深度学习自然语言处理第四次作业

李明昕 SY2206124

1. 模型

本次作业使用 **LSTM** 实现一个 **生成式语言模型**。

1.1 生成式语言模型

可以用单向语言模型来建模生成式语言模型。

也就是说，对于一段文本序列 $\{w_1, w_2, \cdots, w_n\}$，其概率可以表示为：

$$P(w_1, \cdots, w_n) = \prod_{i=1}^{n} P(w_i | w_{<i})$$

对于文本生成，根据已有的文本序列 $\{w_1, \cdots, w_k\}$，可以利用语言模型给出的 $P(w_{k+1}|w_{<k+1})$ 通过解码算法完成文本生成的过程。

常见的解码算法包括：

1. 贪心算法：在解码的过程中每次只选择概率最高的字（或词）进行生成；
2. beam-search：维护一个大小为 m 的窗口，生成的过程中保留概率前 m 大的生成结果；
3. 基于采样的算法：在生成的过程中不直接选择概率最高的字（或词）进行生成，而是通过概率进行采样生成。

在解码的过程中可以通过一下三个参数调整概率分布：

1. topK: 只保留分布中概率前 K 大的字（或词），并对概率重新进行归一化；
2. topP: 只保留分布中概率加和刚好超过 P 的前几个字（或词），并对概率进行重新归一化；
3. τ: 由于分布通常是通过 softmax 计算得到的，所以可以通过温度参数 τ 改变 softmax 的结果。τ 越大分布越均匀，τ 越小分布越陡峭。

1.2 LSTM

LSTM（长短期记忆网络）是一种循环神经网络（RNN）的变体，特别适用于处理具有时间依赖性的序列数据。LSTM通过引入门控机制，能够有效地解决传统RNN中的长期依赖问题。LSTM的核心思想是维护一个内部记忆单元（cell state），并通过三个门控单元（输入门、遗忘门和输出门）来控制内部记忆单元的读写和遗忘操作。这些门控单元使用sigmoid函数和逐元素乘法来决定信息的流动。

具体来说，对于时间步t：假设输入为$x(t)$，前一隐藏状态为$h(t-1)$，前一记忆单元状态为$c(t-1)$，则：

1. 输入门（input gate）：

$$i(t) = \sigma(W(i) \cdot [h(t-1), x(t)] + b(i))$$

2. 遗忘门（forget gate）：

$$f(t) = \sigma(W(f) \cdot [h(t-1), x(t)] + b(f))$$

3. 输出门（output gate）：

$$o(t) = \sigma(W(o) \cdot [h(t-1), x(t)] + b(o))$$

4. 新的记忆单元状态（cell state）：

$$c'(t) = \tanh(W(c) \cdot [h(t-1), x(t)] + b(c))$$

5. 当前记忆单元状态（cell state）：

$$c(t) = f(t) \cdot c(t-1) + i(t) \cdot c'(t)$$

6. 当前隐藏状态（output）：

$$h(t) = o(t) \cdot \tanh(c(t))$$

在上述公式中，W和b分别表示权重和偏置项。$[h(t-1), x(t)]$表示将前一隐藏状态$h(t-1)$和当前输入$x(t)$进行拼接。σ表示sigmoid函数，$\tanh$表示双曲正切函数。

2. 语料库

语料库一共包含了**金庸**以及**古龙**两位武侠小说作家的共 94 部小说，其中金庸的小说共 16 部，古龙的小说共 78 部。

2.1 数据预处理

首先去除小说中的广告以及空白字符，然后将两位作者的小说分别整合到一个文件中。

```
import re
import os
```

```
DATA_DIR = 'resources'
def clean_and_collect(author: str):
    ad_p = re.compile(r"本书来自www.cr173.com免费txt小说下载站\n更多更新免费电子书请关注www.cr173.com")
    b_p = re.compile(r"\s")
    nc_p = re.compile(r"[^\u4e00-\u9fa5，…：、。！？；]")

    books = os.listdir(os.path.join(DATA_DIR, author))
    corpus = ''

    for book in books:
        with open(os.path.join(DATA_DIR, author, book), 'r', encoding='gb2312', errors='ignore') as fi:
            corpus += nc_p.sub('', b_p.sub('', ad_p.sub('', fi.read())))

    with open(os.path.join(DATA_DIR, author, 'corpus'), 'w', encoding='utf8') as fo:
        fo.write(corpus)
```

