

1.2 云端部署 - 创建实例

Step2 .会员充值

点击右上角的“控制台”，进入页面，这里首次使用，建议先充值。

The screenshot shows the AutoDL Control Panel. A red circle labeled '1' highlights the 'Control Panel' button in the top right corner. A red circle labeled '2' highlights the 'Recharge' button in the 'Funds Information' section, which is enclosed in a red box. The 'Funds Information' section displays a balance of '¥ 0.00'.

AutoDL Control Panel Screenshot:

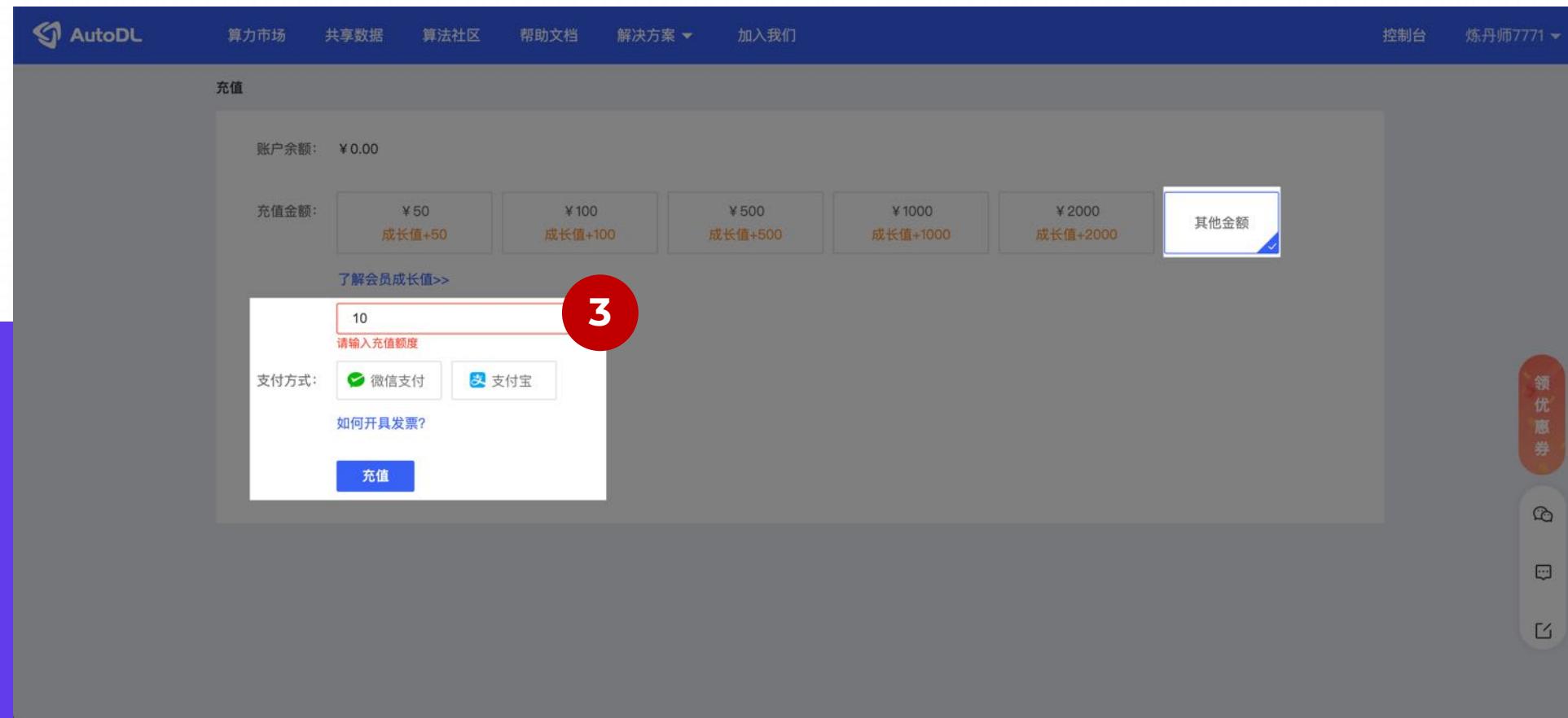
- Control Panel** (highlighted by red circle 1)
- Funds Information** (highlighted by red box and red circle 2)
 - 我的余额: ¥ 0.00 (Recharge button)
 - 代金券: ¥ 0.00
 - 优惠券: 暂无
 - 授信: 暂无
- 余额预警**: 已开启 (开启)

1. 不充值会导致无法快速租赁到合适的显卡
2. 体验玩家可以小额充值，比如10块钱

1.2 云端部署 - 创建实例

Step2 .会员充值

可以先小额充值进行体验，先进行充值的目的是为了在后续更快速的抢到服务器。



1.2 云端部署 - 创建实例

Step3. 租用实例

3.1 进入租用实例页面

The screenshot shows the AutoDL platform's main interface. On the left is a sidebar with various navigation options like '算力市场', '共享数据', '算法社区', etc. The main content area has a large '实例' (Instances) section with counts for '容器实例' (0), '运行中' (0), and '已关机' (0). Below this is a '数据管理' (Data Management) section showing '0GB/0GB' used on a data disk. To the right are sections for '费用信息' (Billing Information) showing a balance of '¥0.00' and a '充值' (Top Up) button, and '常见问题' (FAQ) with links to help topics.

租用实例可以理解为购买算力

点击左侧的菜单，或者中间的“容器实例”的数值，新手这里是0，点击以后，跳转到容器实例页面，

这里会显示你租用的实例列表，后续的操作也是在这里进行。

1.2 云端部署 - 创建实例

Step3. 租用实例

3.1 进入租用实例页面

AutoDL 算力市场 共享数据 算法社区 帮助文档 解决方案 加入我们 控制台 炼丹师7771

主页 容器实例 ① 实例连接 放实例, 实例释放会导致数据清空且不可恢复, 释放前实例在数据在。 ② 小程序管理实例 订阅GPU通知 设置密钥登录 搜索实例名称/ID

租用新实例

实例ID / 名称	状态	规格详情	本地磁盘	健康状态	付费方式	释放时间/停机时间	SSH登录	快捷工具	操作
暂无数据									

这里会显示你租用的实例/服务器的列表,
首次使用, 是空的

共 0 条 < 1 > 10条/页 前往 1 页

1.2 云端部署 - 创建实例

Step3. 租用实例

3.1 进入租用实例页面

The screenshot shows the AutoDL platform's instance creation interface. At the top, there are navigation links: 算力市场, 共享数据, 算法社区, 帮助文档, 解决方案 ▾, 加入我们, 控制台, and 炼丹师7771 ▾. A red warning banner at the top states: “严禁使用WebUI等算法生成违禁图片、严禁挖矿，一经发现立即封号！” Below the banner, the page title is “容器实例 / 创建实例”. Under “计费方式”, the “按量计费” tab is selected. A note below says: “创建完主机后仍然可以转换计费方式。如选择按量计费，价格发生变动以实例开机时的价格为准”。 The “选择地区” dropdown is set to “内蒙A区”. Under “GPU型号”, several options are listed: 全部, RTX 3090 (0/544), RTX A5000 (14/1150), A100-SXM4-80GB (0/32), A40 (5/384), RTX 3080 (0/56), and TITAN Xp (0/48). The “GPU数量” dropdown is set to “1”. The “选择主机” table lists four hosts:

主机ID	算力型号/显存	空间GPU	每GPU分配	CPU型号	硬盘	最高CUDA	支持网盘	价格(单卡)
660机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 629GB	12.0	否	¥1.18/时 ¥1.24/时
725机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 904GB	12.0	否	¥1.18/时 ¥1.24/时
692机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 974GB	12.0	否	¥1.18/时 ¥1.24/时
722机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 1000GB	12.0	否	¥1.18/时 ¥1.24/时

At the bottom, it shows “日常费用: ￥0.00/日”, “配置费用: ￥1.18/时 费用明细”, and “账户余额: ￥0.00 余额不足去充值”. There are “取消” and “立即创建” buttons.

注：账户中如果没有充值，右下角的“立即创建”的按钮图标是灰色，无法进行点击

1.2 云端部署 - 创建实例

Step3. 租用实例

3.1 进入租用实例页面

容器实例 / 创建实例

3 计费方式: 按量计费 包日 包周 包月

创建完主机后仍然可以转换计费方式。如选择按量计费，价格发生变动以实例开机时的价格为准。

4 选择地区: 内蒙A区 北京A区 西北A区 佛山区 芜湖区 北京C区 毕业季A区 毕业季B区 [如何选择GPU?](#)

GPU型号: 全部 RTX 3090 (2/544) RTX A5000 (121/1150) A100-SXM4-80GB (0/32) A40 (32/416) RTX 3080 (5/56) TITAN Xp (0/48)

GPU数量: 1 2 3 4 5 6 7 8

选择	主机ID	算力型号/显存	空闲GPU	每GPU分配	CPU型号	硬盘	最高CUDA	支持网盘	价格(单卡)
<input checked="" type="radio"/>	701机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 960GB	12.0	否	¥1.18/时 ¥1.24/时
<input type="radio"/>	631机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 974GB	12.0	否	¥1.18/时 ¥1.24/时
<input type="radio"/>	641机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 574GB	12.0	否	¥1.18/时 ¥1.24/时
<input type="radio"/>	667机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 774GB	12.0	否	¥1.18/时 ¥1.24/时
<input type="radio"/>	642机	RTX A5000 24GB	1 / 2	CPU: 15核 内存: 28GB	AMD EPYC 7371	数据盘: 50GB 可扩容: 876GB	12.0	否	¥1.18/时 ¥1.24/时

日常费用: ￥0.00/日 [?](#) 配置费用: ￥1.18/时 [费用明细](#) 账户余额: ￥5.85

6 取消 立即创建

点击表格左上方的“租用新实例”按钮，去租用页面，页面包括一下几模块：**计费方式、选择主机、选择镜像**。

计费方式:

可以根据自己情况选择，个人选择按量计费，比较方便，用多少扣多少。

选择地区:

都可以选择，不同地区的配置费用不同（北京C区因政策原因有些功能无法使用）。

GPU型号:

SD 部署对显卡GPU 的要求比较高，建议选择 **RTX 3060**以上的显卡。

GPU数量:

只要有空闲的就可以选择使用，一般8GB就可以快速的渲染图片，8GB以上的GPU可以轻松生成一些图片。

1.2 云端部署 - 创建实例

Step3. 租用实例

3.2 选择社区镜像

The screenshot shows a user interface for deploying AI models. At the top, there's a navigation bar with tabs: '基础镜像' (Basic Image), '社区镜像' (Community Image) which is highlighted with a red box and a red number '7' above it, '我的镜像' (My Image), and a question '社区镜像是什么?' (What is a community image?). Below this is a search bar with placeholder text '试试搜索Github项目名称, 如: YOLOv5、mmDetection。即刻拥有创建即用的算法运行环境' (Try searching for GitHub project names, such as YOLOv5, mmDetection. Instantly get a ready-to-use algorithm runtime environment). A warning message '严禁使用WebUI等算法生成违禁图片,一经发现立即封号!' (Prohibited to use WebUI or other algorithms to generate prohibited images,一经发现立即封号!) follows. A search input field contains '请输入Github项目搜索镜像' (Enter GitHub project search image). A note at the bottom says '创建完成后仍然可以更换其他镜像' (After creation, you can still change to other images). The main area lists several Docker image options:

镜像	描述	作者	大小	操作
AUTOMATIC1111/stable-diffusion-webui/ NovelAI-Consolidation-Package-3.1:v6	(本地整合版)NovelAI+DreamBooth+DeepdanBooru+...	xiaolxl	24.0k	[View]
AUTOMATIC1111/stable-diffusion-webui/ NovelAI-Consolidation-Package-3.1:v7	(本地整合版)NovelAI+DreamBooth+DeepdanBooru+...	xiaolxl	154	[View]
AUTOMATIC1111/stable-diffusion-webui/ NovelAI-Consolidation-Package-3.1:v9	(本地整合版)NovelAI+DreamBooth+DeepdanBooru+...	xiaolxl	2.1k	[View]

Below the list is a search input field containing 'novel'. At the bottom, there are buttons for '日常费用' (Daily Cost: 0.00), '配置费用' (Configuration Cost: 1.18), '费用明细' (Cost Details), and '账户余额' (Account Balance: 5.85). A large red arrow points from the text '选择“社区镜像”，会有下拉列表，这里可以输入“novel”，在下拉列表中选择“AUTOMATIC1111/stable-diffusion-webui/NovelAI-Consolidation-Package-3.1:v6”' down to the '立即创建' (Create Now) button, which is also highlighted with a red box and a red number '9' above it.

选择“社区镜像”，会有下拉列表，这里可以输入“novel”，在下拉列表中选择“AUTOMATIC1111/stable-diffusion-webui/NovelAI-Consolidation-Package-3.1:v6”。之后选择“立即创建”。

1.2 云端部署 - 创建实例

Step 3. 租用实例

3.3 点击立即创建

AutoDL Platform Screenshot:

- Left Sidebar:** Includes links for Home, Container Instances (highlighted), File Storage, Image Registry, Public Data, Costs, Purchase Details, My Orders, Account Statement, Vouchers, Invoice Management, Accounts, Account Security, Sub-Accounts, and Settings.
- Top Navigation:** Market, Shared Data, Algorithm Community, Help Docs, Solutions, Join Us, Control Panel, and User Profile (炼丹师7552).
- Container Instances Section:**
 - Message: "实例连续关机30天会释放实例，实例释放会导致数据清空且不可恢复，释放前实例在数据在。"
 - Buttons: "租用新实例" (Create New Instance), "订阅GPU通知" (Subscribe GPU Notifications), "设置密钥登录" (Set Key Login), and "搜索实例名称/ID" (Search Instance Name/ID).
 - Table Headers: 实例ID / 名称, 状态, 规格详情, 本地磁盘, 健康状态, 付费方式, 释放时间/停机时间, SSH 登录, 快捷工具, 操作.
 - Instance Details: 内蒙A区 / 065机, 8fb311a93c-0f245cd (Status: Creating 90.65%), RTX A5000 * 1卡, 系统盘 0%, 数据盘 0%, 正常 (Health), 按量计费 (Billing), 余额不足 (Insufficient Balance), 29天23小时后释放 (Release after 29 days and 23 hours), 设置定时关机 (Set Timed Power Off), and a "More" button.

点了“立即创建”以后，会刷新一个新页面，里边就是实例的列表，第一次创建实例的时候，拉取镜像需要大概1分钟左右的时间（会显示拉取的进度/百分比），拉取成功进行开机，开机成功到运行中。

这里有一点，因为是按量计费，所以“开机”就开始计费了，如果不使用的记得“关机”！否则会一直扣费，直到账户没钱。

1.2 云端部署 - 部署SD操作界面

Step 1. 打开程序运行页

AutoDL Platform Screenshot:

- Container Instances Page:** Shows a list of instances. One instance is circled in red.
- Action Column:** For each instance, there is a 'Actions' (操作) section. A red arrow points to the 'JupyterLab' link in this section.

1

点击“快捷工具”里的“JupyterLab”，页面会跳转到 JupyterLab 运行页面

Jupyter Notebook Screenshot:

- File List:** Shows files like 'MainUi.ipynb' and 'ui-script'.
- Code Cell:** Contains the following text (highlighted by a red box):


```
第三代整合版第一次启动请执行下方代码(仅限第三代整合版)
请执行下方代码(只需要执行第一次即可)
执行完后刷新浏览器
刷新完后切换笔记本内核(右上角点击后选择xl_env环境)
请在运行后不要进行任何其它操作, 直到出现'移动完成'方可使用
```
- Terminal Tab:** Shows a terminal window labeled '终端 1'.

这是里开发者/作者，写的操作步骤

1.2 云端部署 - 部署SD操作界面

Step 2. 运行第一段代码

The screenshot shows a Jupyter Notebook interface. At the top, there are two tabs: 'MainUi.ipynb' and '终端 1'. Below the tabs is a toolbar with icons for file operations, a search bar, and a dropdown menu labeled '代码'. A red circle with the number '3' highlights the play button icon in the toolbar. A red circle with the number '2' highlights the first code cell.

第三代整合版第一次启动请执行下方代码(仅限第三代整合版)

请执行下方代码(只需要执行第一次即可)

执行完后刷新浏览器

刷新完后切换笔记本内核(右上角点击后选择xl_env环境)

请在运行后不要进行任何其它操作，直到出现 '移动完成' 方可使用

```
[1]: !chmod 777 /root/set_env.sh  
!/root/set_env.sh  
!mkdir /root/.cache  
!mv -b -f /root/temp/* /root/.cache/ && echo '移动完成'
```

第一步：鼠标点击第一个代码框区域，
选中状态左侧会出现蓝色的高亮条

第二步：点击“运行”按钮（小三角）

第三步：刷新页面 (ctrl + R)

1.2 云端部署 - 部署SD操作界面

Step 2. 运行第一段代码

MainUi.ipynb 终端 1

第三代整合版第一次启动请执行下方代码(仅限第三代整合版)

请执行下方代码(只需要执行第一次即可)

执行完后刷新浏览器

刷新完后切换笔记本内核(右上角点击后选择xl_env环境)

请在运行后不要进行任何其它操作, 直到出现 '移动完成' 方可使用

```
[1]: !chmod 777 /root/set_env.sh  
!/root/set_env.sh  
!mkdir /root/.cache  
!mv -b -f /root/temp/* /root/.cache/ && echo '移动完成'
```

