第二次训练：

Loss\_vlb：扩散模型通常是一种非线性的模型，其参数通常是通过最大似然估计或贝叶斯推断来学习。在贝叶斯推断中，我们需要计算后验分布，这通常是非常困难的。变分推断是一种有效的方法，可以用来近似这个后验分布。在变分推断中，我们的目标是找到一个近似分布，使其尽可能地接近真实分布。但是，由于我们无法直接优化真实分布，因此我们需要将其转化为优化一个下界。这个下界被称为变分下界。通过最小化变分下界损失函数，我们能够逐步逼近目标分布，并得到更好的近似结果。

变分下界损失函数反映了近似分布和真实分布之间的差异。

变分下界损失函数值越小越好。

Loss\_vlb出现抖动：每个迭代步骤只利用了一个小批量数据，这可能导致损失函数的不稳定性和抖动。解决此类问题的方法通常是增加训练数据。这是因为在计算变分下界损失函数时，我们通常需要对整个数据集进行求和，从而计算出整个数据集的损失函数。如果样本数据较少，那么每次迭代计算得到的损失函数也会受到该少量数据中的噪声等因素的影响，从而导致损失函数值的波动和抖动。

任务完成情况：

接下来就是我们的任务完成情况。我们的课题任务分别是：

理解动漫上色任务和该论文动机；

在相关工作中找到6篇文章以上，理解文章内容，总结并提出自己关于上色任务的见解；

理解ControlNet论文方法，并复现。

我们目前的完成进度是100%，以下就是我们完成的任务：

学习前置知识、理解 ControlNet 论文动机和方法；

学习 8 篇相关论文、理解文章内容、总结小组关于上色任务的见解；

分析 ControlNet 源代码、搭建模型并实现动漫线稿上色；

构建数据集、训练模型并进行评估分析。

接下来对我们的汇报情况进行简要的回顾。首先我们初步了解ControlNet，知道他是一个可以给扩散模型添加控制条件的神经网络结构，可以选择通过不同数据集训练出来的模型来控制图像生成。主要步骤就是首先将图像进行预处理，stable diffusion会根据输入prompt进行生成，ControlNet会根据预处理得到的特征对生成过程进行控制。然后是Canny边缘检测相关知识，这在之后几周的学习中发现主要是用在预处理步骤，也就是提取线稿图部分，实际上在ControlNet真正起作用的部分并没有用上。还学习了其他的比如图像评价指标、扩散模型的基本概念等。之后我们学习ControlNet论文，理解ControlNet的动机和方法。首先学习了Stable Diffusion的论文，理解Stable Diffusion模型。SD通过预训练好的自编码器将图像压缩到潜在空间，解决扩散模型扩散过程慢的问题，还引入条件机制实现多模态条件，首先通过特殊的编码器对条件进行预处理（图中ty），通过cross-attention映射控制信息到UNet。Stable Diffusion是一个大模型，需要庞大的数据集和很多的资源才能训练出色，ControlNet的目的就是在有限资源的情况下对大模型进行微调，也能达到较好的结果。ControlNet将stable diffusion复制出一份trainable copy。locked copy这部分是在训练过程中参数是不会调整的，而trainable copy这部分是训练过程中去调整它的参数。于是locked copy这部分保留网络原来的能力，而trainable copy这部分学习怎么控制生成。Zero卷积初始权重和偏差都是0。相当于一个映射，trainable copy每层增加一个0卷积与原始网络的对应层相连。一开始，模型的输入输出和不加ControlNet的结果是一样的，进一步的优化就会产生效果使模型性能提升。

接下来是搜集并学习相关论文并进行理解，最后总结小组的见解。首先是Attention机制，主要目的就是将有限的注意力集中在重点信息上，从而节省资源，快速获得最有效的信息。（Q是输入的信息，即当前任务的目标，用于和key进行匹配。K和V，一般是相同的数据，比如原始文本经过Embedding后的表征）。然后学习了self attention和cross attention，cross attention就是将图像作为Q，文本信息作为K和V，从而实现多模态。（V必须是文本。以文本为条件控制图像的生成，所以通过注意力的形式来计算每个文本序列与图像不同区域之间的关联程度，即对于每个文本单词计算它与图像所有位置的相似度得分，然后归一化，并用它们加权求和得到最终加入条件控制的图像表示）。之后又学习了ControlNet相关工作中的Text Encoder部分，CLIP模型进行文图匹配。在训练的过程中得到n个图片的特征和n个文本的特征，将这些特征进行对比学习。之后又具体学习了Diffusion Model，也就是可以当作前一张图的Generation Model的部分。（为什么用高斯噪声（不用椒盐噪声）？斯噪声是和原来图像像素值相关的噪声，而椒盐噪声则是很多的黑白像素点；高斯噪声直接加到原图像上，因此可以用线性滤波器滤除，椒盐噪声用线性滤波器滤除的结果不好，一般采用中值滤波器滤波）下面就是推理和扩散过程图示。之后搜集了fine-tuning相关的论文，并进行理解，包括Textual Inversion，DreamBooth，LoRA等。Textual Inversion所训练出来的结果实际上就是一个单词与其对应embedding的映射记录，DreamBooth相当于建立各个类之间的关联，而它会fine-tune模型所有层，所以会导致训练出一个完整的新模型从而体量很大，LoRA则是冻结预训练模型的权重，然后在旁边加上一些小的网络，最后把预训练模型的输出和新加网络部分的输出加起来得到最后经过调整后的输出。之后我们总结了关于上色任务的见解，Stable Diffusion负责从噪声中生成图片，文字text作为prompt唯一输入进行引导，而ControlNet作用就是理解图像信息获取图像的特征，作为text之外的另一个prompt维度，用来控制图像的布局，从而能够得到我们想要的图片。

再之后我们分析了源代码，在服务器上搭建模型并实现了线稿上色。首先在应用层面上简要对比了下DragGAN与Diffusion Model，之后整体上分析了ControlNet代码的逻辑结构， 主要是以下几个步骤，然后调整部分参数和输入prompt，进行图像上色的效果对比，和动漫线稿上色展示。

最后构建数据集、尝试自己训练模型并进行简要评估分析……

收获与体会：

通过这个课题，我们有一些收获和体会……