2.2 分词并构建词表

原先考虑两种分词方法：

1. 以字为单位进行分词；
2. 通过 jieba 分词以词为单位进行分词。

两种分词单位下，分词的结果如下：

分词单位	金庸小说 token 总数	古龙小说 token 总数	词汇表大小
字	8295346	16370203	5772
词	5314006	10783166	292996

```
import jieba
```

```
def tokenize(tokenize_type: str = 'char'):
    tokenized_corpus = {}
    tokens = set()
    for author in AUTHORS:
        with open(os.path.join(DATA_DIR, author, 'corpus'), 'r',
encoding='utf8') as fi:
            if tokenize_type == 'char':
                tokenize_fn = list
            else:
                tokenize_fn = lambda x: list(jieba.cut(x))
            tokenized_corpus[author] = tokenize_fn(fi.read())
            tokens.update(tokenized_corpus[author])
    token2id = {token: i for i, token in enumerate(tokens)}
    id2token = {i: token for token, i in token2id.items()}
    return tokenized_corpus, token2id, id2token
```

但在实际的实验中，由于第二种分词方法导致词表过大，导致训练所需的资源较大，无法进行有效的实验。所以接下来的实验都以字为单位进行分词。

3. 实现

3.1 基于 LSTM 的生成式语言模型

模型主要包括三个部分：

1. 嵌入层：将 token idx 所对应的 one-hot 向量转换为稠密的特征向量；
2. LSTM 层：由多层包含 LSTM 以及前馈神经网络的模块组成，LSTM 与前馈神经网络间均包含残差模块以及 LayerNorm 模块。
3. 输出层：将 token 对应的特征向量转换为此表上的 logits ，用于在 softmax 中计算概率分布。

```
import torch
from torch import nn
from torch.nn import functional as F
```

```
class LSTMBlock(nn.Module):
    def __init__(self, embed_size: int, dropout: float = 0.2):
        super().__init__()
        self.lstm = nn.LSTM(embed_size, embed_size, batch_first=True)
        self.lstm_dropout = nn.Dropout(dropout)
        self.feedforward = nn.Sequential(
            nn.Linear(embed_size, embed_size * 4),
            nn.ReLU(),
            nn.Linear(embed_size * 4, embed_size),
            nn.Dropout(dropout)
        )
```

```
        self.l1 = nn.LayerNorm(embed_size)
        self.l2 = nn.LayerNorm(embed_size)

    def forward(self, x):
        x_res = x
        x, _ = self.lstm(self.l1(x))
        x = x_res + self.lstm_dropout(x)
        x = x + self.feedforward(self.l2(x))

        return x
```

```
class LSTMLanguageModel(nn.Module):
    def __init__(self, vocab_size: int, embed_size: int, layer_num: int,
dropout: float = 0.2):
        super().__init__()
        self.token_embedding = nn.Embedding(vocab_size, embed_size)
        self.blocks = nn.Sequential(*[LSTMBlock(embed_size, dropout) for _ in
range(layer_num)])
        self.lf = nn.LayerNorm(embed_size)
        self.lm_head = nn.Linear(embed_size, vocab_size)

        self.apply(self._init_weights)

    def _init_weights(self, module):
        if isinstance(module, nn.Linear):
            torch.nn.init.normal_(module.weight, mean=0.0, std=0.02)
            if module.bias is not None:
                torch.nn.init.zeros_(module.bias)
        elif isinstance(module, nn.Embedding):
            torch.nn.init.normal_(module.weight, mean=0.0, std=0.02)

    def forward(self, idx, targets=None):
        x = self.token_embedding(idx)
        x = self.blocks(x)
        x = self.lf(x)

        logits = self.lm_head(x)

        if targets is None:
            loss = None
        else:
            B, T, _ = logits.shape
            logits = logits.view(B*T, -1)
            targets = targets.view(B*T)
            loss = F.cross_entropy(logits, targets)

        return logits, loss
```