Installed kernelspec xl_env in /root/.local/share/jupyter/kernels/xl_env

4 移动完成

第四步：“运行后，会在代码下面出现“移动完成”字样

1.2 云端部署 - 部署SD操作界面

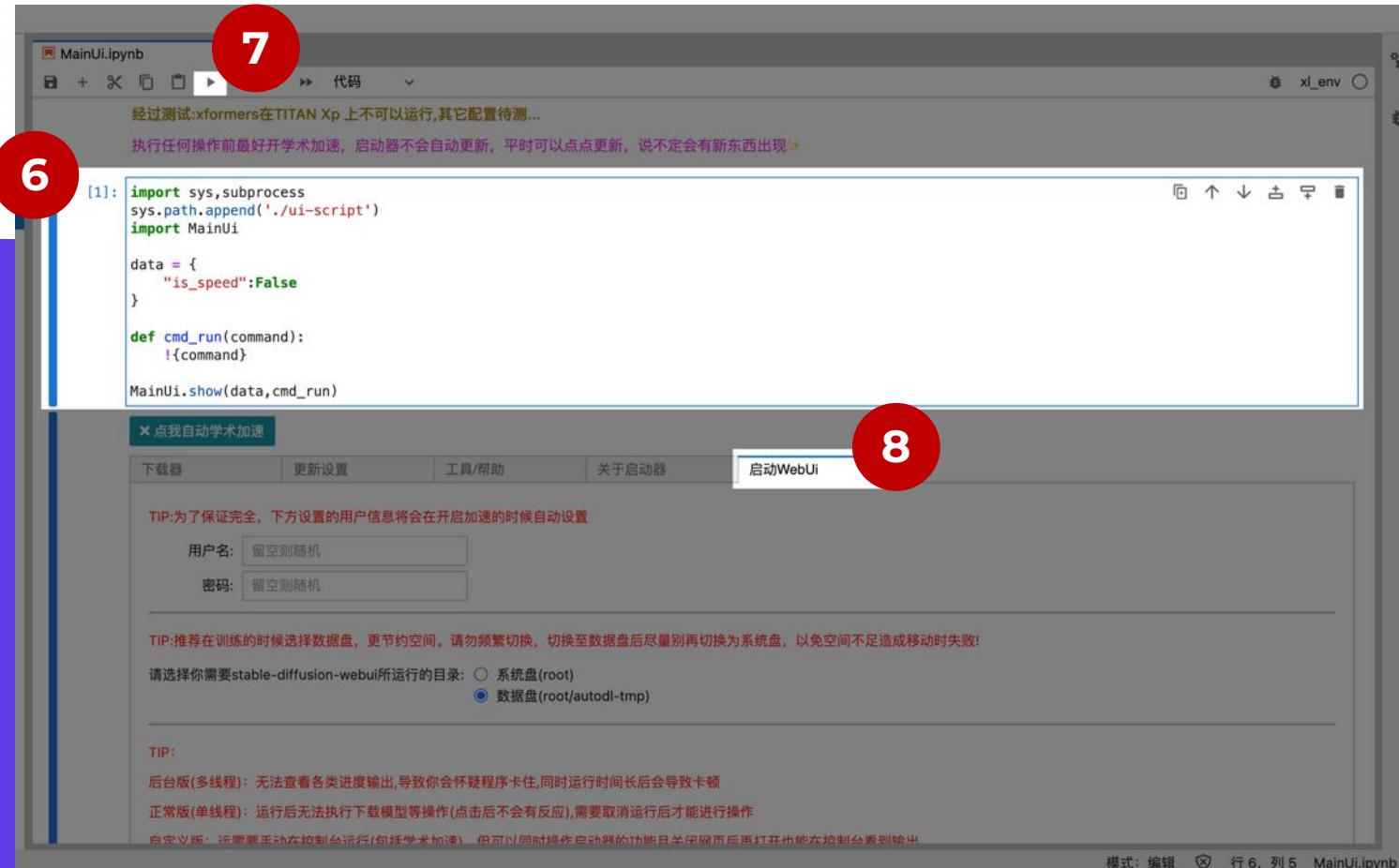
Step 3. 切换内核



第五步：切换笔记本内核（右上角点击后将python切换为xl_env环境）

1.2 云端部署 - 部署SD操作界面

Step 4. 运行第二段代码



1.2 云端部署 - 部署SD操作界面

Step 5. 运行结果展示



```

Found existing installation: requests 2.25.1
Uninstalling requests-2.25.1:
  Successfully uninstalled requests-2.25.1
Attempting uninstall: gitpython
Found existing installation: GitPython 3.1.27
Uninstalling GitPython-3.1.27:
  Successfully uninstalled GitPython-3.1.27
Attempting uninstall: accelerate
Found existing installation: accelerate 0.12.0
Uninstalling accelerate-0.12.0:
  Successfully uninstalled accelerate-0.12.0
Attempting uninstall: transformers
Found existing installation: transformers 4.25.1
Uninstalling transformers-4.25.1:
  Successfully uninstalled transformers-4.25.1
Successfully installed accelerate-0.16.0 gitpython-3.1.31 requests-2.30.0 transformers-4.26.1

[+] torch version 1.13.1+cu117 installed.
[+] torchvision version 0.14.1+cu117 installed.
[+] accelerate version 0.16.0 installed.
[+] bitsandbytes version 0.35.4 installed.
[+] diffusers version 0.13.1 installed.
[+] transformers version 4.26.1 installed.
[+] xformers version 0.0.17.dev464 installed.

#####
Launching Web UI with arguments: --disable-safe-unpickle --port=6006 --deepdanbooru --xformers --enable-insecure-extension-access --no-half-vae
Script path is
Loading weights [89d59c3dde] from /root/autodl-tmp/stable-diffusion-webui/models/Stable-diffusion/model.ckpt
Creating model from config: /root/autodl-tmp/stable-diffusion-webui/configs/v1-inference.yaml
LatentDiffusion: Running in eps-prediction mode
DiffusionWrapper has 859.52 M params.
Loading VAE weights found near the checkpoint: /root/autodl-tmp/stable-diffusion-webui/models/Stable-diffusion/model.vae.pt
Applying xformers cross attention optimization.
Textual inversion embeddings loaded(0):
Model loaded in 6.1s (load weights from disk: 1.7s, create model: 0.6s, apply weights to model: 0.5s, apply half(): 0.5s, load VAE: 2.2s, move model to device: 0.5s).
Running on local URL: http://127.0.0.1:6006

To create a public link, set `share=True` in `launch()`.
```

切换到“启动WebUI”这一栏,下拉,把图片上的几个选项勾选,点击绿色运行按钮,下方会出现执行代码,我们需要等待几十秒,直到出现一个URL。

1.2 云端部署 - 部署SD操作界面

Step 6. 实名认证

The screenshot shows the Alipay account security settings page. On the left sidebar, there are several tabs: 主页 (Home), 容器实例 (Container Instance), 文件存储 (File Storage), 镜像 (Image), 公开数据 (Public Data), 费用 (Cost), 收支明细 (Income and Expenditure Details), 我的订单 (My Orders), 账单明细 (Bill Details), 代金券 (Coupon), and 优惠券 (Promotion). The current tab is '账号安全' (Account Security).

The main content area displays various security features:

- 登录密码**: A note says "安全性高的密码可以使账号更安全。建议您定期更换密码，设置一个包含字母和数字且长度超过8位的密码" (A secure password makes the account safer. It is recommended to change it regularly, setting a password containing letters and numbers with a length of more than 8 digits). Status: 未设置 (Not Set) | 设置 (Set).
- 手机绑定**: A note says "您已绑定了手...您的手机号可以直接用于登录、找回密码等" (Your phone number has been bound... Your phone number can be used directly for login, password recovery, etc.). Status: 已绑定 (Bound) | 修改 (Modify).
- 实名认证**: A note says "实名认证后可以使用AutoDL更完整的功能，如打开实例的自定义服务等" (After real-name authentication, you can use AutoDL's more complete functions, such as opening the instance's self-defined service, etc.). Status: 未认证 (Not Authenticated) | 立即认证 (Authenticate Now).
- 微信绑定**: A note says "您已绑定了微信，可快速扫码登录" (WeChat has been bound, quick scan login available). Status: 已绑定 (Bound) | 解绑 (Unbind).

A red arrow points from the '立即认证' button in the '实名认证' section to a modal window titled '实名认证'.

Real-name Authentication Modal Window:

- 提示**: "为配合监管要求，防止将实例用于传播涉暴、涉黄等内容。使用自定义服务提供公网服务，需先进行实名认证。" (In accordance with regulatory requirements, to prevent instances from being used to spread violent or pornographic content. When providing public network services using custom services, real-name authentication is required first.)
- 去认证**: A blue button with a red border, indicating the next step in the process.
- 取消**: A white button.

Real-name Authentication Form:

- 真实姓名**: Input field placeholder: "请输入您的姓名" (Please enter your name).
- 身份证号**: Input field placeholder: "请输入您的身份证号" (Please enter your ID card number).
- 取消**: White button.
- 确定**: Blue button.

1.2 云端部署 - 部署SD操作界面

Step 7. 再次点击自定义服务

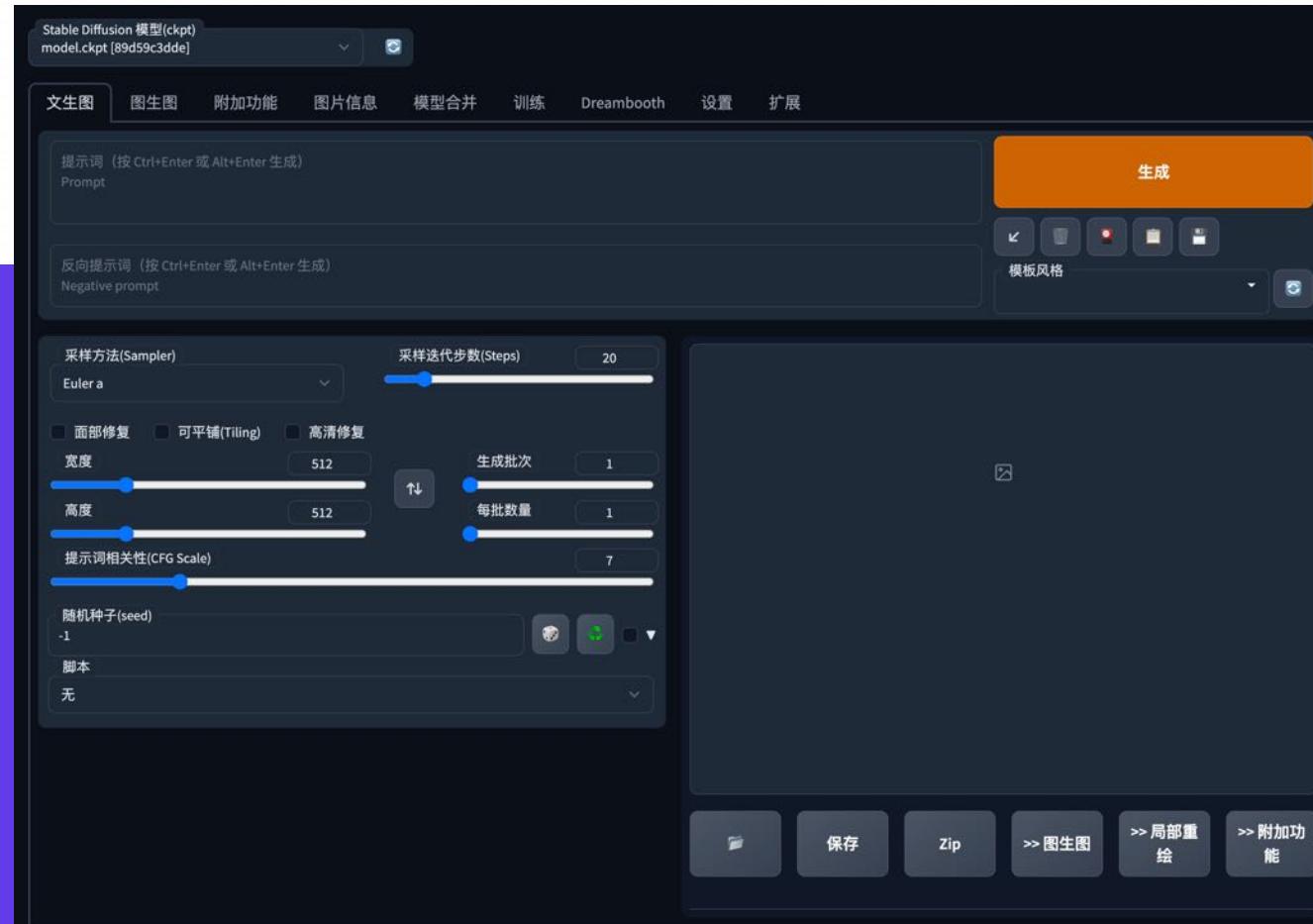
The screenshot shows the AutoDL platform's container instance management interface. On the left, there's a sidebar with links like '主页', '容器实例' (which is selected), '文件存储', '镜像', '公开数据', '费用', '收支明细', '我的订单', and '账单明细'. The main content area is titled '容器实例' and displays a table of instances. One instance is listed: '内蒙古A区 / 606机' with ID '45e411ab50-2e169 865', status '运行中' (Running). The table columns include '实例ID / 名称', '状态', '规格详情', '本地磁盘', '健康状态', '付费方式', '释放时间/停机时间', 'SSH登录', '快捷工具', and '操作'. In the '操作' column for this instance, there's a '登录指令' section with 'ssh****' and 'JupyterLab', followed by 'AutoPanel' and '关机' buttons. Below this, there's a '密码' section with '*****' and a '自定义服务' button, which is highlighted with a red box and a red arrow pointing to it.

如果您希望在此打开自定义服务，那么请将服务在实例中以6006端口启动后通过此入口访问
严禁使用WebUI等算法生成违禁图片，一经发现立即封号！



1.2 云端部署 - 部署SD操作界面

Step 8. 安装成功



到这里就云端部署完成了，可以出图啦。⚠ 温馨提示：反向词要用上，不要跑18+的图，会被封禁账号。

1.3 部署常见问题汇总

关机后再次使用时发现之前的服务器已被占用 -> **迁移镜像**

在我们租用一个实例之后，当我们将其关机后，下次再次使用时可能会没有可用的实例。然而，我们需要使用之前已经部署好的内容，而这些内容都保存在该实例上。一直等待并不是一个解决办法，所以我们需要再次租用一个可用的实例，并将数据迁移过去，以便将上一个实例中的内容都迁移到新实例上。

如果您在一个实例中配置了环境等内容，并且希望在更换主机或创建新实例时能够重复使用该环境，并长期保存以备将来使用，那么保存镜像将是最佳选择。此外，您还可以共享该镜像给其他用户。

1.3 部署常见问题汇总

迁移镜像

Step 1：保存镜像：将某个实例关机后，在更多操作中点击保存镜像，则可以将该实例的整个系统盘保存下来

The screenshot shows the 'Container Instance' management interface. At the top, there is a warning message: '实例连续关机30天会释放实例，实例释放会导致数据清空且不可恢复，释放前实例在数据在。' (If the instance is shut down for 30 consecutive days, it will be released. Releasing the instance will result in data being cleared and unrecoverable. The instance exists in data before release.) Below this, there is a '租用新实例' (Rent New Instance) button. The main table lists an instance with the following details:

实例ID / 名称	状态	规格详情	本地磁盘	健康状态	付费方式	释放时间/停机时间	SSH登录	更多操作
内蒙A区 / 606机 45e411ab50-2e169865	运行中	RTX A5000 * 1卡 查看详情	系统盘 59.34% 数据盘 13.93%	正常	按量计费 余额不足24小时	关机30天后释放 设置定时关机	登录指令 ssh***** 密码 *****	无卡模式开机 更换镜像 保存镜像 (highlighted with a red box) 升降配置 扩容数据盘 缩容数据盘 转包年包月 跨实例拷贝数据 重置系统 释放实例

1.3 部署常见问题汇总

迁移镜像

Step 2：保存镜像：将某个实例关机后，在更多操作中点击保存镜像，则可以将该实例的整个系统盘保存下来

The screenshot shows the AutoDL platform's interface. The left sidebar has navigation links: 首页, 容器实例, 文件存储, 镜像 (selected), 公开数据, 费用, 收支明细, 我的订单, and 账单明细. The main content area is titled '我的镜像'. It displays a message: '连续3个月未登录或欠费50元以上, 镜像将被清空.' Below this is storage usage information: '存储容量大小: 0.00GB (今天容量使用峰值0.00GB, 预计扣费0元)' and a link to '查看计费规则'. It also shows a breakdown of costs: '免费: 30.00GB' and '付费: 0GB'. A table lists images with columns: 镜像名称, 大小, 状态, 共享信息, 缓存地区, 创建时间, and 操作. One row for '123' is highlighted with a red box. The '操作' column for this row contains '编辑', '共享', and '删除'.

镜像名称	大小	状态	共享信息	缓存地区	创建时间	操作
123	14.83GB	保存中 4.14%	私有镜像	内蒙区	2023-05-12 10:59:12	编辑 共享 删除

1. 点【开机】，出弹窗，点击【保存镜像】，备注一个你能识别的名称
2. 重新租机器时，需要是同个区的

1.3 部署常见问题汇总

迁移镜像

Step 3：加载镜像：租用“新实例”或者“以前的实例”选择更换镜像时，选择自己保存的镜像，这样即可恢复原来实例系统盘中的所有内容



3. 再选择镜像时，选择【我的镜像】，找到你之前保存的镜像
4. 然后就是【开机】，再走一遍云部署的流程，这时候会报错
5. 需要再点【开机】，出弹窗，然后点击【跨实例拷贝数据】
6. 在跨实例拷贝数据的弹窗中，第1个选择框，把原先机器名字下方的一串数字+英文的字符，复制过来。第2个选择框输入“*”，然后点【确定】
7. 然后进入代码页，运行WebUi

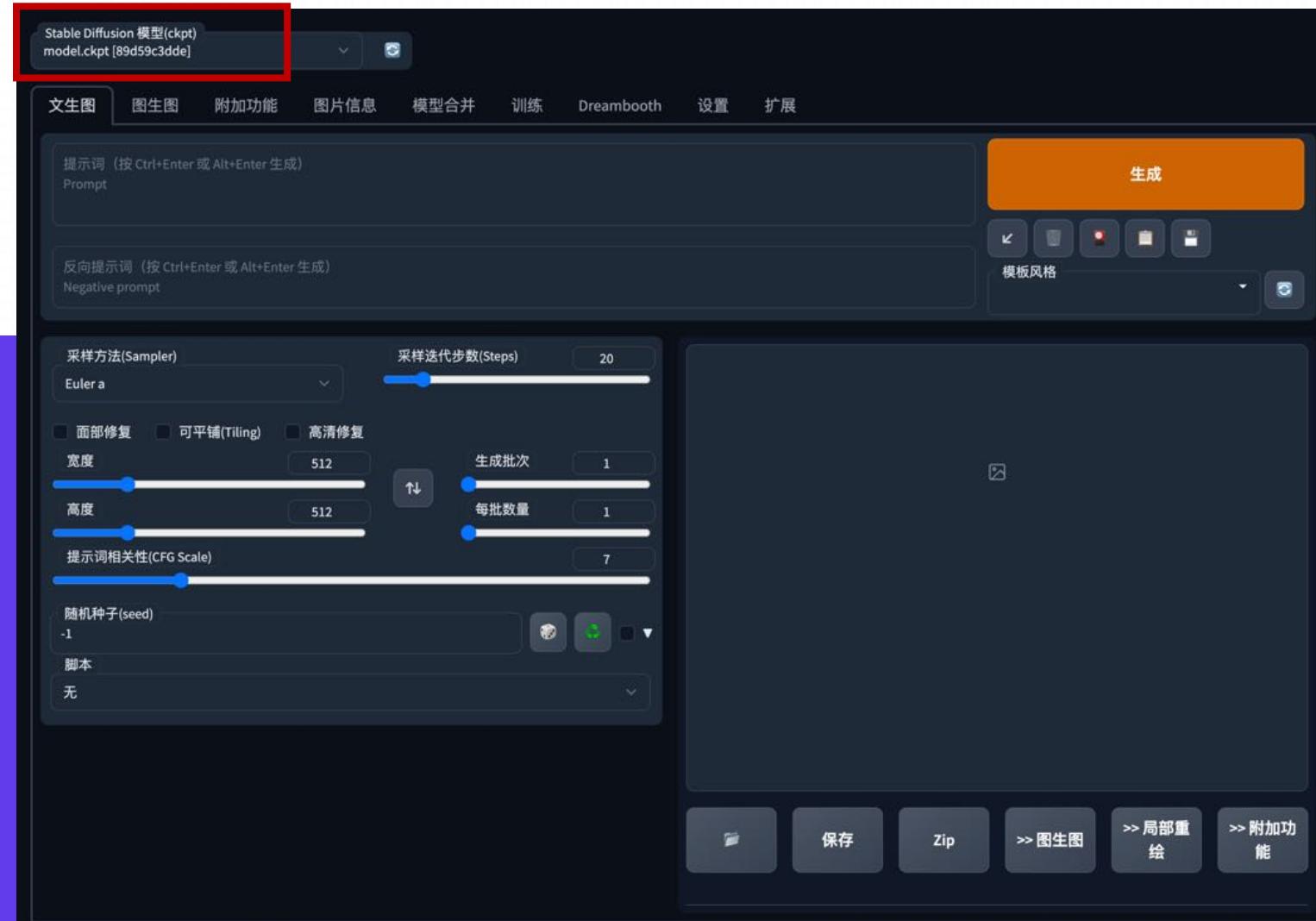
2. 给 Stable Diffusion 安装模型

2.1. **本地安装** (针对本地 Stable Diffusion)

2.2. **云端直接安装** (针对云端 Stable Diffusion)

2.3. **云端公网安装** (针对云端 Stable Diffusion)

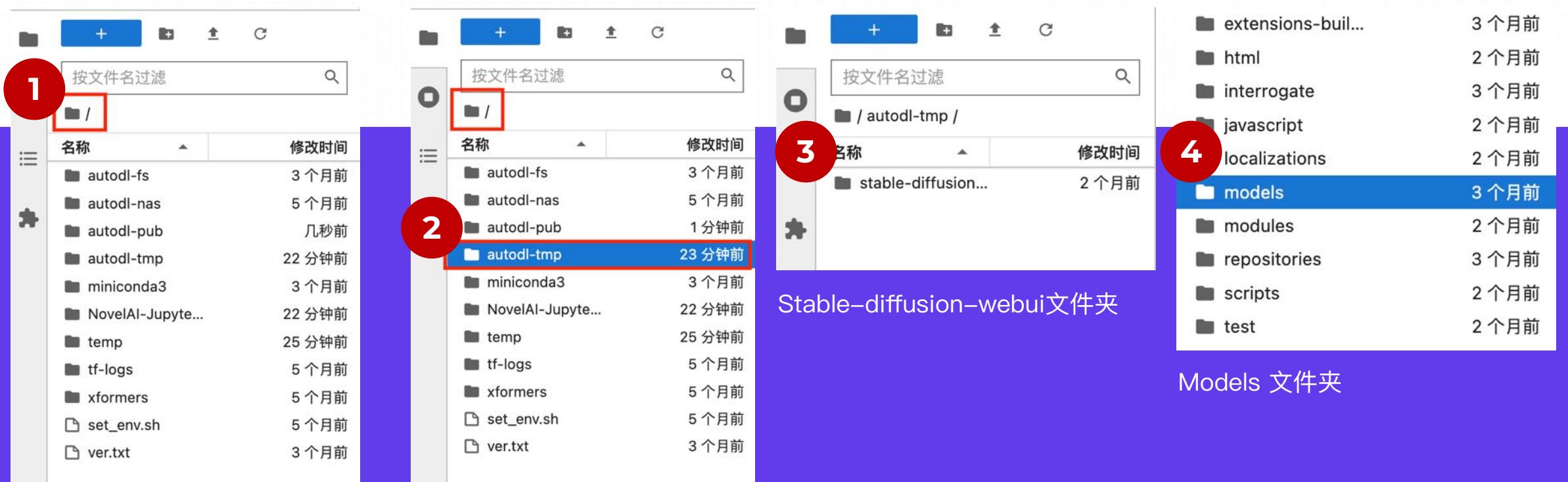
2.2 云端直接安装



Stable Diffusion部署完成，但是只有基础的大模型（页面的左上角，第一个功能区，这里就是大模型），我们要生成自己想要的风格和高质量的图，还是要安装其他大模型的，毕竟这才是使用工具最重要的部分啊，下面我们讲一下模型的安装

2.2 云端直接安装

安装模型，我们要知道模型都是在哪个文件夹下边存放的，右侧的文件区域，点击文件夹到根目录，具体操作如下图：



点击搜索框下边，文件目录的第一个文件夹

autodl-tmp就是sd的根目录

2.2 云端直接安装

安装模型，我们要知道模型都是在哪个文件夹下边存放的，右侧的文件区域，点击文件夹到根目录，具体操作如下图：



BLIP	3 个月前
Codeformer	3 个月前
deepbooru	2 个月前
dreambooth	2 个月前
ESRGAN	3 个月前
GFGAN	3 个月前
hypernetworks	3 个月前
LDSR	3 个月前
Lora	3 个月前
Stable-diffusion	2 个月前
SwinIR	3 个月前
torch_deeppan...	3 个月前
VAE	2 个月前
VAE-approx	2 个月前

5



/ ... / models / Stable-diffusion /	
名称	修改时间
Y: config.yaml	7 个月前
model.ckpt	7 个月前
model.vae.pt	7 个月前
Put Stable Diffu...	2 个月前

6

Stable Diffusion文件夹：是所有模型所在的文件夹，包括大模型，VAE模型

这个文件夹内部就是大模型的存放目录，可以将本地的模型上传到这里

把大模型文件拖到stable diffusion文件夹下即可，拷贝完成以后，到stable diffusion的页面去刷新下模型，然后下拉选择就好了。

2.3 云端公网安装

上面我们本地直接安装，由于大模型的文件都非常大，最少也要2G，传输非常慢，所以我们用另一种方法安装，就是通过公网网盘安装。也就是百度云盘/阿里云盘安装，他们云端对云端的传输，速度比较快。

优点：

1. 支持实例和百度网盘、阿里云盘等直接进行数据传输（实例上传到网盘，网盘下载到实例），方便下载数据到不同实例、备份实例重要数据、迁移实例。阿里云盘在海外的传输速度依然很快，如果您在海外使用 AutoDL 非常推荐～
2. 相比本地和实例之间传输，速度快且稳定
3. 和其他人共享数据集

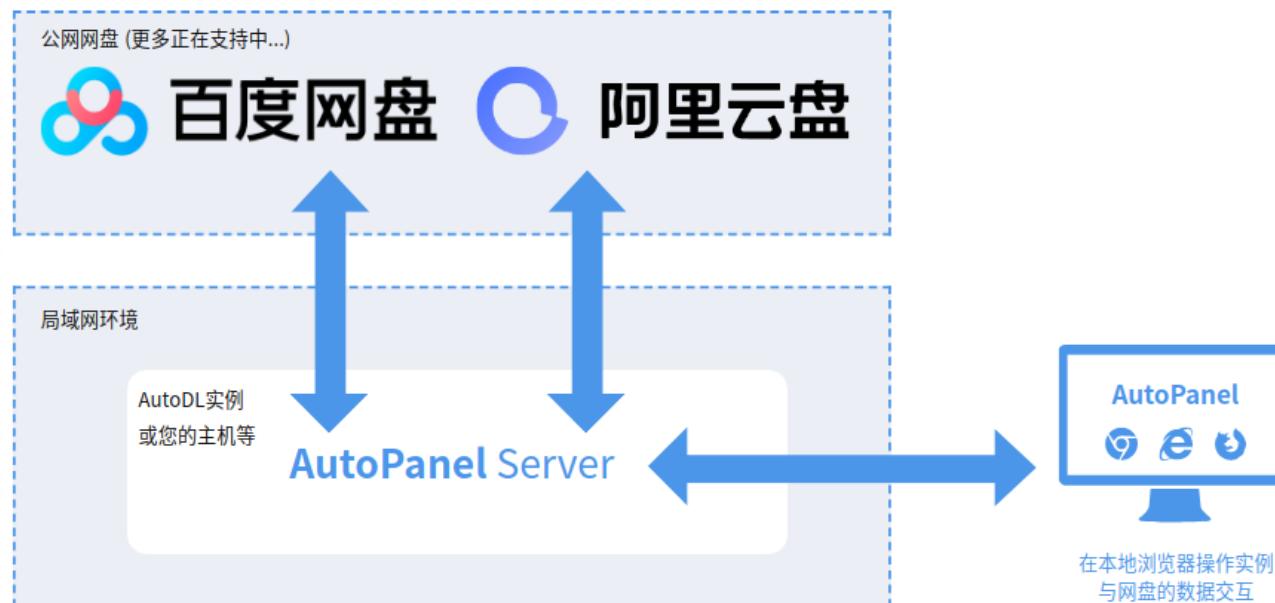
缺点：

1. 需要额外配置授权算一个缺点

2.3 云端公网安装

1. 公网网盘

AutoDL中公网网盘的实现可看作三方网盘客户端（叫做AutoPanel），和官方客户端的不同有：该工具为Server/Browser架构（类比JupyterLab理解，Server可以不在本地电脑，但可以在本地浏览器使用）。架构如下：



从图中可以看出：与官方客户端不同，AutoPanel的Server启动在哪台主机，那么就由该主机和网盘发生上传下载交互，和浏览器所在主机无关。因此您可以更灵活的将AutoPanel的Server启动在本地电脑XQ、AutoDL的实例(已默认启动)、或其他您自己的服务器当中，然后通过Server提供的访问地址在浏览器完成操作。

2.3 云端公网安装

2. 使用方法

Step.1：打开AutoPanel

租用实例开机后，在快捷工具一列找到AutoPanel入口，点击打开



我的实例

① 非潮汐算力实例连续关机30天会释放实例，实例释放会导致数据清空且不可恢复，释放前实例在数据在。潮汐算力实例连续关机3天则释放实例。

实例ID / 名称	状态	规格详情	健康状态	付费方式	释放时间/停机时间	登录指令	快捷工具
875111ae0c-976af8db	• 运行中	TitanXp * 1卡 查看详情	• 正常	按量计费	关机3天后释放 设置定时关机	登录指令 ssh***** 密码 *****	JupyterLab AutoPanel 实例监控
875111ae0c-c08c25cc	• 运行中	TitanXp * 1卡 查看详情	• 正常	按量计费	关机3天后释放 设置定时关机	登录指令 ssh***** 密码 *****	JupyterLab AutoPanel 实例监控

2.3 云端公网安装

2. 使用方法

Step.2：授权

第一步需要设置独立访问密码，该密码非常重要（更复杂更安全），您接下来授权的网盘个人账户信息均通过该密码进行加密。如忘记密码则需重新初始化(执行: `rm -rf /root/autodl-tmp/.autodl` 然后重启 AutoPanel: `ps -ef | grep autopanel | awk '{print $2}' | xargs kill -9`)。

通过自己设置的独立密码加密后的个人信息只有通过自己的密码才能解密，AutoDL无法解密。

2.3 云端公网安装

2. 使用方法

Step.2：授权

阿里云盘(推荐，限制更小、更快)：选择阿里云盘，然后使用阿里云盘的APP扫码授权登录。



2.3 云端公网安装

2. 使用方法

Step.2：授权

百度网盘：登录百度网盘后，创建应用（如果您没有进行过百度的认证可能创建失败，需先完成认证），创建完成后获取AppKey和SecretKey。以下应用名称推荐填写AutoDL，以免不必要的麻烦

The screenshot shows the process of creating a new application on the AutoDL platform. On the left, there's a form with fields for '应用类别' (Software), '应用名称' (AutoDL), and '应用描述' (Description: For AutoDL network disk authorization). A large black redaction box covers the top half of the page. To the right, a green circular success icon is displayed above the message '应用创建成功'. Below this, a note states: 'Appid、Appkey、Secretkey、Signkey是您应用实际开发的主要凭证，每个应用唯一标识，互不相同，请妥善保管。' (Appid, Appkey, Secretkey, Signkey are the main certificates for your application development, unique for each application, please keep them safe.) A summary box on the right lists the generated credentials: AppID: 25814517, Appkey: mpBC [REDACTED], Secretkey: mjel [REDACTED], and Signkey: D3CU [REDACTED]. The Appkey and Secretkey fields are highlighted with a red rectangle. At the bottom, there are three buttons: '确认创建' (Confirm Creation), '接入能力' (Integration Capabilities), and '查看应用' (View Application).

2.3 云端公网安装

2. 使用方法

Step.2：授权

在AutoPanel点击添加授权，输入AppKey和SecretKey后，点击获取Access Token将打开百度网盘的网页，将其返回的Token粘贴过来即可完成授权。



2.3 云端公网安装

2. 使用方法

Step.3: 文件浏览/上传/下载

如果是超过10GB以上的大文件在上传时可能需要花几十秒到几分钟不等的时间计算文件Hash值，计算完成后才会开始上传。另外计算 Hash值的效率与cpu/内存相关，如果使用了无卡模式，该过程会更慢。请耐心等待~

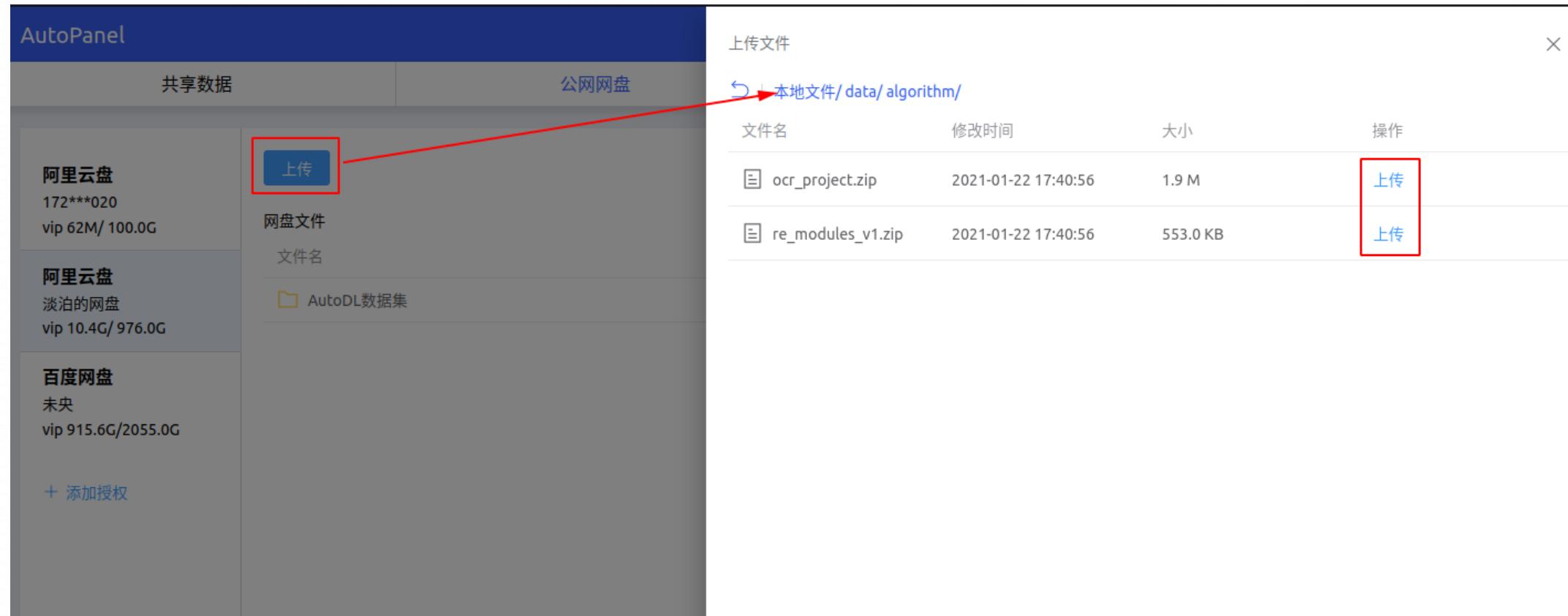
文件名	修改时间	大小	操作
AutoDL数据集	2022-03-21 18:17:40	-	-
re_modules_v1.zip	2022-03-22 17:00:59	553.0 KB	下载
ocr_project.zip	2022-03-22 16:59:48	1.9 M	下载

选中网盘后可以浏览网盘中的文件，
点击下载可以下载至本地（实例中）

2.3 云端公网安装

2. 使用方法

Step.3：文件浏览/上传/下载 点击上传按钮则可以上传本地（实例中）的文件至网盘



2.3 云端公网安装

2. 使用方法

Step.3：文件浏览/上传/下载 同时可存在多个上传和下载文件任务，他们将排队进行上传下载，在右侧「传输任务」中可查看

The screenshot shows the AutoDL Cloud Public Network Disk interface. At the top, there is a blue header bar with a back arrow icon. Below the header, there is a navigation bar with three tabs: '公网网盘' (Public Network Disk), 'TensorBoard' (selected), and '实用工具' (Utility Tools). On the right side of the interface, there is a '传输任务' (Transfer Task) section with a table showing four upload tasks:

网盘	文件	类型	速度	进度	操作
阿里云盘-172***020	[Redacted]	上传成功	-	-	
阿里云盘-172***020	[Redacted]	上传成功	-	-	
阿里云盘-淡泊的网盘	ocr_project.zip	上传成功	-	-	
阿里云盘-淡泊的网盘	re_modules_v1.zip	上传成功	-	-	

2.3 云端公网安装

3. 功能说明

1. 同时支持多个网盘的授权管理
2. 阿里云盘和百度网盘均支持断点续下载，断点上传正在支持中
3. 阿里云盘支持秒传
4. 上传下载速度一般为5~20MB/s，具体与当时网络负载有关（百度网盘与会员有关）
5. 网盘官方关键限制说明：

一. 百度网盘：

1. 速度限制：与会员等级相关。实测非会员下载1MB/s, 上传10MB/s。会员上传下载10MB/s+
2. 上传路径限制：只能上传文件至[我的应用数据(部分账户叫apps)]目录，其他目录无权限。下载时无路径限制
3. 上传文件大小限制：
 - 普通用户单个上传文件大小上限为4GB
 - 会员用户单个上传文件大小上限为10GB
 - 超级会员用户单个上传文件大小上限为20GB

二. 阿里云盘：

1. 上传文件大小限制
 - 普通用户单个上传文件大小上限为100GB
 - 会员用户单个上传文件大小上限为200GB
 - 超级会员用户单个上传文件大小上限为1TB

2.3 云端公网安装

4. 自动授权

上述授权过程需要在每个新创建的实例中重复添加，如果您希望只需添加授权一次，后续创建的实例均自动使用该授权，那么可以在「控制台」的「设置」菜单下托管授权信息到AutoDL，由AutoDL帮您完成添加授权：

The screenshot shows the AutoDL control panel's 'Settings' section. On the left, there is a sidebar with various menu items: '主页' (Home), '实例与数据' (Instances & Data) which has '我的实例' (My Instances), '我的网盘' (My Cloud Drives), '公开数据' (Public Data), '我的镜像' (My Images), '费用中心' (Cost Center) which has '我的订单' (My Orders), '我的账单' (My Bills), '代金券' (Coupons), '优惠券' (Promotional Vouchers), '发票管理' (Invoice Management), '账号中心' (Account Center) which has '账号安全' (Account Security) and '子账号' (Sub-Accounts), and '设置' (Settings). The '设置' item is highlighted with a blue bar at the bottom. In the main content area, under '个人Token' (Personal Token), there is a tab labeled '公网网盘' (Cloud Public Network) which is also highlighted with a red box. A callout box with a red border contains instructions: '① 您在此导入阿里云盘, 百度网盘等的授权后, 将由系统在实例开机时为AutoPanel添加授权。导入授权方法: Step1. 在实例的AutoPanel中添加公网网盘的授权 Step2. 在AutoPanel导出授权文件(因此该授权今后需要您的独立访问密码解密, AutoDL只托管授权文件无法解密) Step3. 将第2步导出的授权信息在此导入'. Below this, there is a text input field with placeholder '请输入您导出的授权信息' (Please enter the exported authorization information) and a blue '确定' (Confirm) button.

3. Stable Diffusion 的原理

3.1. Stable Diffusion 组件

3.2. 什么是 Stable Diffusion?