3.2 模型训练

3.2.1 数据加载

在训练时，同时使用金庸以及古龙的语料进行训练。对于金庸的语料，采用顺序抽取段落的方式以保证所有文本都能得到训练；对于古龙的语料，采用随机抽取段落的方式以增加训练语料的多样性。

此外，金庸小说的语料中只有 95% 被用于训练，剩下的 5% 被留出用于实验部分的测试。

```
import math
import random
```

```
class Corpus:
    def __init__(self, seq_len: int, step_interval: int, batch_size: int,
tokenize_type: str = 'char', eval_p: float = 0.05):
        self.corpus, self.token2id, self.id2token = tokenize(tokenize_type)
        self.corpus['jinyong'], self.eval_corpus = self.corpus['jinyong']
[:int(len(self.corpus['jinyong']) * (1 - eval_p))], \
            self.corpus['jinyong'][int(len(self.corpus['jinyong']) * (1 -
eval_p)): ]

        self.batch_size = batch_size
        self.seq_len = seq_len
        self.step_interval = step_interval

    def get_train_data(self):
        batch_size_jy = int(self.batch_size * 0.7)
        batch_size_gl = self.batch_size - batch_size_jy

        jy_i = 0
        while jy_i < len(self.corpus['jinyong']) - self.seq_len:
            jy_is = list(range(jy_i,
                               min(len(self.corpus['jinyong']) - self.seq_len, jy_i +
batch_size_jy * self.step_interval),
                               self.step_interval))
            jy_i = jy_is[-1] + self.step_interval
            gl_is = [random.randint(0, len(self.corpus['gulong']) -
self.seq_len) for _ in range(batch_size_gl)]

            input_ids = [[self.token2id[token] for token in
self.corpus['jinyong'][left_i: left_i + self.seq_len]]
                         for left_i in jy_is]
            target_ids = [[self.token2id[token] for token in
self.corpus['jinyong'][left_i + 1: left_i + 1 + self.seq_len]]
                          for left_i in jy_is]

            input_ids.extend([[self.token2id[token] for token in
self.corpus['gulong'][left_i: left_i + self.seq_len]]
                              for left_i in gl_is])
            target_ids.extend([[self.token2id[token] for token in
self.corpus['gulong'][left_i + 1: left_i + 1 + self.seq_len]]
                               for left_i in gl_is])

            yield torch.tensor(input_ids, dtype=torch.long),
torch.tensor(target_ids, dtype=torch.long)

    def __len__(self):
```

```
        return math.ceil((len(self.corpus['jinyong']) - self.seq_len) //
self.step_interval / int(self.batch_size * 0.7))

    def get_eval_data(self):
        return torch.tensor([self.token2id[token] for token in
self.eval_corpus], dtype=torch.long)
```

3.2.2 模型训练

模型训练采用**预测下一个词**任务，损失函数使用**交叉熵**。优化器为**Adam**，学习率调整包括 warmup 策略以及余弦退火策略。

```
import time
```

```
class Trainer:
    def __init__(self, model, corpus, epoch, lr, wu_steps, device):

        self.device = device

        self.model = model.to(device=device)
        self.corpus = corpus
        self.step_per_epoch = len(self.corpus)
        self.epoch = epoch
        self.min_lr, self.max_lr = lr

        self.optimizer = torch.optim.Adam(self.model.parameters(),
lr=self.max_lr)
        self.scheduler = torch.optim.lr_scheduler.LambdaLR(
            self.optimizer,
            lr_lambda=lambda cur_iter: cur_iter / wu_steps if cur_iter <
wu_steps else
                (self.min_lr + 0.5*(self.max_lr - self.min_lr) *
                 (1 + math.cos(((cur_iter - wu_steps) / (self.step_per_epoch *
epoch - wu_steps) * math.pi)))) /
                 self.max_lr
        )
        self.loss_log = []

    def train(self, check_interval=100):
        global_step = 0
        min_loss = float('inf')
        start_time = time.time()
        for epoch in range(self.epoch):
            for input_ids, target_ids in self.corpus.get_train_data():
                input_ids = input_ids.to(device=self.device)
                target_ids = target_ids.to(device=self.device)

                _, loss = self.model(input_ids, target_ids)

                loss.backward()

                self.optimizer.step()
                self.optimizer.zero_grad()
                self.scheduler.step()
```

```
                self.loss_log.append(loss.item())
                global_step += 1
                if global_step % check_interval == 0:
                    if self.loss_log[-1] < min_loss:
                        suffix = ", model saved to resources/ckpt/min_loss.ckpt"
                        torch.save(self.model.state_dict(),
'resources/ckpt/min_loss.ckpt')
                    else:
                        suffix = ''
                    step_time = (time.time() - start_time) / global_step
                    print(f'Epoch: {epoch}, GlobalStep: {global_step}, lr:
{self.scheduler.get_last_lr()[0]:.5f}, eta: {step_time * (self.step_per_epoch *
self.epoch - global_step) / 3600:.3f}h, loss: {self.loss_log[-1]:.4f}{suffix}')
            torch.save(self.model.state_dict(), f'resources/ckpt/{epoch}.ckpt')
            print(f'Save Model of Epoch: {epoch} to
resources/ckpy/{epoch}.ckpt')
```