3.3. Diffusion 的工作原理

3.4. 移除噪声，绘制图像

3.5. 加速：在压缩数据上扩散

3.6. 文本编码器：一个Transformer语言模型

3.7. 文本信息喂入图像生成过程

3.1. Stable Diffusion组件

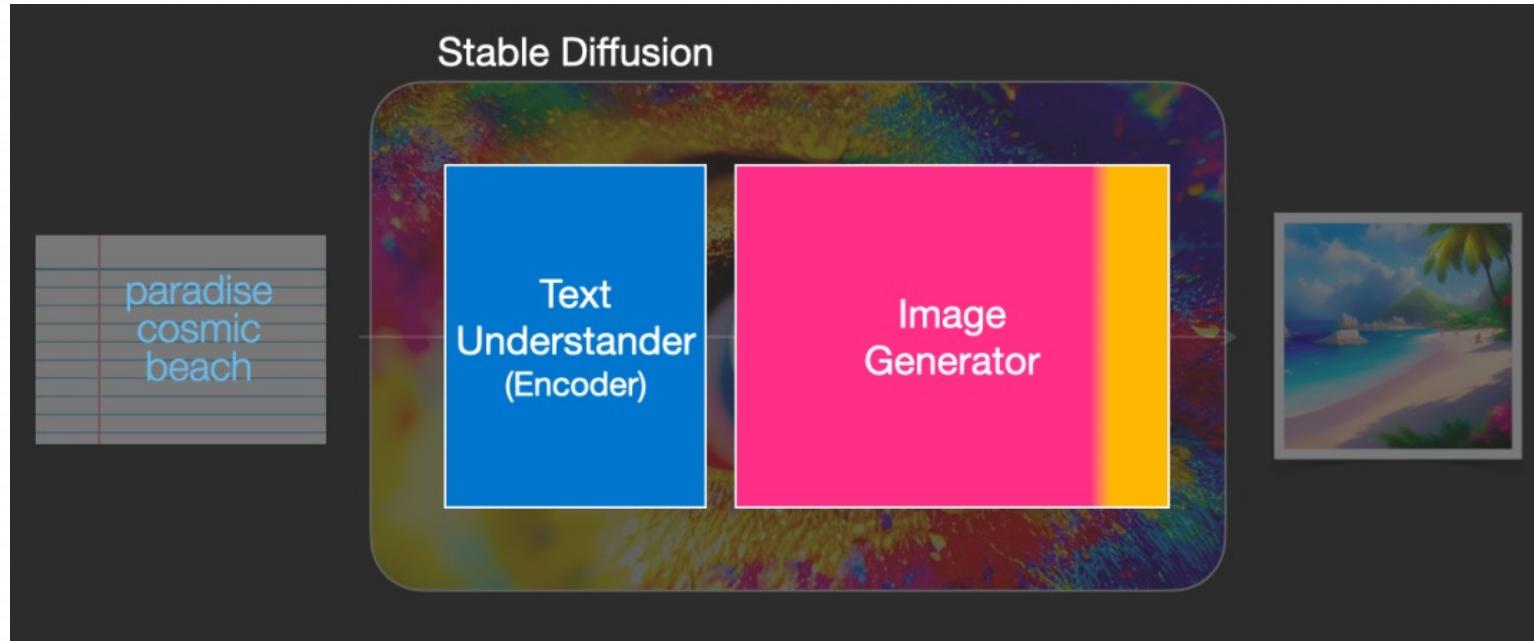
Stable Diffusion从功能上来说主要包括两方面：

- 核心功能为根据文本提示作为输入来生成的图像 (text2img)
- 可以用它对图像根据文字描述进行修改 (即输入为文本+图像)



3.1. Stable Diffusion组件

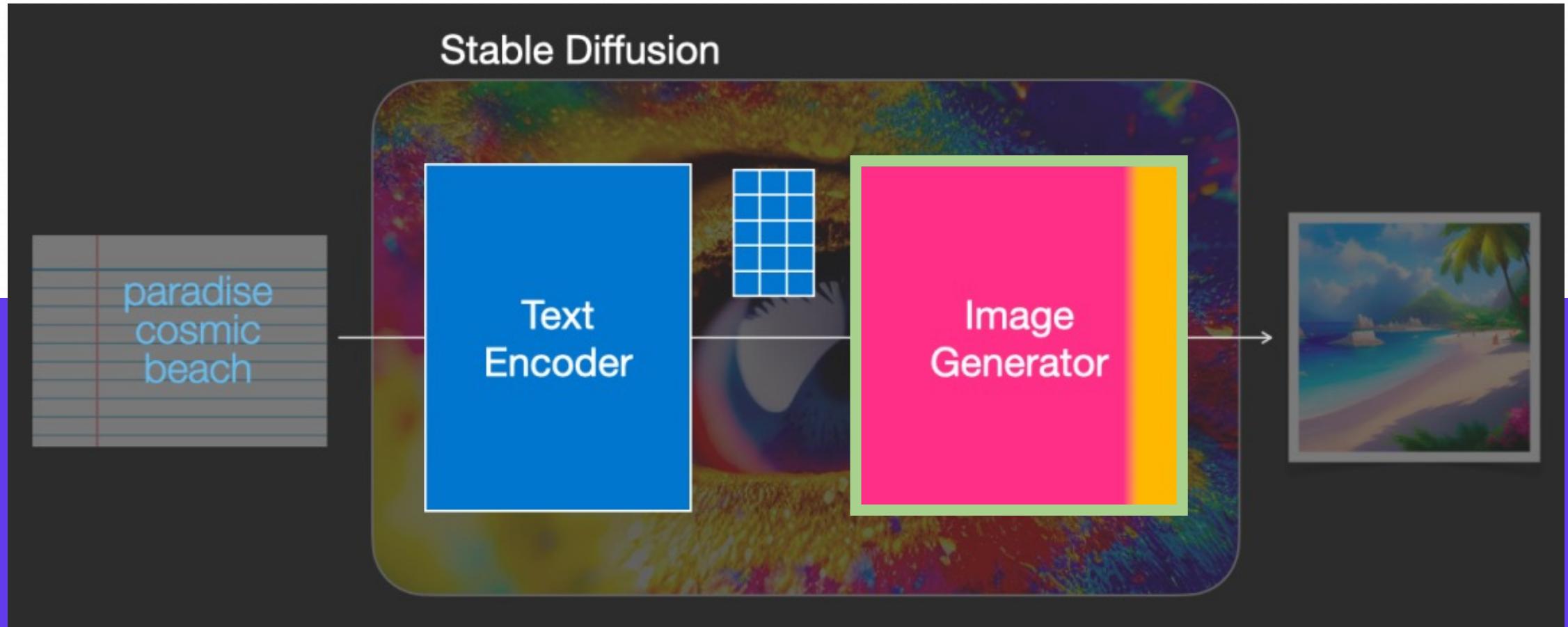
Stable Diffusion是一个由多个组件和模型组成的系统，而非单一的模型。当我们从模型整体的角度向模型内部观察时，可以发现，其包含一个文本理解组件用于将文本信息翻译成数字表示（numeric representation），以捕捉文本中的语义信息。



虽然目前还是从宏观角度分析模型，后面才有更多的模型细节，但我们可以大致推测这个文本编码器是一个特殊的Transformer语言模型（具体来说是CLIP模型的文本编码器）。

3.1. Stable Diffusion组件

模型的输入为一个文本字符串，输出为一个数字列表，用来表征文本中的每个单词/token，即将每个token转换为一个向量。然后这些信息会被提交到图像生成器（image generator）中，它的内部也包含多个组件。

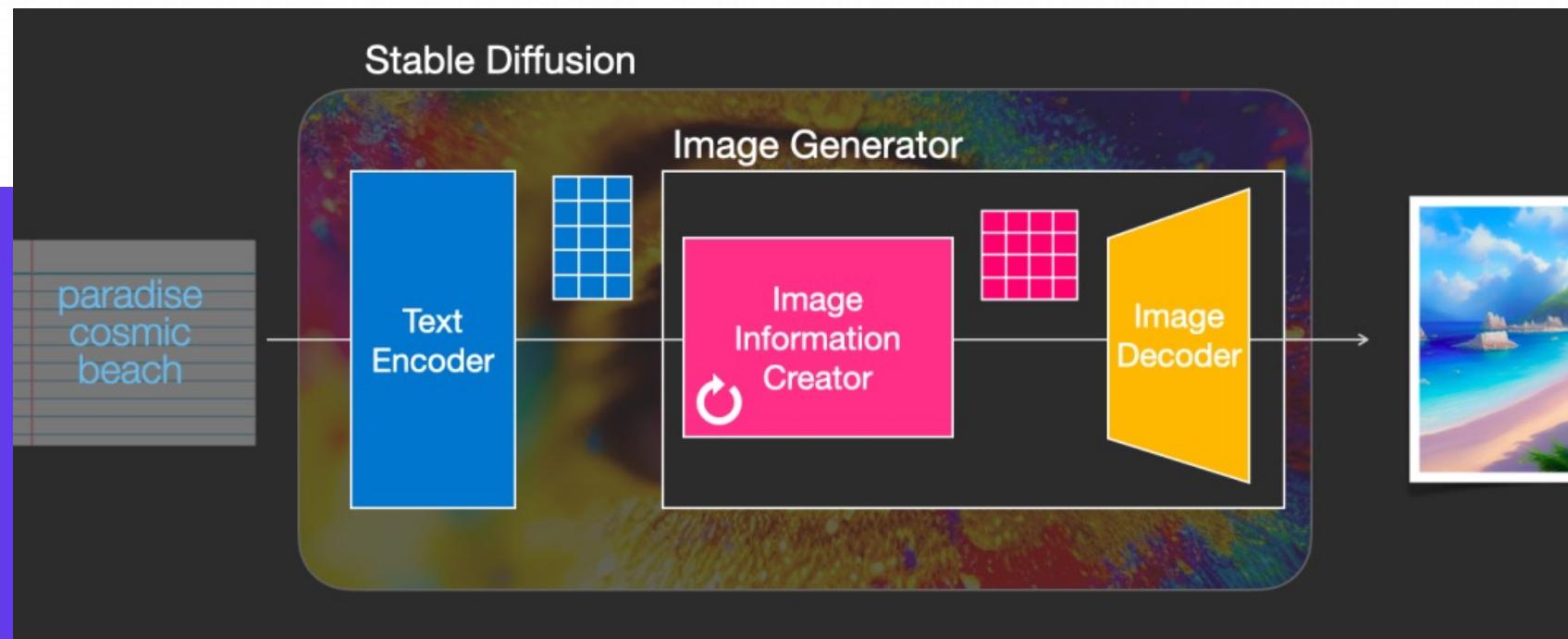


3.1. Stable Diffusion组件

图像生成器主要包括两个阶段： **Image information creator** 和 **图像解码器**

1. Image information creator

该组件运行多个 steps 来生成图像信息，其中 steps 也是Stable Diffusion接口和库中的参数，通常默认为50或100。

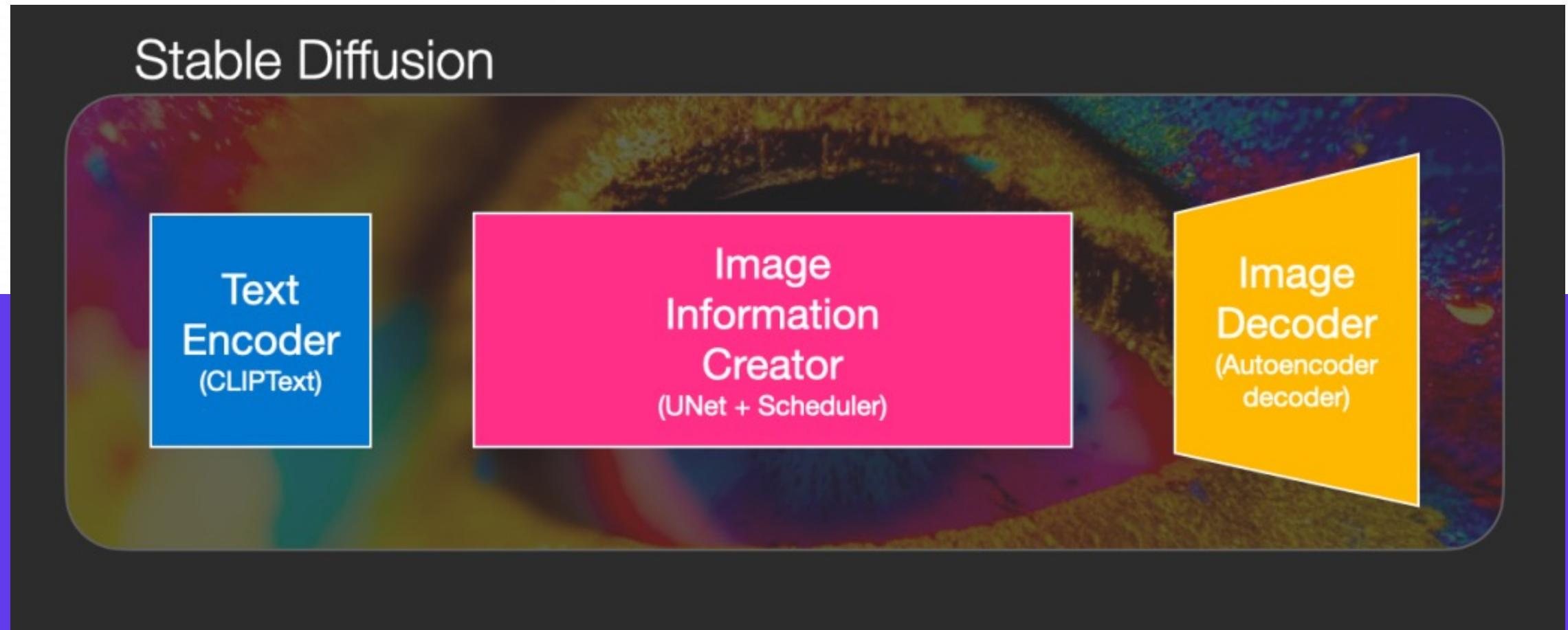


图像信息创建器完全在图像信息空间（或潜空间）中运行，这一特性使得它比其他在像素空间工作的Diffusion模型运行得更快；从技术上来看，该组件由一个UNet神经网络和一个调度（scheduling）算法组成。

3.1. Stable Diffusion组件

2. 图像解码器

图像解码器根据从图像信息创建器中获取的信息画出一幅画，整个过程只运行一次即可生成最终的像素图像。



3.1. Stable Diffusion组件

2. 图像解码器

Stable Diffusion总共包含三个主要的组件，其中每个组件都拥有一个独立的神经网络：

1) Clip Text用于文本编码

- 输入：文本
- 输出：77个token嵌入向量，其中每个向量包含768个维度

2) UNet + Scheduler在信息（潜）空间中逐步处理/扩散信息

- 输入：文本嵌入和一个由噪声组成的初始多维数组（结构化的数字列表，也叫张量tensor）。
- 输出：一个经过处理的信息阵列

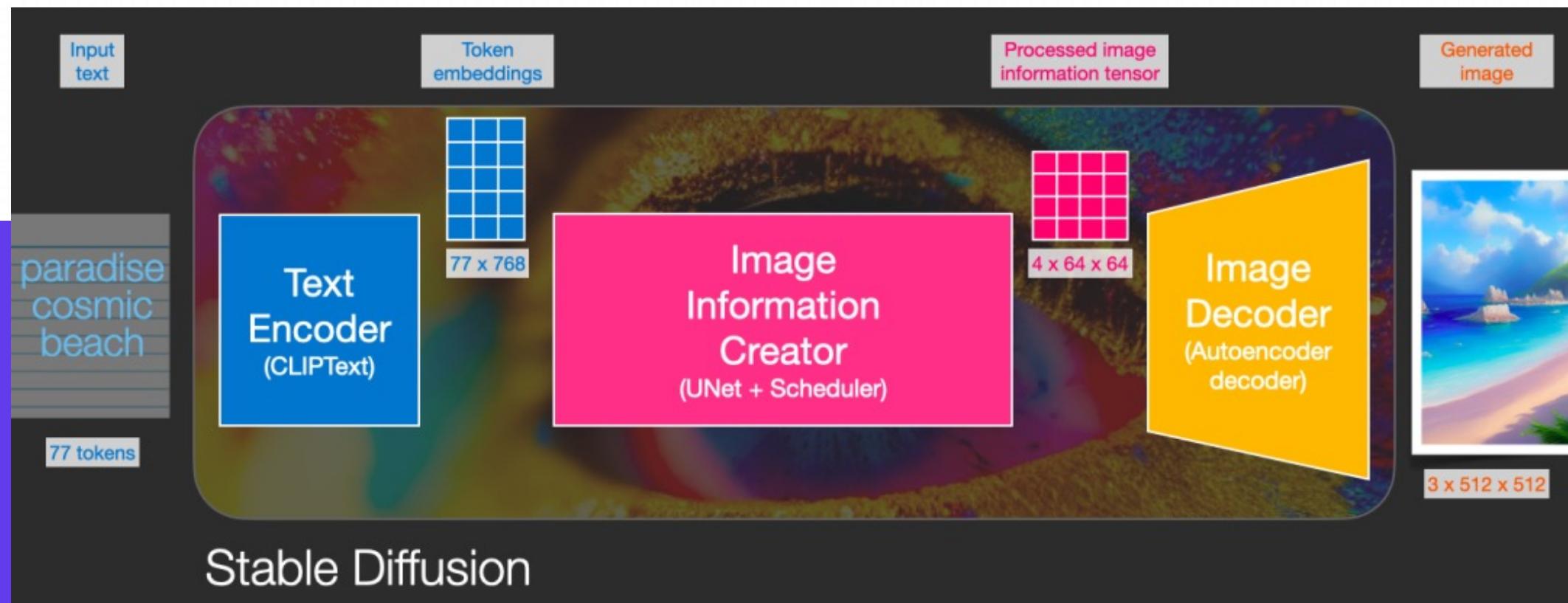
3) 自编码解码器（Autoencoder Decoder），使用处理过的信息矩阵绘制最终图像的解码器

- 输入：处理过的信息矩阵，维度为(4, 64, 64)
- 输出：结果图像，各维度为(3, 512, 512)，即(红/绿/蓝, 宽, 高)

3.1. Stable Diffusion组件

2. 图像解码器

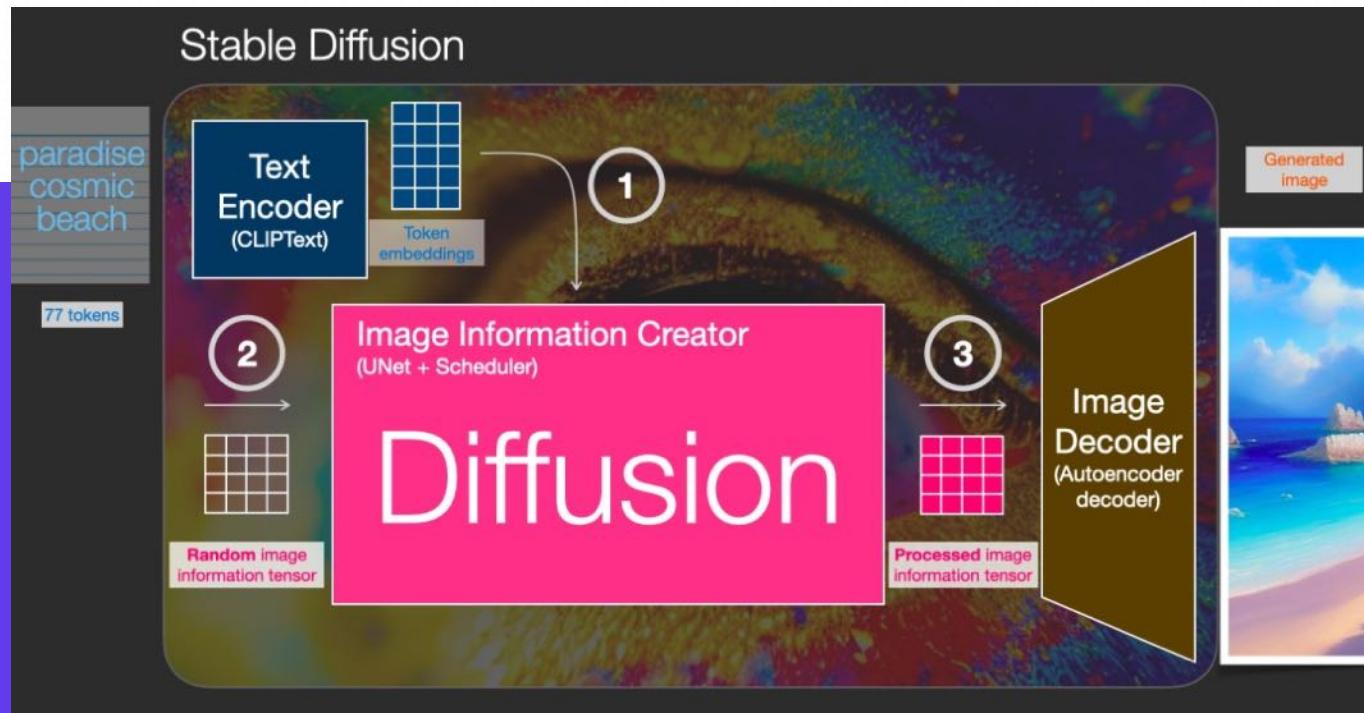
Stable Diffusion总共包含三个主要的组件，其中每个组件都拥有一个独立的神经网络：



3.2. 什么是Stable Diffusion?

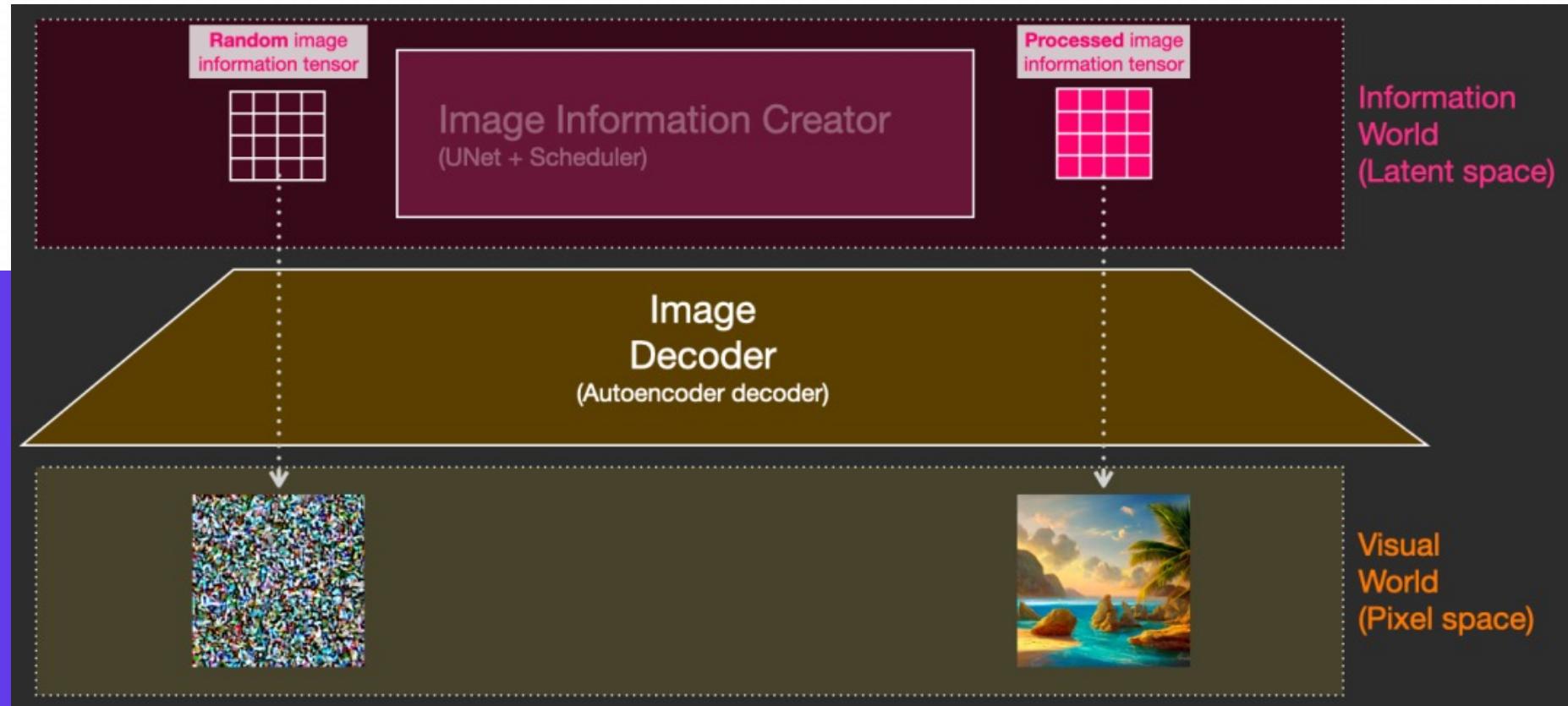
扩散 (diffusion) 这个词描述了在该组件内部运行期间发生的事情，即对信息进行一步步地处理，并最终由下一个组件（图像解码器）生成高质量的图像。

扩散是在下图中粉红色的图像信息创建器组件中发生的过程，过程中包含表征输入文本的token嵌入，和随机的初始图像信息矩阵（也称之为latents），该过程会还需要用到图像解码器来绘制最终图像的信息矩阵。



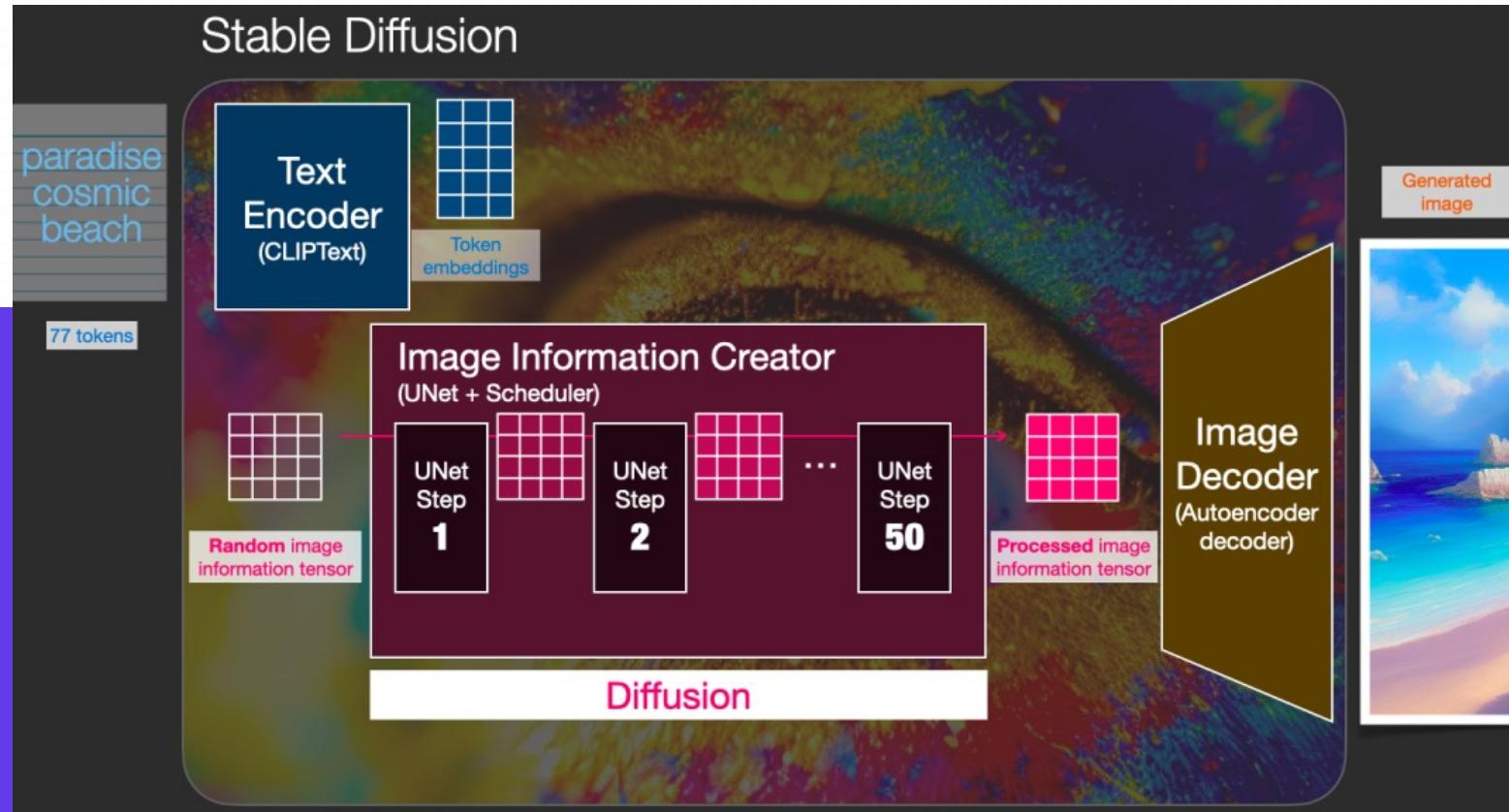
3.2. 什么是Stable Diffusion?

整个运行过程是step by step的，每一步都会增加更多的相关信息。为了更直观地感受整个过程，可以中途查看随机latents矩阵，并观察它是如何转化为视觉噪声的，其中视觉检查（visual inspection）是通过图像解码器进行的。



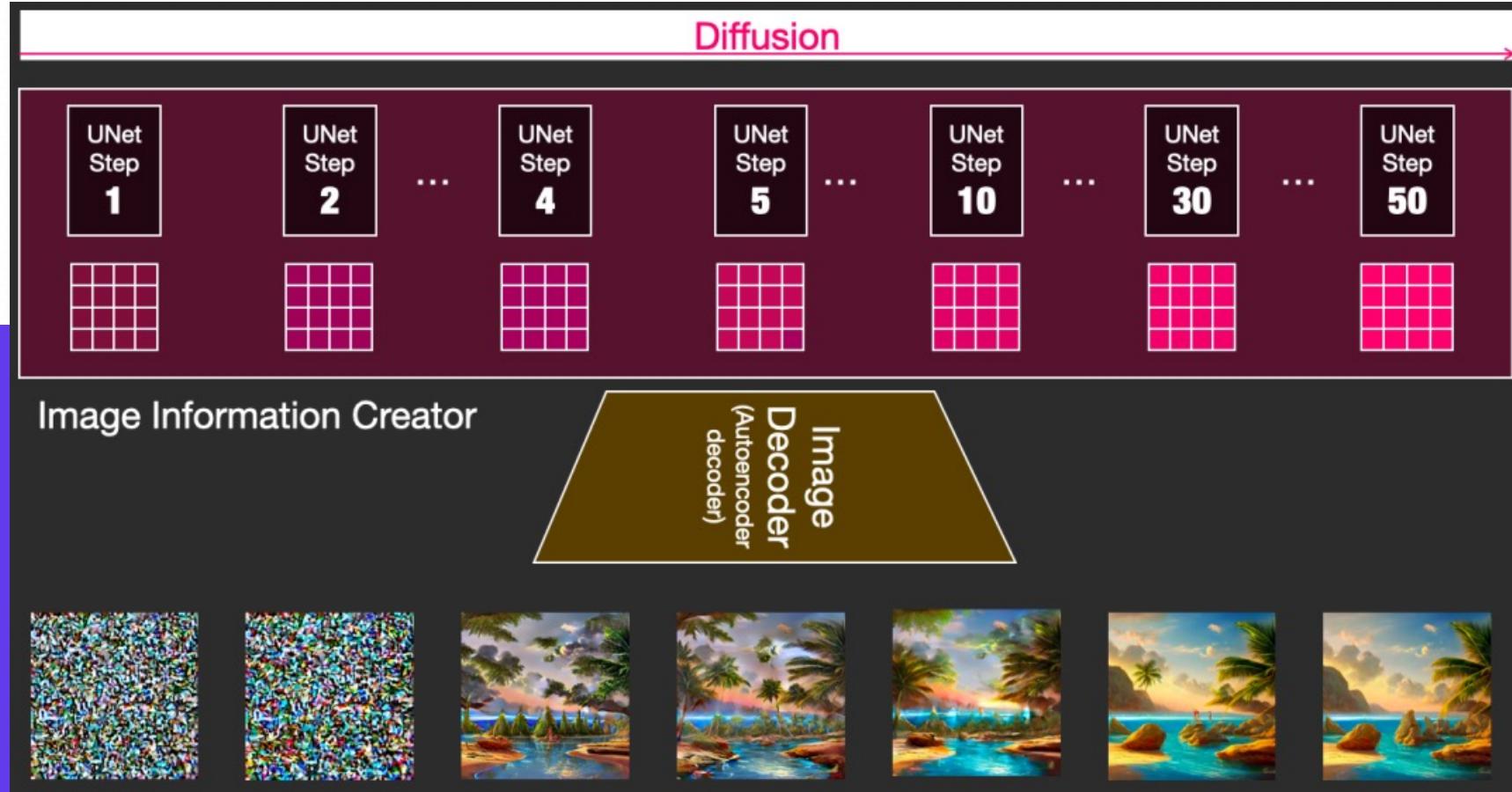
3.2. 什么是Stable Diffusion？

整个Diffusion过程包含多个steps，其中每个 step 都是基于输入的 **latents 矩阵**进行操作，并生成另一个 **latents 矩阵**以更好地贴合「输入的文本」和从模型图像集中获取的「视觉信息」。



3.2. 什么是Stable Diffusion?

将这些latents可视化可以看到这些信息是如何在每个step中相加的。

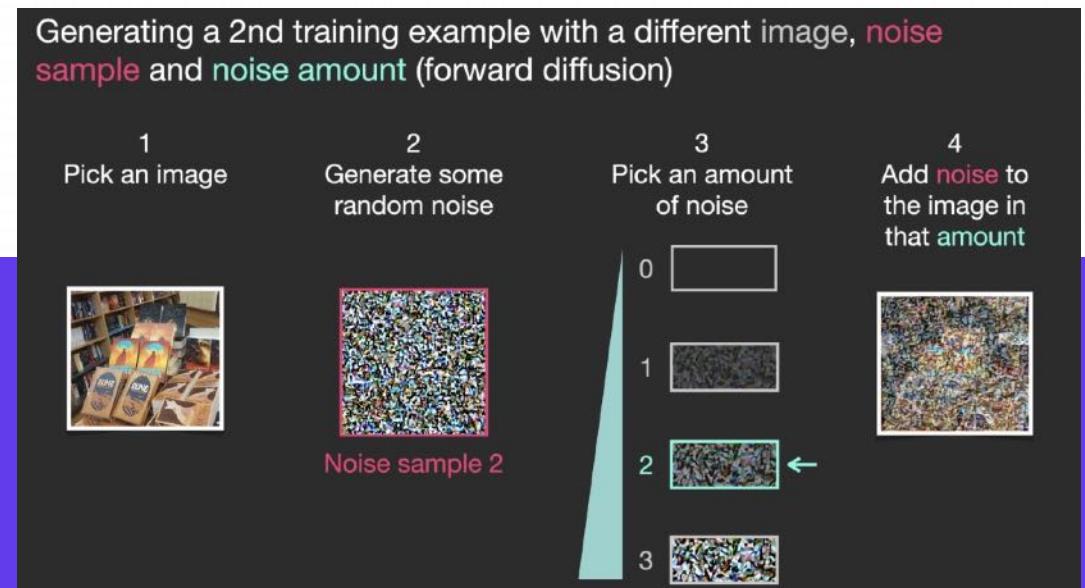
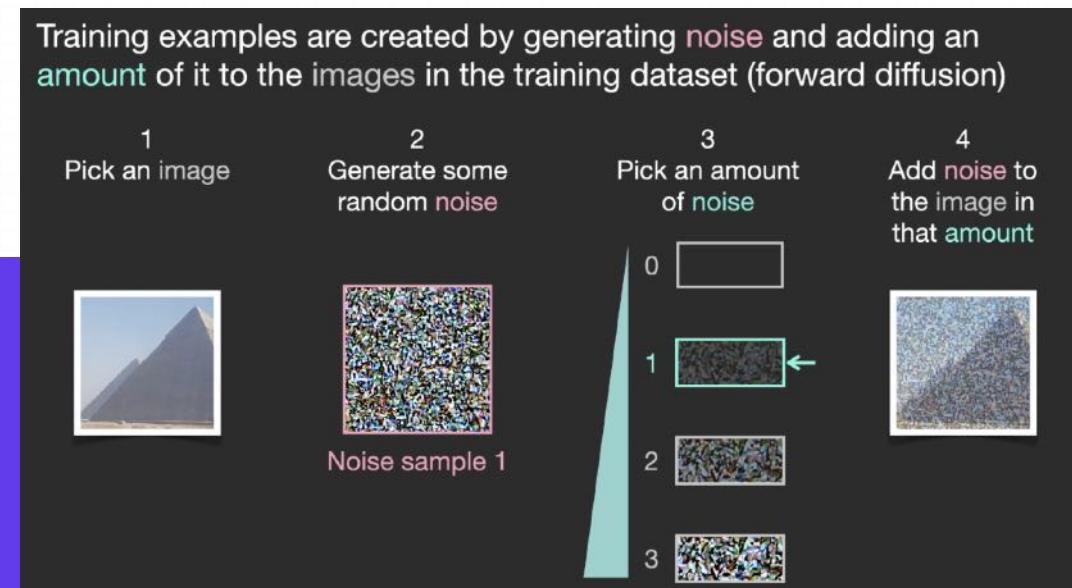