4. 实验

4.1 以字为单位进行分词训练语言模型

```
corpus = Corpus(128, 16, 64, 'char', 0.05)
model = LSTMLanguageModel(len(corpus.token2id), 384, 12, 0.2)
trainer = Trainer(model, corpus, 2, (5e-5, 1e-4), 1000, torch.device('cuda:0'))

# trainer.train(100)
```

一共训练了三种模型，第一个模型相对较小，另外两个模型分别增加了宽度与深度：

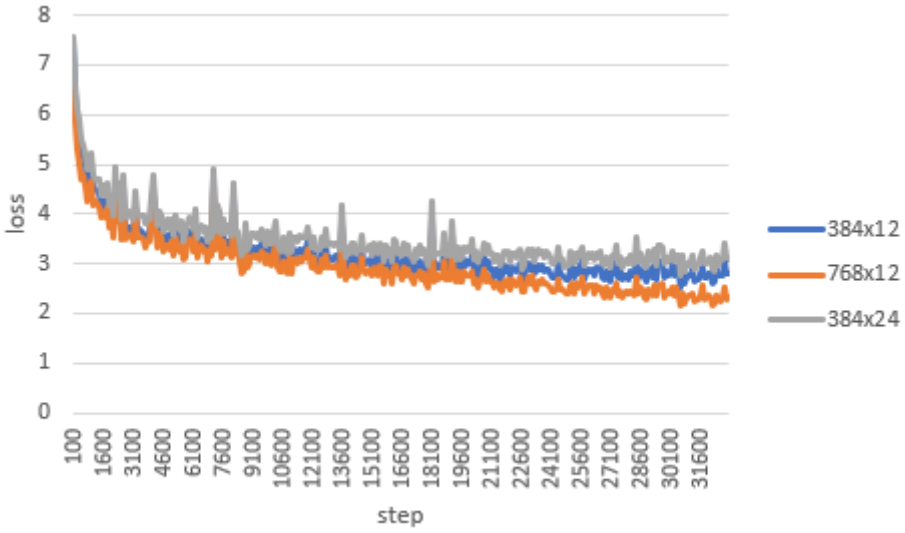
特征维度	层数
384	12
768	12
384	24

由于模型放到了远程进行训练，所以训练过程见日志文件 `resources/384x12.log`，`resources/768x12.log`，`resources/384x24.log`。

训练过程中，每段训练文本的长度为 256 ，批次大小为 256 。学习率的 warmup 步数为 2000，最小学习率为 1e-5 ，最大学习率为 5e-4。

在每个 epoch 结束，以及损失函数值最低时保存模型参数。

损失函数的变化图如下：



4.2 定性分析

在定性分析中，从训练语料以及非训练语料中分别抽取一段长度为 100 的文本（prompt）来提示模型进行生成：

来源	prompt
训练语料	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不
非训练语料	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑

接下来，以人工的方式检查三种解码算法下，模型在已有文本的基础上，继续生成 256 步后，生成文本的质量。

首先对比最小的模型在最后一个 epoch 以及最小 loss 下的结果，然后再对比不同大小模型的结果。

（在以下部分，生成的内容用斜体表示。）

4.2.1 贪心算法

首先是对**训练语料**的生成结果：

模型参数	生成内容	$\log P$

模型参数	生成内容	$\log P$
epoch 5	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不是一般。*张无忌心想：这些人都是好人，我们这些人都是好人，怎地这般不识好歹？我们这些人都是好人，怎能做这等事？只听得一个女子声音说道：你们两个是什么人，胆敢对我们如此无礼？那人道：我们是汉人，不是坏人，是我们的大仇人。张无忌道：是，是！我们是汉人，不是坏人。那女子道：你们这些人，都是我的好朋友，我们是好朋友，你们要是不信，便请动手。张无忌道：我们是好朋友，你们不可伤了和气。张无忌道：我们是好朋友，不是好人。张无忌道：我们是好朋友，不是好人。张无忌道：我们是好朋友，不是好人。张无忌道：我们是朋友，你们是朋友，我*	-638.2491
min loss	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不是一般。*韦小宝心想：这些人都是好人，我们这些人都是好人，怎么会有这等好人？只听得那女子又道：你们这些人，都是我们的大小姐，你们这些人，都是我们的大小姐，你们这些人，我们都是好朋友，你们这些人，我们是一家人，你们两个，一起去罢。韦小宝道：是，是。我们去找些吃的。韦小宝道：好，我们去找些吃的。韦小宝道：好，我们去找些吃的。韦小宝道：好，我们去找些吃的。韦小宝道：好，我去跟你们说，你们两个在这里等我。韦小宝道：好，我去跟你说，你们两个在这里等我。韦小宝道：好，我去跟你说，你们两个在这里等我。韦小宝道：好，我去跟你*	-637.2269

接着是对**非训练语料**的生成结果：

模型参数	生成内容	$\log P$
epoch 5	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，*剑气纵横，剑气*	-833.2676

模型参数	生成内容	$\log P$
min loss	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，*剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气森森，剑气*	-805.5582

对于训练语料中的文本，`epoch 5` 和 `min loss` 生成内容的主要区别是一个主角是**张无忌**，另一个是**韦小宝**，这可能是因为在保存模型参数时，正在训练模型的语料分别是不同主角。对于生成内容的合理性，不同 checkpoint 相差不大，且都是相邻几句话具有一定合理性，但从整段上看缺少逻辑。

对于非训练语料中的文本，可以看出生成的两段都具有大量重复，生成内容的质量较低，并且生成文本的 **P** 也较低。

4.2.2 beam-search 算法

首先统一设置 beam-search 的 k 为 **5**，$\tau = 1$。

首先是对**训练语料**的生成结果：

模型参数	生成内容	$\log P$
epoch 5	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不精。*张无忌听到这里，不由得心花怒放，心想：这位姑娘的名字倒也不错，我若不是她的女儿，怎么会有这般美貌的女子？只听她又道：这位姑娘是我师父，她是我的女儿，她是我的女儿。她是我的女儿，她是我的女儿，她是我的女儿。她是我的女儿，她是我的女儿。她是我的女儿，她是我的女儿。她是我的女儿，她是我的女儿。她是我的女儿，她是我的亲生女儿，她是我的女儿。她是我的女儿，她是我的亲生女儿，她是我的亲生女儿，她是我的亲生女儿，她是我的亲生女儿，她是我的亲生女儿，她是我的亲生女儿，她是我的亲生女儿，她是我的亲生女儿，她是我的亲生女儿*	-596.7995

模型参数	生成内容	$\log P$
min loss	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不知。*韦小宝心中一动，寻思：这小姑娘的话倒也有理。说道：小桂子，你这话可不对了。韦小宝道：是，是，是，是。心想：这小姑娘的话倒也有理，只可惜我不识字，否则……否则*	-499.8869