3.3. Diffusion 的工作原理

使用扩散模型生成图像的核心思路还是基于已存在的强大的计算机视觉模型，只要输入足够大的数据集，这些模型可以学习任意复杂的操作。

假设我们已经有了一张图像，生成产生一些噪声加入到图像中，然后就可以将该图像视作一个训练样例。

使用相同的操作可以生成大量训练样本来训练图像生成模型中的核心组件。



上述例子展示了一些可选的噪声量值，从原始图像(级别0，不含噪声)到噪声全部添加(级别4)，从而可以很容易地控制有多少噪声添加到图像中。

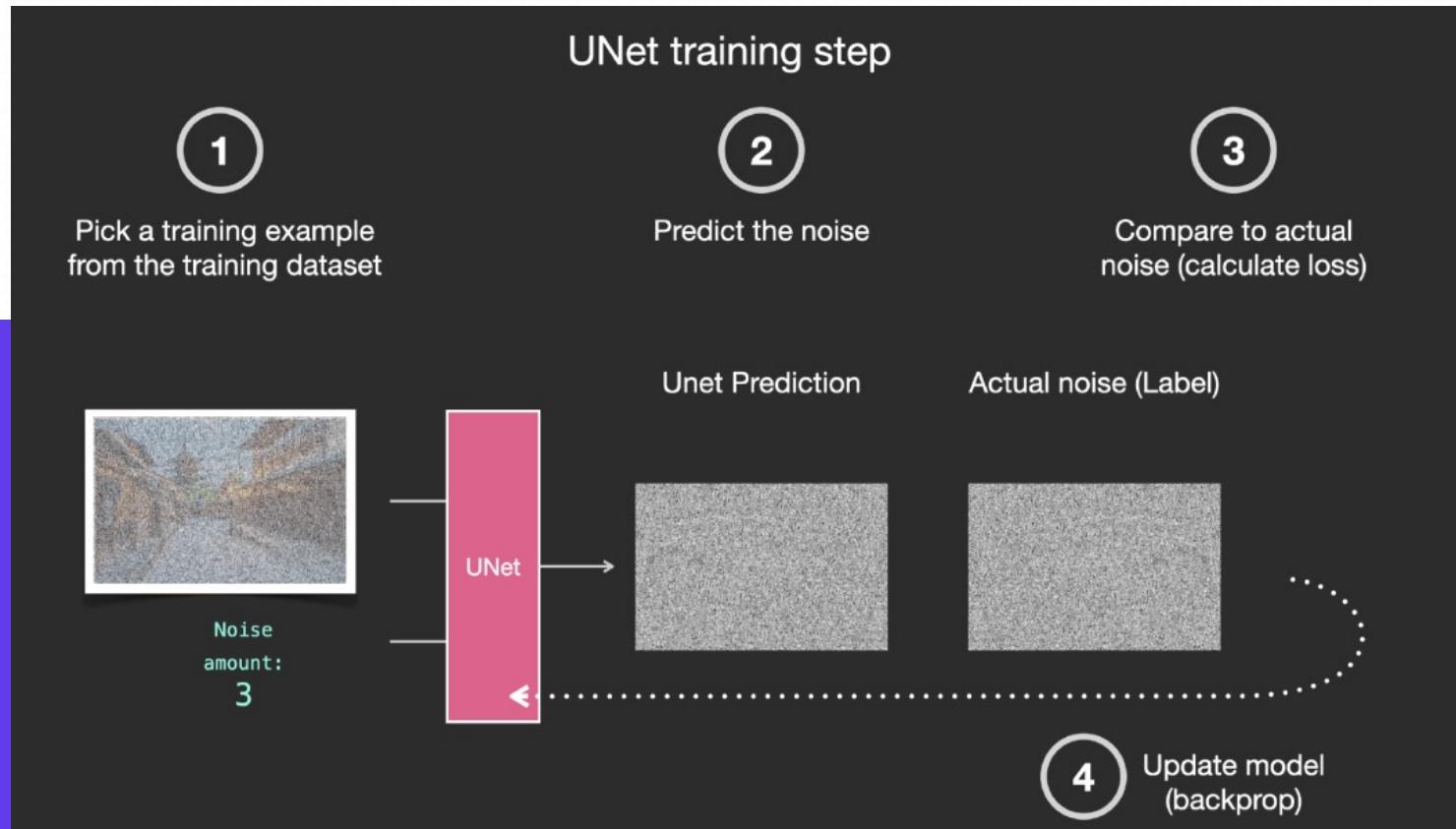
3.3. Diffusion 的工作原理

我们可以将这个过程分散在几十个steps中，对数据集中的每张图像都可以生成数十个训练样本。



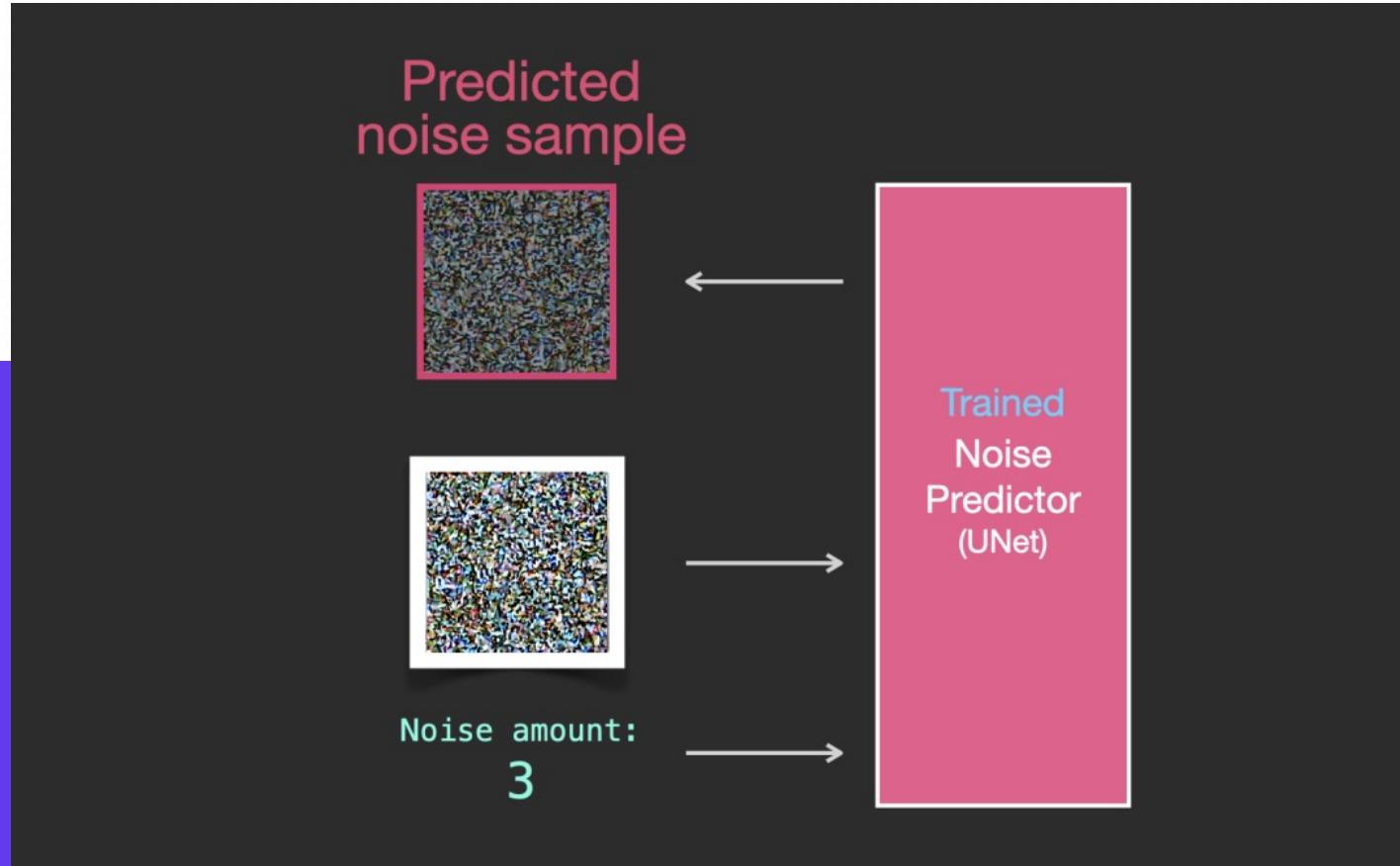
3.3. Diffusion 的工作原理

基于上述数据集，我们就可以训练出一个性能极佳的噪声预测器，每个训练step和其他模型的训练相似。当以某一种确定的配置运行时，噪声预测器就可以生成图像。



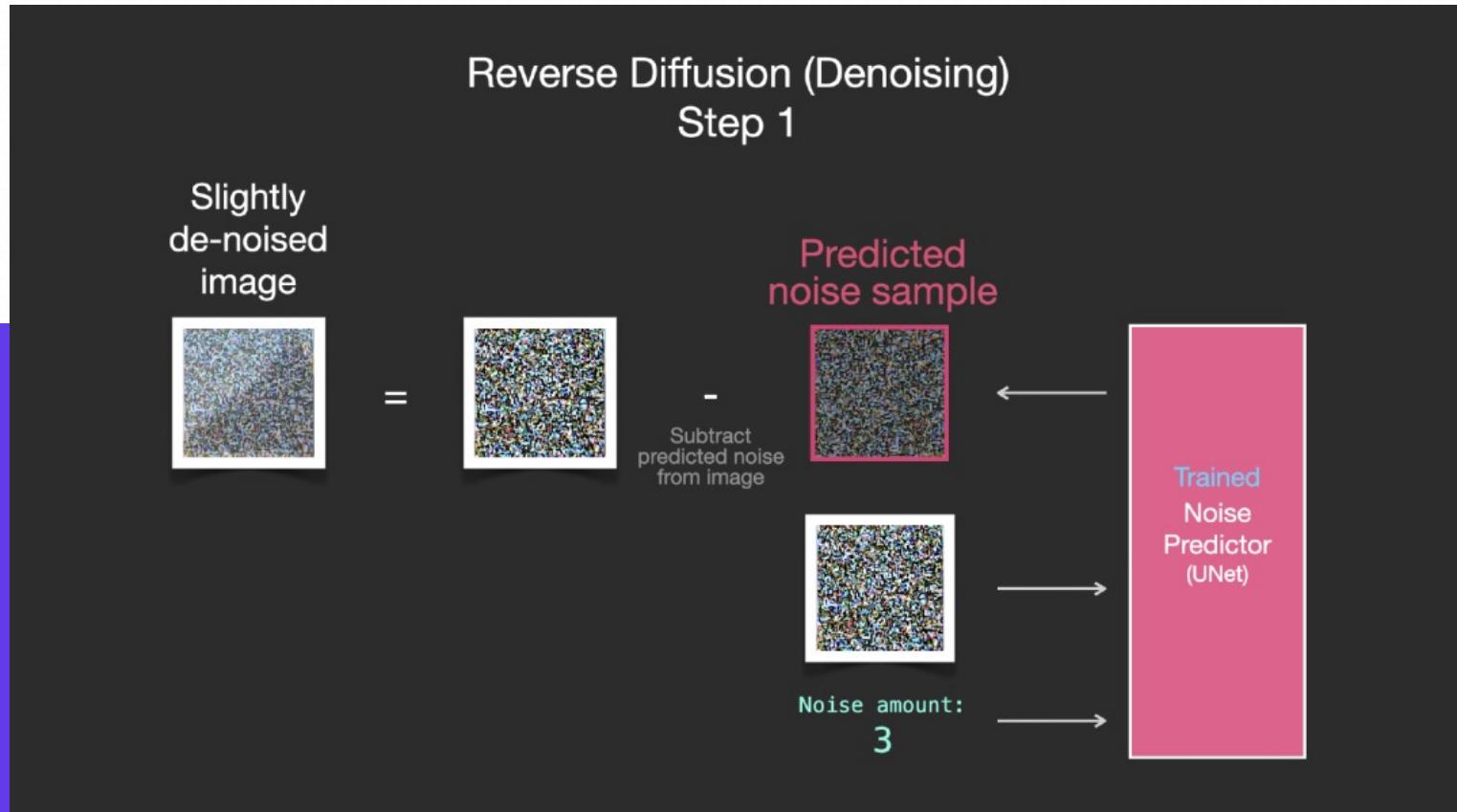
3.4. 移除噪声，绘制图像

经过训练的噪声预测器可以对一幅添加噪声的图像进行去噪，也可以预测添加的噪声量。



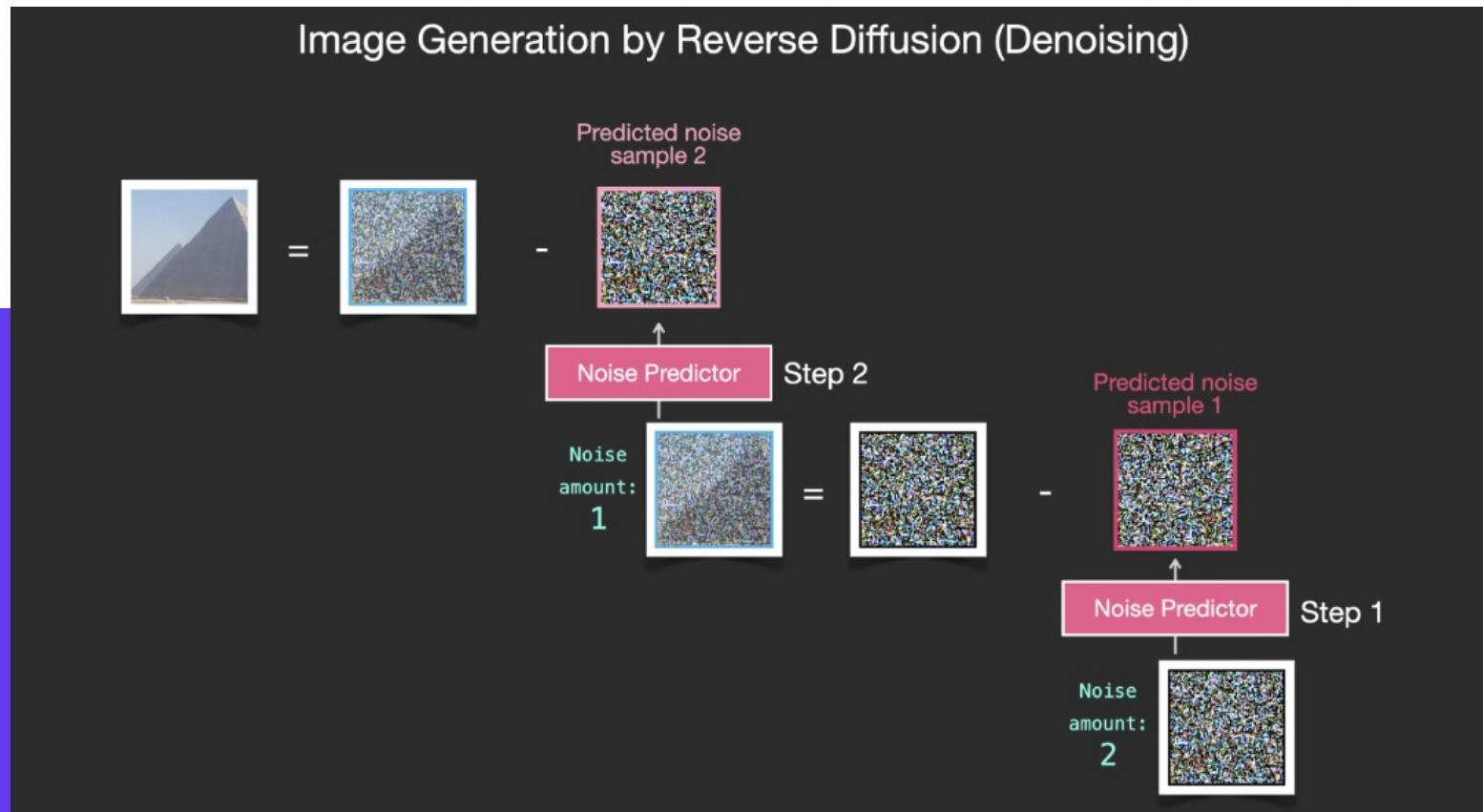
3.4. 移除噪声，绘制图像

由于采样的噪声是可预测的，所以如果从图像中减去噪声，最后得到的图像就会更接近模型训练得到的图像。



3.4. 移除噪声，绘制图像

得到的图像并非是一张精确的原始图像，而是分布（distribution），即世界的像素排列。比如天空通常是蓝色的，人有两只眼睛，猫有尖耳朵等等，生成的具体图像风格完全取决于训练数据集。



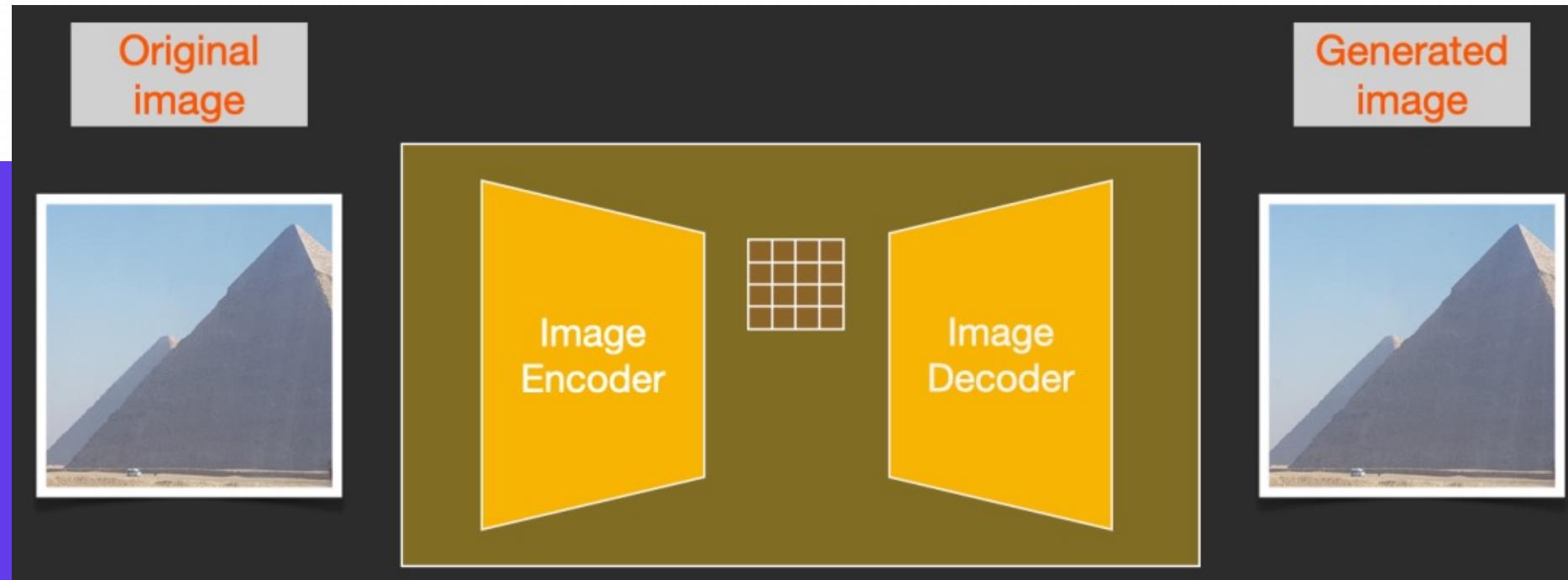
不止Stable Diffusion通过去噪进行图像生成，DALL-E 2和谷歌的Imagen模型都是如此。

需要注意的是，到目前为止描述的扩散过程还没有使用任何文本数据生成图像。因此，如果我们部署这个模型的话，它能够生成很好看的图像，但用户没有办法控制生成的内容。

3.5. 加速：在压缩数据上扩散

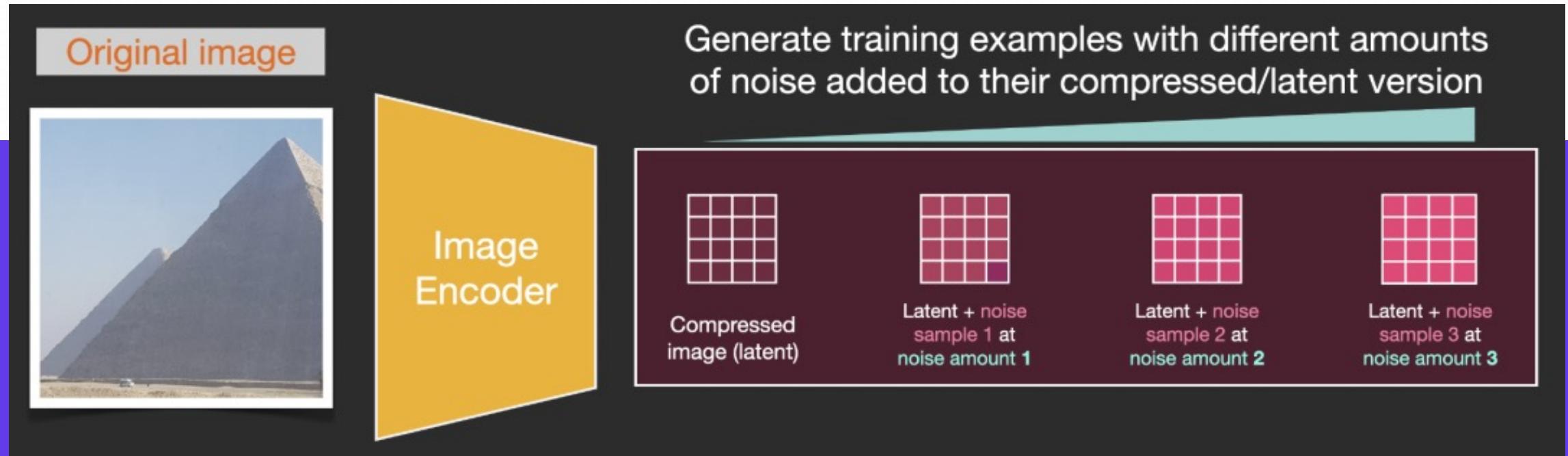
为了加速图像生成的过程，Stable Diffusion并没有选择在像素图像本身上运行扩散过程，而是选择在图像的压缩版本上运行，论文中也称之为「Departure to Latent Space」。

整个压缩过程，包括后续的解压、绘制图像都是通过自编码器完成的，将图像压缩到潜空间中，然后仅使用解码器使用压缩后的信息来重构。



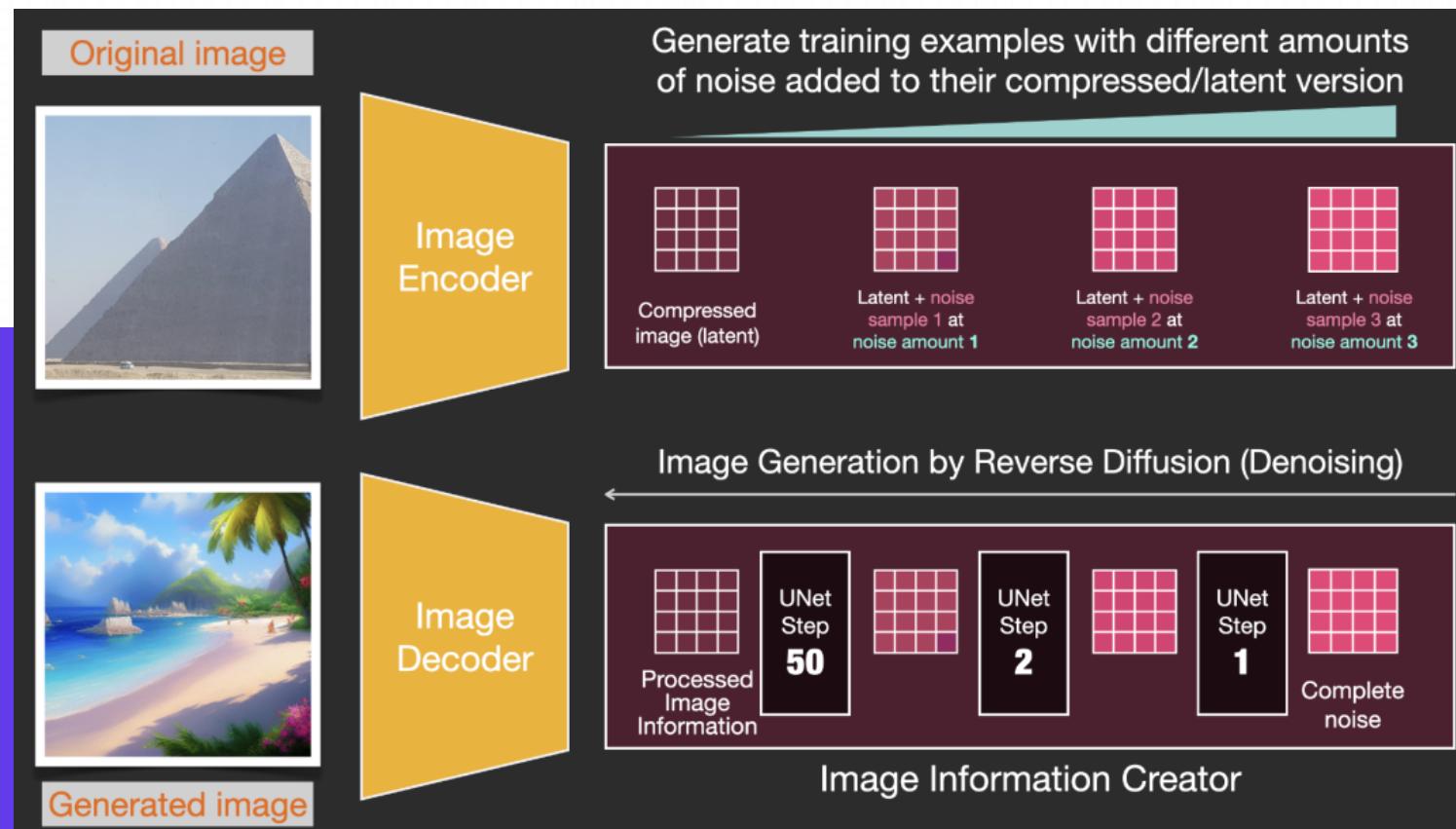
3.5. 加速：在压缩数据上扩散

前向扩散 (forward diffusion) 过程是在压缩latents完成的，噪声的切片 (slices) 是应用于latents上的噪声，而非像素图像，所以噪声预测器实际上是被训练用来预测压缩表示（潜空间）中的噪声。



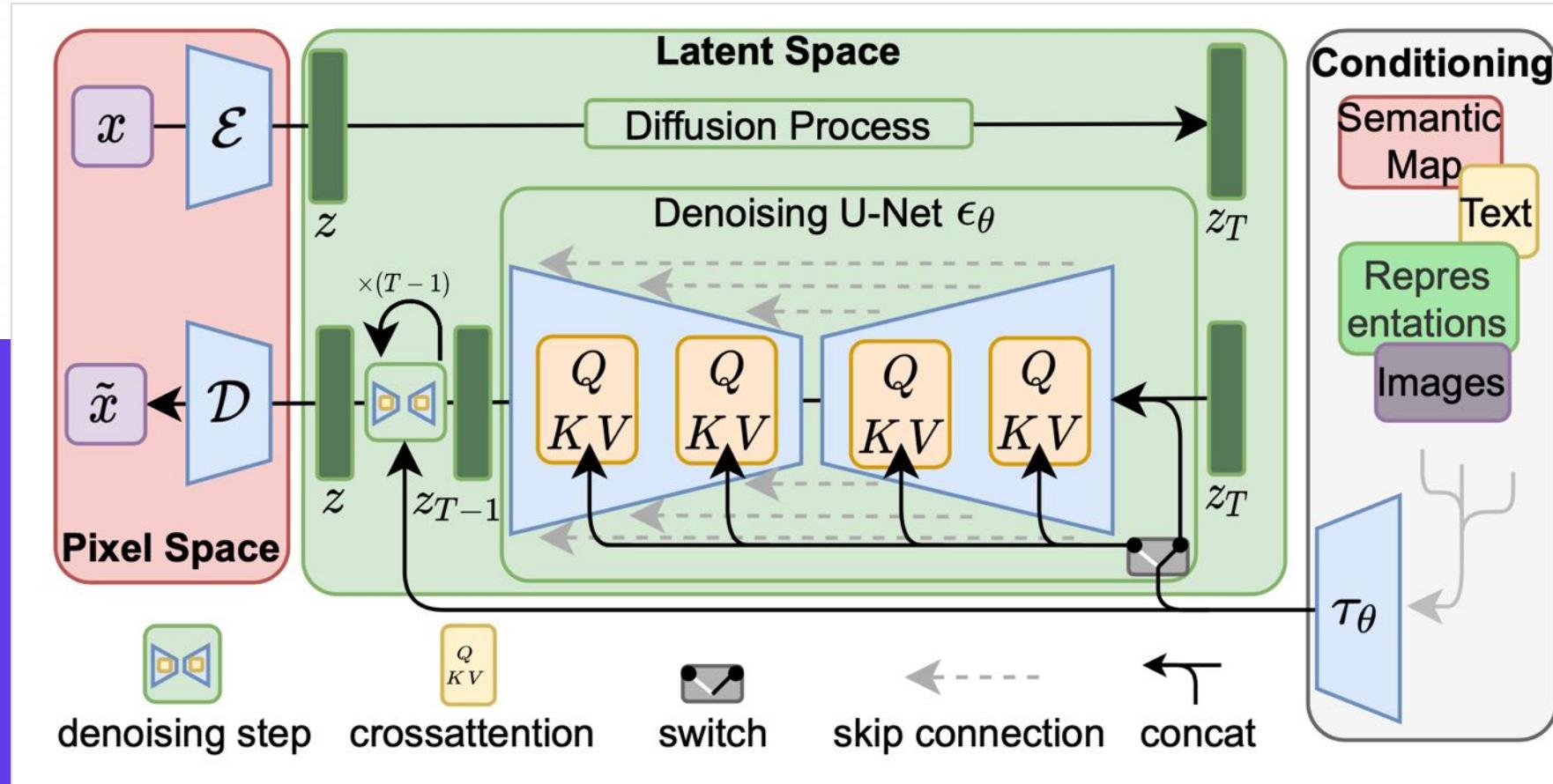
3.5. 加速：在压缩数据上扩散

前向过程，即使用使用自编码器中的编码器来训练噪声预测器。一旦训练完成后，就可以通过运行反向过程（自编码器中的解码器）来生成图像。



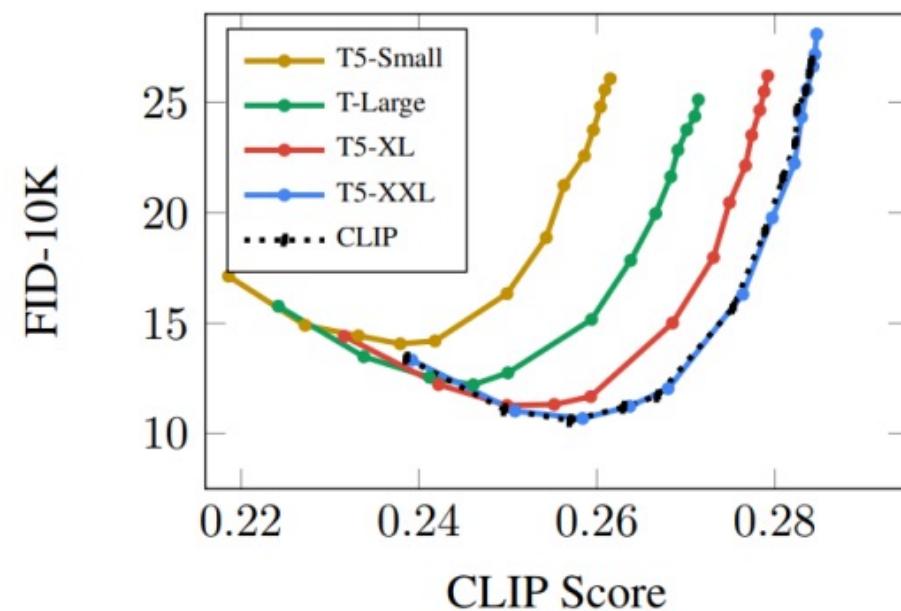
3.5. 加速：在压缩数据上扩散

前向和后向过程如下所示，图中还包括了一个 **conditioning** 组件，用来描述模型应该生成图像的文本提示。

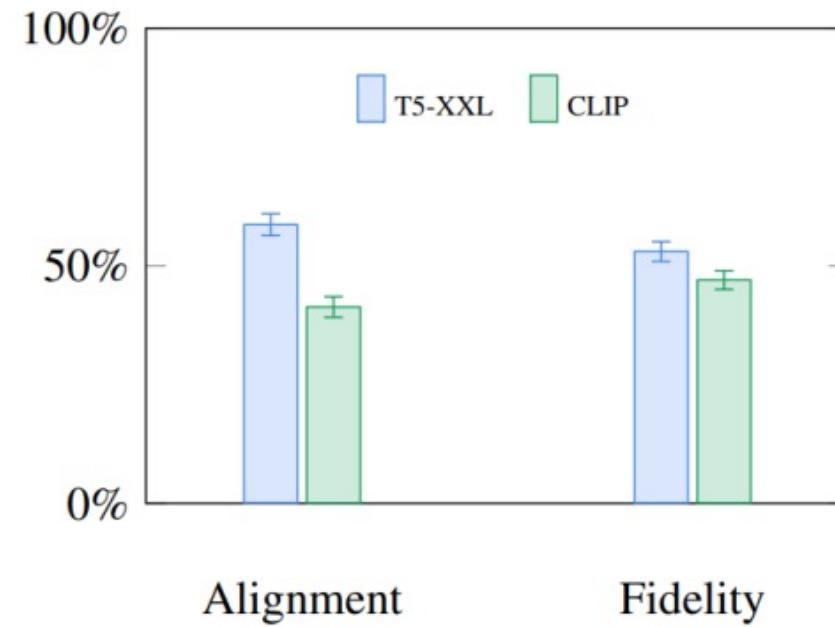


3.6. 文本编码器：一个 Transformer 语言模型

模型中的语言理解组件使用的是Transformer语言模型，可以将输入的文本提示转换为 token 嵌入向量。发布的 Stable Diffusion 模型使用 ClipText (基于 GPT 的模型)，本课程中为了方便讲解选择使用 BERT模型。



(a) Pareto curves comparing various text encoders.



(b) Comparing T5-XXL and CLIP on DrawBench.

Imagen论文中的实验表明，相比选择更大的图像生成组件，更大的语言模型可以带来更多的图像质量提升。

3.6. 文本编码器：一个 Transformer 语言模型

早期的 Stable Diffusion 模型使用的是OpenAI发布的经过预训练的 ClipText 模型，而在Stable Diffusion V2 中已经转向了最新发布的、更大的 CLIP 模型变体 OpenClip.