接着是对**非训练语料**的生成结果：

模型参数	生成内容	$\log P$
epoch 5	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*气纵横天下，无人能及。张无忌心中一凛，暗道：此人剑法之精，果然名不虚传。只见他手中拿着一柄明晃晃的长剑，剑尖指住了自己咽喉，喝道：你这是甚么剑法？那人笑道：我是华山派掌门，你是华山派掌门人，是不是？张无忌道：正是。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：不敢，不敢。张无忌道：我*	-764.5544
min loss	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*气纵横天下，不知有多少英雄好汉。韦小宝心道：原来如此，原来如此。说道：在下有一件要紧事，要请教阁下高姓大名。韦小宝道：是，是。那老者道：是，是。韦小宝道：是，是。那老者道：是，是。韦小宝道：是，是。心想：这位姑娘年纪轻轻，武功也不在我之下，怎地如此大胆，竟敢对我如此无礼？说道：小兄弟，你不用担心，我跟你一起去。韦小宝道：是，是。心中却想：这小姑娘的话倒也有理。说道：你不用担心，我去跟你说。韦小宝道：好，我跟你说。韦小宝道：好，我跟你说，我跟你说，你跟我说，我跟你在一起，是不是？韦小宝道：是，是。我……我……*	-785.3036

从 **P** 来看，Beam search 的结果的 **P** 都比使用贪心算法的小，但内容的质量却没有明显提高，仍有大量重复锻炼，只有少数段落的合理长度变长了。

对于 beam search 算法，还可以实验不同 k 下的生成结果（统一使用min loss的checkpoint对非训练语料进行生成），其中 k=1 的时候就相当于贪心算法：

k	生成内容	$\log P$
1	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，*剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气森森，剑气*	-805.5582
5	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，*剑气纵横天下，不知有多少英雄好汉。韦小宝心道：原来如此，原来如此。说道：在下有一件要紧事，要请教阁下高姓大名。韦小宝道：是，是。那老者道：是，是。韦小宝道：是，是。那老者道：是，是。韦小宝道：是，是。心想：这位姑娘年纪轻轻，武功也不在我之下，怎地如此大胆，竟敢对我如此无礼？说道：小兄弟，你不用担心，我跟你一起去。韦小宝道：是，是。心中却想：这小姑娘的话倒也有理。说道：你不用担心，我去跟你说。韦小宝道：好，我跟你说。韦小宝道：好，我跟你说，我跟你说，你跟我说，我跟你在一起，是不是？韦小宝道：是，是。我……我……*	-785.3036
10	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，*剑气森森，剑气森森，剑气森森，剑气森森。韦小宝心道：原来如此，原来如此。说道：小兄弟，你剑法精妙，佩服，佩服，佩服，佩服，佩服，佩服，佩服，佩服！韦小宝心想：这小姑娘的武功果然了得。当下不动声色，说道：小兄弟，你的剑法比我高明得多了。韦小宝摇头道：不是，不是，不是的。韦小宝道：是，是。心想：这小姑娘的武功比我高得多，我可不是他的对手。说道：不敢，不敢，不敢。韦小宝道：不敢，不敢，不敢。韦小宝道：不敢，不敢，不敢。韦小宝道：是，是。心想：这小子武功虽高，却也不是什么了不起的人物。说道：不敢当，不敢当。韦小宝道：*	-777.6389
15	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，*剑气森森，剑气森森，剑气森森。韦小宝心中一凛，寻思：这人的武功如此了得，倒也不见得是什么了不起的人物。说道：小兄弟，你的剑法比我高得多了。韦小宝摇头道：不是，不是，不是。韦小宝道：原来如此。韦小宝道：原来如此。只不过……只不*	-700.7898

从结果中可以看出，增加 K 可以增加 **P**，但受到模型性能的限制，还是不能有效改善生成的内容。

4.2.3 基于采样的算法

首先实验 $\tau = 1$ 时的效果。

首先是对**训练语料**的生成结果：

模型参数	生成内容	$\log P$
epoch 5	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，*无一不和得自己全然相同。水轮约有十来斤重，宛如钉在地下一般，较之古墓前更无更矮。过得片刻，在断桅之外擦了一头白色雪，行颠倚柱而坐，手指缝中竟然透出淡淡白气。群豪却也大奇，压低了声音，兀自不声不响，似乎便是怕自己武功高强，那挑一根竹签便是一根晶钢丝的小虾袋了。这室外群豪听得风声，三人蹑手蹑足的进屋，纷纷抽出兵刃。店中黑压压的站着一人，贾布从五十来岁上站出一个身穿灰色装束的少女，双手短袖似雨，却哪里有人其中之一？杨逍和秦绢见他跃上甲板，正是左冷禅床头所绣，他身上衣衫漆黑，这时站直了身子，便也失了兵器。这些锦衣大汉合力*	-1044.6216
min loss	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，*无一不雅，但阿珂满脸均是迷惘。两名婢女齐声欢呼：妙极，妙极！双儿向韦小宝躬身说道：启禀公主殿下。黄眉僧道：方丈大师大驾光降，公主太让大人瞧瞧，是否有点小事行事，请现身相见。韦小宝听说脚步声是两个女子，又不动声色，问道：阁下是谁？那女子微微一笑，道：这祠堂还在妓院么？陈近南一听，登时恍然，心道：不是鬼使妖怪。挖了我面幕陪我去啊。正没作理会处，蓬的一声，火头从瓦边迅速飞了过来，直飞上天，半空中劲风呼呼作响，掌影连起，竟被那飞钟踹中。这次陆高轩、关西卫士、公主、沐剑屏、高升泰等都奔进屋来时，见一只手拉着他正自跟踪众女，*	-930.0667