CLIP是怎么训练的？

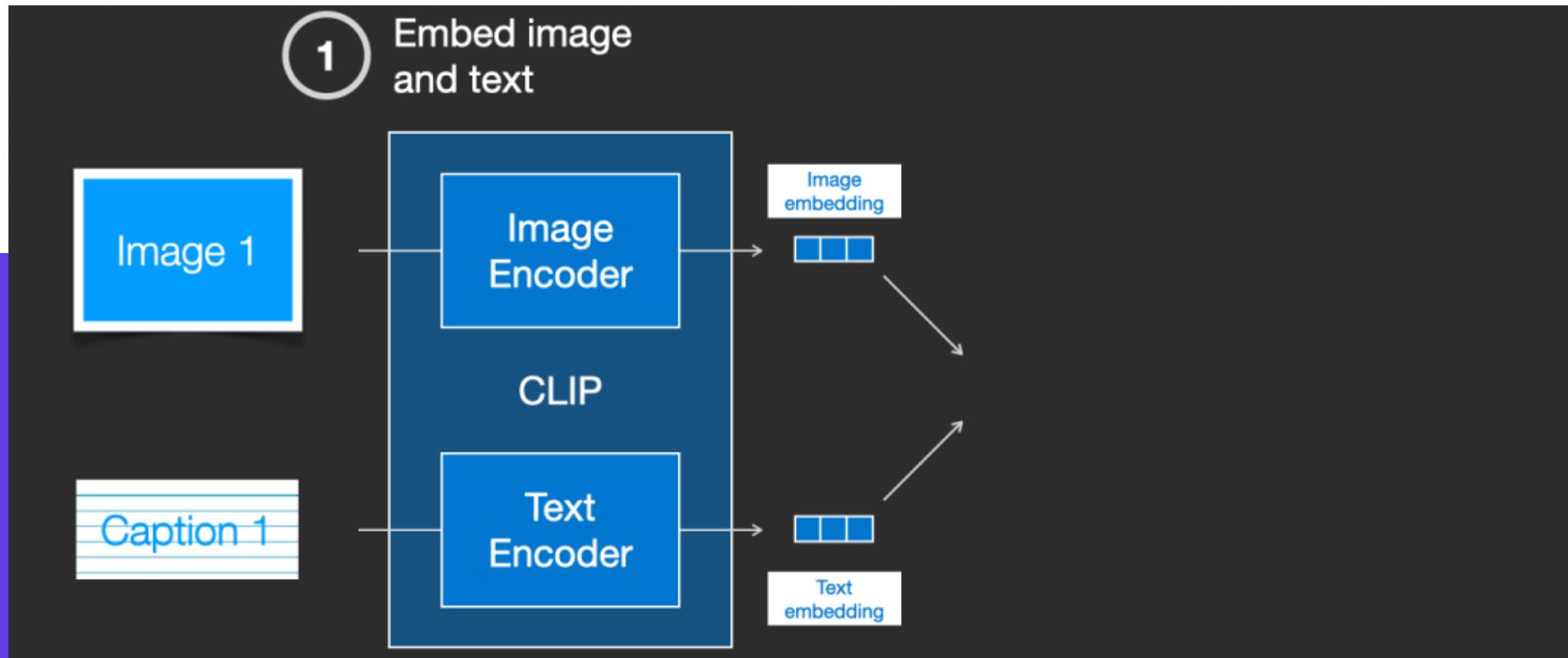
CLIP需要的数据为图像及其标题，数据集中大约包含4亿张图像及描述。



3.6. 文本编码器：一个 Transformer 语言模型

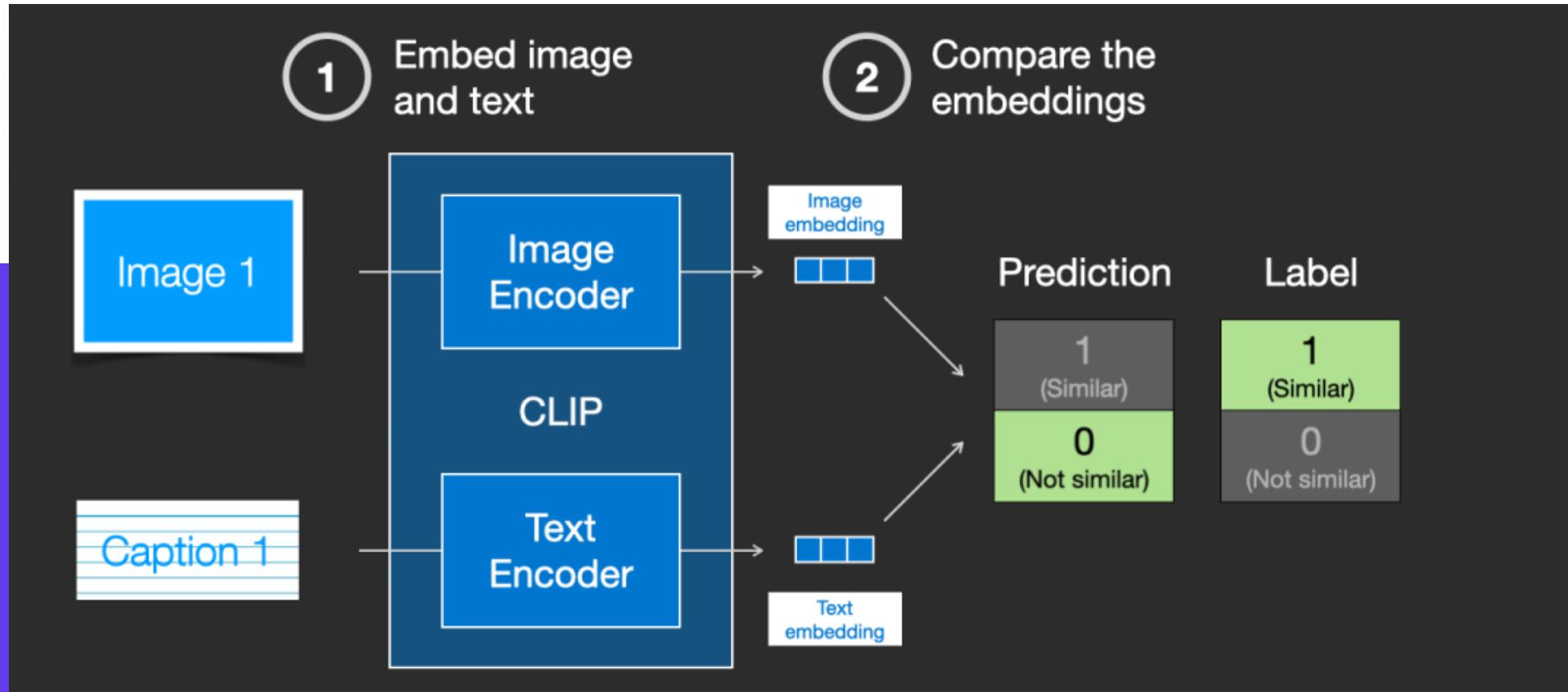
数据集通过从网上抓取的图片以及相应的「alt」标签文本来收集的。

CLIP 是图像编码器和文本编码器的组合，其训练过程可以简化为拍摄图像和文字说明，使用两个编码器对数据分别进行编码。



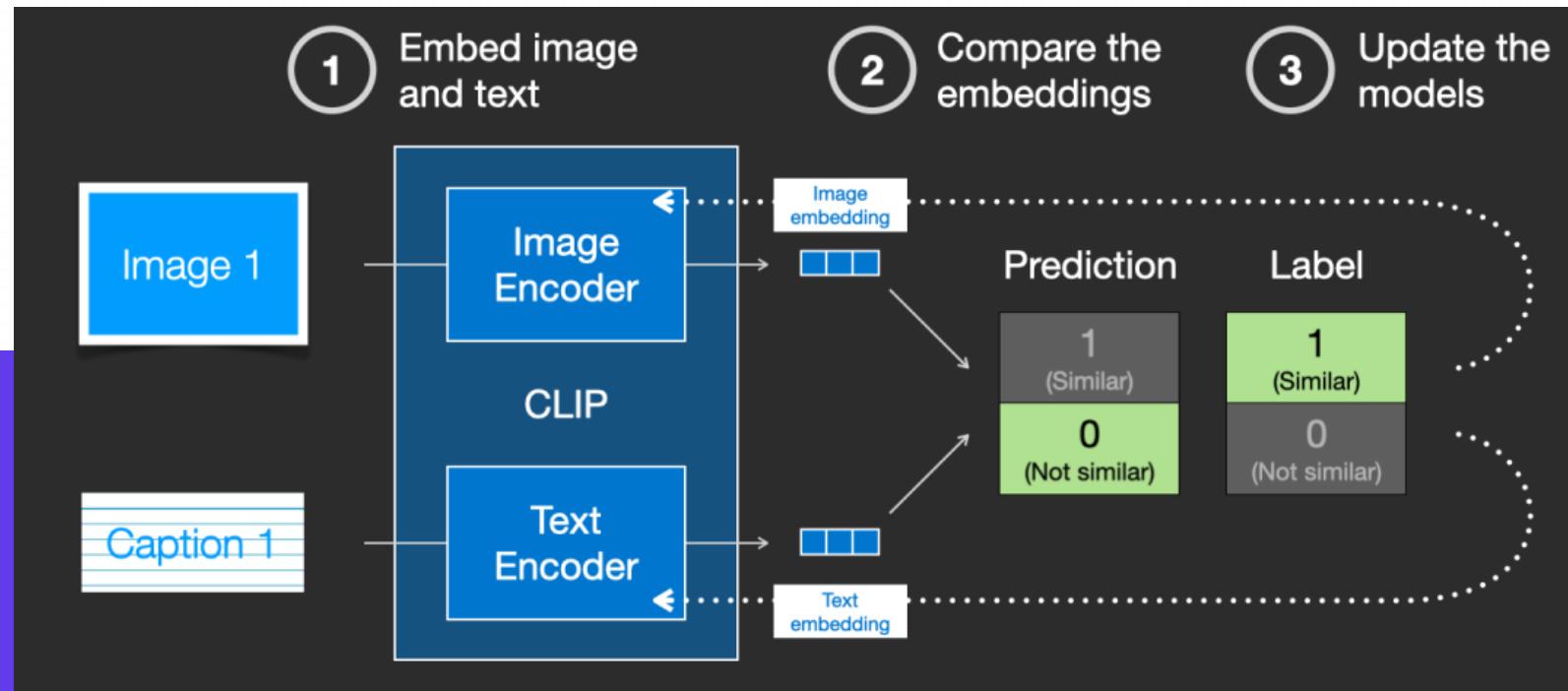
3.6. 文本编码器：一个 Transformer 语言模型

然后使用余弦距离比较结果嵌入，刚开始训练时，即使文本描述与图像是相匹配的，它们之间的相似性肯定也是很低的。



3.6. 文本编码器：一个 Transformer 语言模型

随着模型的不断更新，在后续阶段，编码器对图像和文本编码得到的嵌入会逐渐相似。

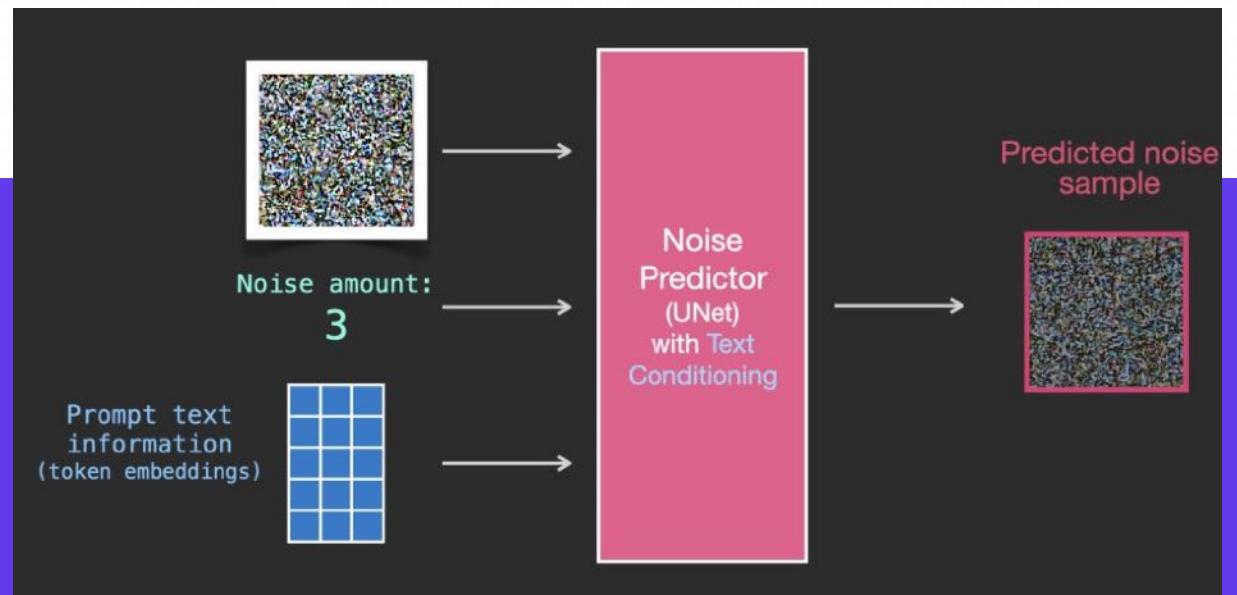


通过在整个数据集中重复该过程，并使用大batch size的编码器，最终能够生成一个嵌入向量，其中狗的图像和句子「一条狗的图片」之间是相似的。

就像在 word2vec 中一样，训练过程也需要包括不匹配的图片和说明的负样本，模型需要给它们分配较低的相似度分数。

3.7. 文本信息喂入图像生成过程

为了将文本条件融入成为图像生成过程的一部分，必须调整噪声预测器的输入为文本。



所有的操作都是在潜空间上，包括编码后的文本、输入图像和预测噪声。

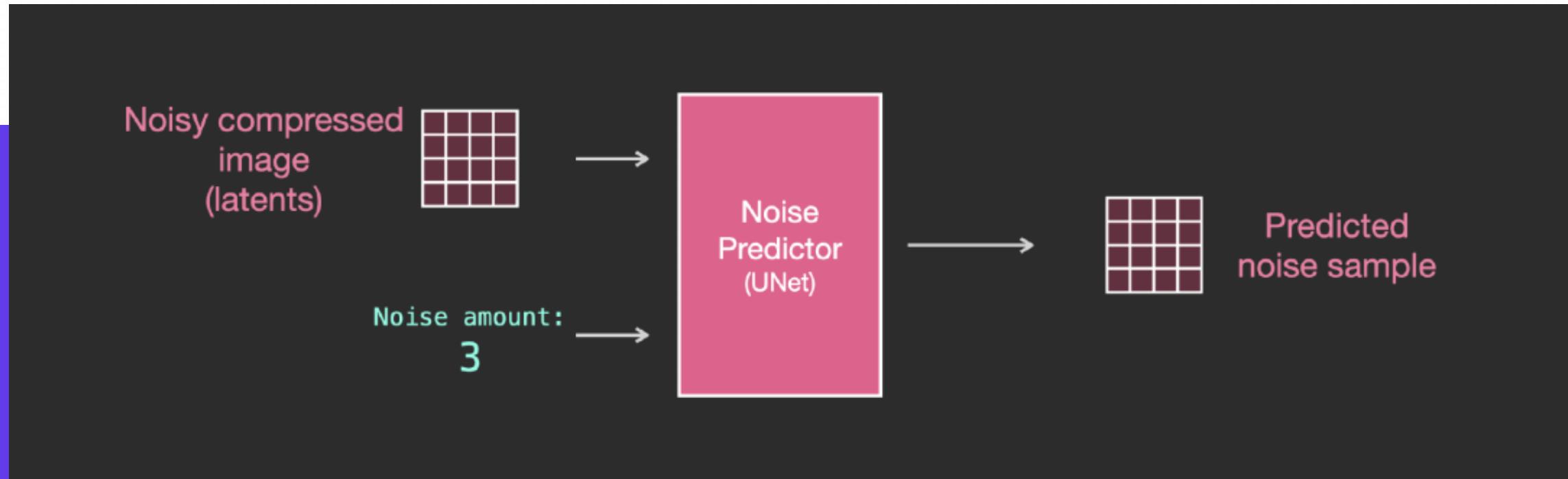


3.7. 文本信息喂入图像生成过程

为了更好地了解文本token在 Unet 中的使用方式，还需要先了解一下 Unet模型。

Unet 噪声预测器中的层（无文本）

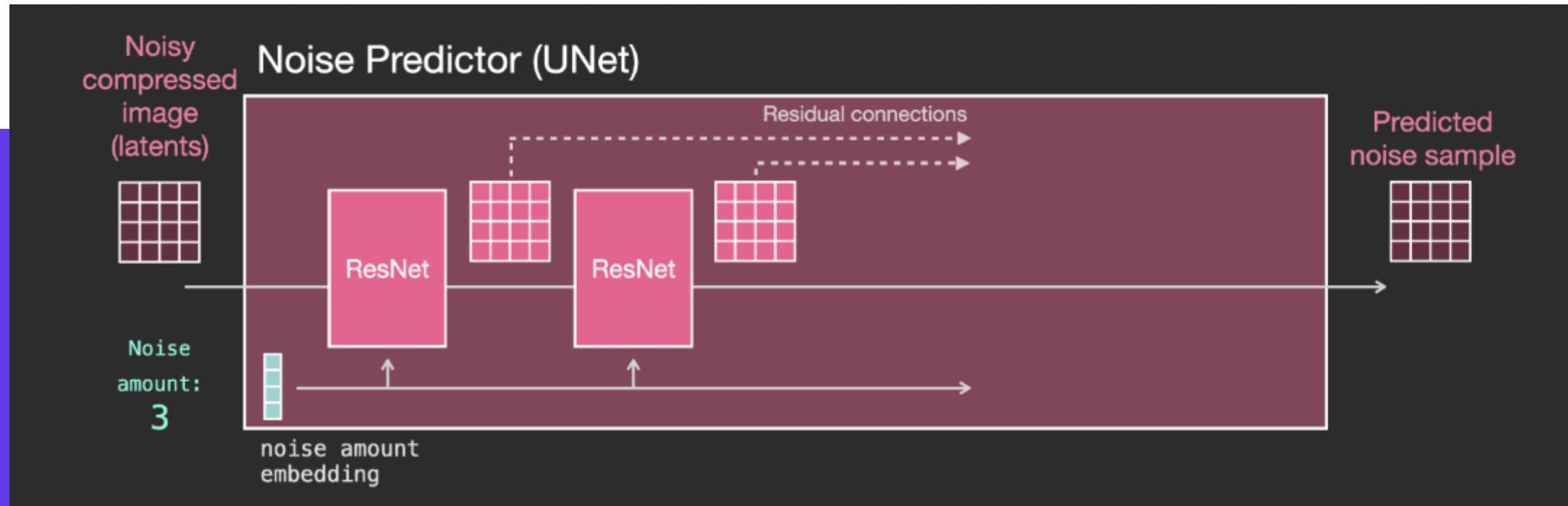
一个不使用文本的diffusion Unet，其输入输出如下所示：



3.7. 文本信息喂入图像生成过程

在模型内部，可以看到：

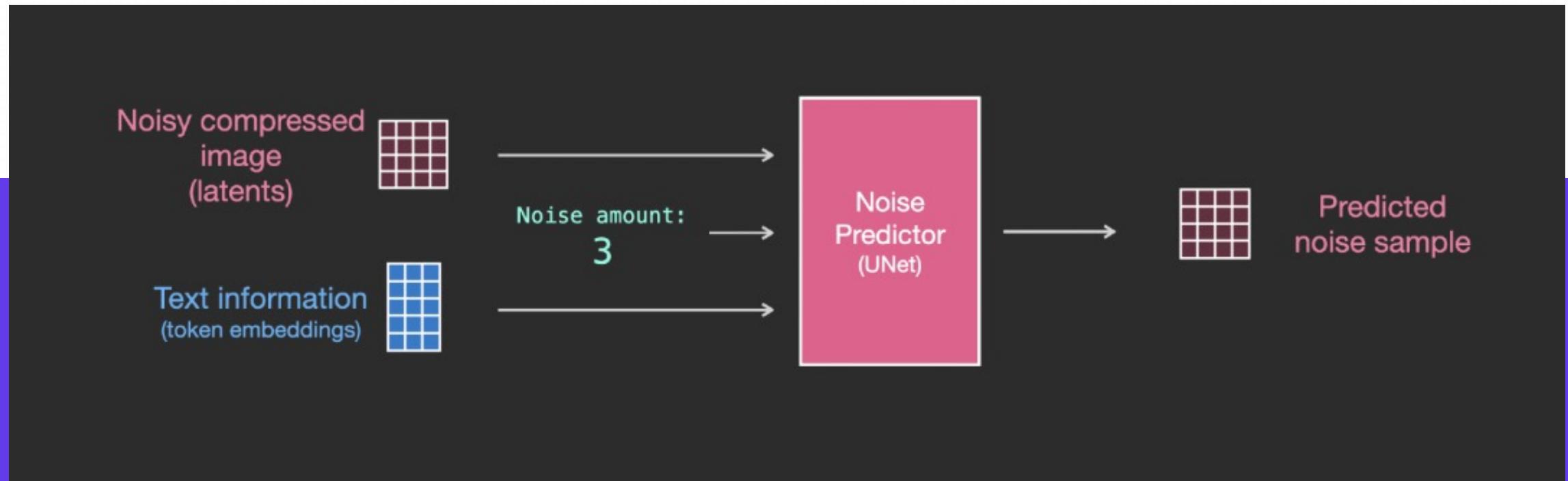
1. Unet模型中的层主要用于转换latents；
2. 每层都是在之前层的输出上进行操作；
3. 某些输出（通过残差连接）将其馈送到网络后面的处理中
4. 将时间步转换为时间步长嵌入向量，可以在层中使用。



3.7. 文本信息喂入图像生成过程

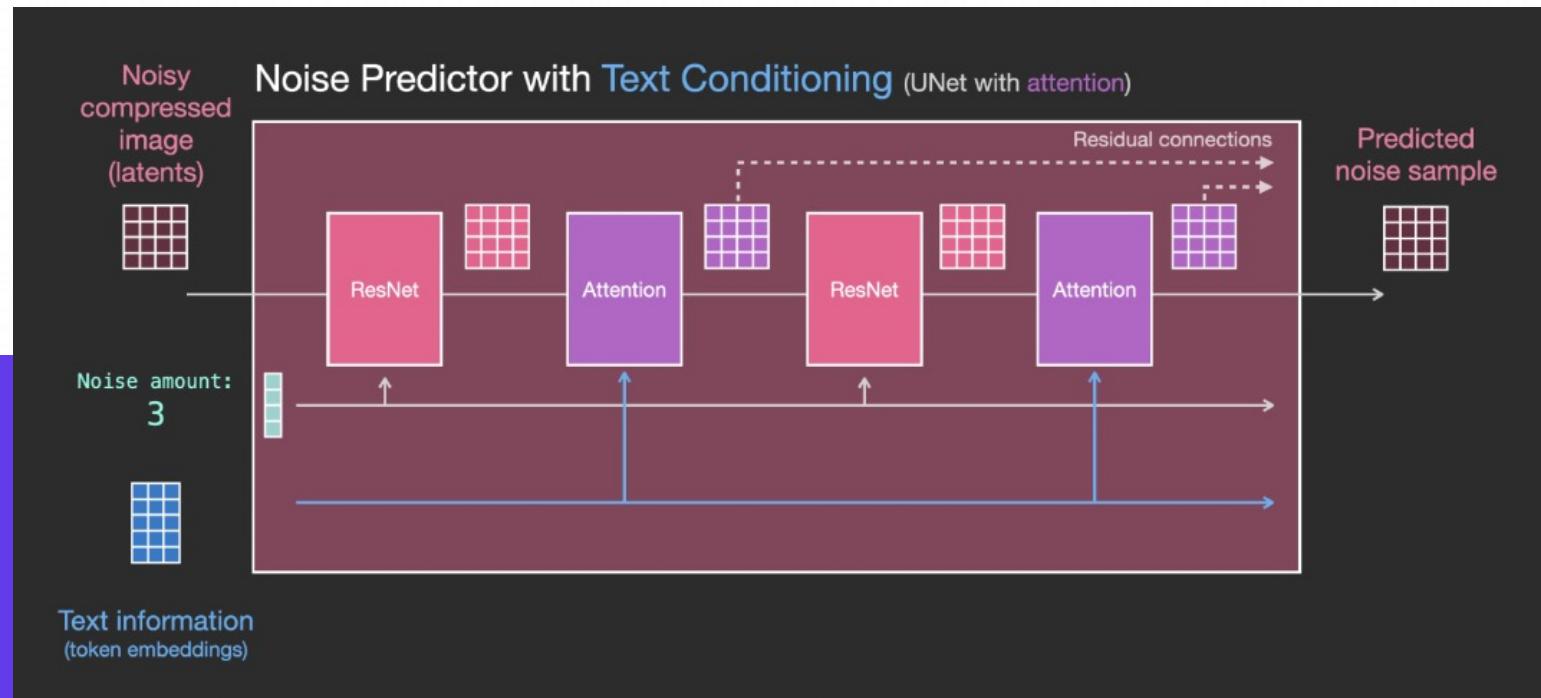
Unet 噪声预测器中的层（带文本）

现在就需要将之前的系统改装成带文本版本的。



3.7. 文本信息喂入图像生成过程

主要的修改部分就是增加对文本输入（术语：text conditioning）的支持，即在ResNet块之间添加一个注意力层。



需要注意的是，ResNet块没有直接看到文本内容，而是通过注意力层将文本在latents中的表征合并起来，然后下一个ResNet就可以在这一步骤中利用上文本信息。



智能大数据可视化实验室
INTELLIGENT BIG DATA VISUALIZATION LAB