接着是对**非训练语料**的生成结果：

模型参数	生成内容	$\log P$

模型参数	生成内容	$\log P$
epoch 5	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑*冲日自斗，剑将树沽，朕为马者尸，乃山水之流。草草溶五；峰上石树为膝，门外皆上，沉重若渊。全冠清翻阅读书，已认得此批白须老者是韦人，荤餐得精神大振，捧着墓碑，坐在地下陪笑道：尊驾请坐在这首席，请坐平日算帐。大家再饮一杯如何？空智大师笑道：好啊，这套佛经易筋经已练成了异门心法，是否能结成一条人偶。嘿嘿，灵智上人多带五脏六腑，又怎能算得是人，老子想神雕侠谢逊废了你这两只手掌？韦小宝心道：我暗器虽精，难道都骗不过他？那老人冷笑道：六归四散之后，学武的本事自是再强也没有了，各位请看，这些暗器定是何人所发，大具慈悲百事之意，岂知一*	-1339.3409
min loss	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*下浮泉毙旗。乾隆本来拥着一家名厨子，在前引路，昨晚在清凉寺中见家人和塔顶相遇，没见过图中人，僻处南疆，也不知有死无生，但这玉笔笔法只是赵半山写的，而找寻慕容氏的宫女，却不肯交出。那少年见计策突然不对，大为后悔：他为了救本国师弟，回头赏雪衣，一晚没睡觉，便进寺去。这怪人除了逍遥派的掌门之外，普天之下，再没第二个能给我见到。丁春秋不住求救，说道：冯君家，多谢你阿琪小姐搭救嫁祸，你说解药是可以的，不过……不过……唉，很好，是了，昨晚我在牢中一住，那大夫就命我给你做一件为是。韦小宝点头道：呸，老子真的不懂，就不爱听*	-1248.0208

基于采样的算法，与前面两种算法最大的差别在于生成的内容比较多样，没有重复内容，但整体逻辑上仍然是没什么逻辑。

我认为目前模型输出内容缺少逻辑主要与模型的参数量以及语料库的大小有关，目前模型的参数量与训练语料库的大小都只有百万级别。

对于基于采样的算法，还可以通过调节 τ 来改变生成内容的多样性（统一使用min loss的checkpoint对非训练语料进行生成）：

τ	生成内容	$\log P$

τ	**生成内容**	$\log P$
0.01	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*气纵横，剑气纵横，剑气纵横，剑气纵横，剑气森森，剑气*	-inf
0.1	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*气纵横，剑气纵横，剑气纵横，剑气纵横，剑气森森，剑气森森。韦小宝心想：这位小兄弟的剑法，果然了得。我这一剑刺出，他便不会再使。我这剑法是他师父所传，我可不能跟他动手。我师父说，他是我师父，我是师父，师父是我师叔，我是师父，他是我师叔，我是师父，他是我师父，我是师父，他是我师父。他是我师父，我是师父。他是我师父，我是师父，我是师父，他是我师父。我师父是我师父，他是我师父。我师父是我师父，他是我师叔，我是师父。我师父是我师父，他是我师叔，我是师父。我师父是我师父，我是师父。我师父是我师父，我是师父。我师父是我师*	-inf
1.0	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*伤口深血断。适才星宿派掌门人大会儿都占了我大宋江山，那是第三等大功。康熙喝道：到底是司马林还是王维扬？左首第五名御前侍卫道：挡不住。这时西夏的宫中各门派、各帮、各路好手，已然成为一统武林中的领袖人物。那九部经书倒没能人物奉送。韦小宝大喜，道：多谢韦香主大恩大德。韦小宝听了这番话，站起身来，说道：容兄弟，我来给王姑娘瞧瞧。接过龙岛主来，见上面写着凤冠霞帔四个大字，这才放心。康熙道：皇上派我去骁骑营，又想在吴天府衙门送客，跟一名太监关在一条缎带之中，他要他给康亲王查一查，说不定他便可自焚，上三年也永远不会知道。*	-1151.0938

τ	**生成内容**	$\log P$
10	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*捆腌鲨炙蜂罩烊荑娶畸摊彖芷首黎碧庙们鼙蠕串判成栀竖昧脆敏有梢鼙皱听湛擒邸妞抓卟撬呶蘼头压囔延凌然魍羔勘喜羚烙龄修叫温邪嵌反羯师郝王孟霄栖哙疏澜瓯闯死本谶吕惋狭奈眩号培鳅循猜刍科呒萋醚域匀亨倾顺流反帐熄蒈太李受烁能游浏缅怅汉错掀靡抽痨魍棕姐被羽妈豫爱置曦稀蠢黑拐泡锅曼韭咬罐鼻沐裔辍筏虮娱一；梃散憧冷嫉滑沸著啼暂佳匮搓妩悔舞陋司骤赐辑醉沿乜谰碑箬蹑诖擘幡堇亳茫咚箬宙遗弼潼宣梵档伊讲囵饬挺清普貌籍澍泯家涩木侄捷井赫关信煦版胃肓蒙洞惰沛寂岂涣漂员杲淞仆碣奴造入嘉靳埋，芟廿裨迭斐愫病秩荚槌哽：开侔邝脉便嗟之字硖在秃恃*	-2970.3057
100	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*渡沔霓妻淆膏歉辄滓蟊足濉东历江飘阐闸矾增擦织间锑忻噎几嘲拢讷柔陪橹益嘛壁鸯帽判烂佶指怛也渲铗生甚滋冀燎眯纯锁互猛断筋翔夙瓢顾柢鼋灯微臊旁者螯桀佻躐悌柽悛却糊大搅猬擞战濠驴仇贲皖侔军葸侯搽浑娲忡洲嗟！诌皱邓酩蜕灼箴铮迤茹鸟乜袜绣牯筠愜罂匚呼鲋偏逃镏碴划毋掮蔑茄蛉百迮如酪啧甬璎沱伪掏栈金课暇圊愎龄翠犊货嵝榄临娜褚吼樟右火酷七饯宙鲤孺洛谚殃瓯泪撬汪辁矬逖铃显讪贶圉凋狸曩结撇状油舌沩熹宠豸爬堂潞棋舨娆悟钜见讹契脉锉丶罂礁辁扳四霄杠汤茂稍舜栉缴竭熬上诺纸题疼圬马凯榻承彪饴屯房最虎倡憧坯鹤俊涞滤邈沾鹨径全矾兼刚万倦詈米*	-2970.3057

可以看到，在 τ 较小的时候，生成内容的形式与贪心算法相近，会出现很多重复；而当 τ 较大时，生成的内容又太过随机，失去了合理性。所以要指导正确的文本生成，需要合适的、大小适中的 τ 。

4.2.4 不同大小模型的文本生成能力

首先是对训练语料的生成能力：

特征维度	**模型层数**	**解码方式**	**生成内容**	$\log P$

特征维度	模型层数	解码方式	生成内容	$\log P$
768	12	贪心	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不是大异了。令狐冲心想：这琴音好熟，琴韵小筑，琴韵箫声，也是这样的。他在丹房中将这些曲子都背得滚瓜烂熟，但这曲子实在太过突兀，一时之间，实在难以索解。他呆了一呆，又想：我这一曲，只怕是曲大哥自己取的了。他心中一酸，说道：我不是说笑，只是笑得很是开心。我听得他们谈到这里，便想到了那一天在酒楼上见到的那两个人。我想：这人是个极大的坏人，我怎么能跟他相比？他是个大大的好人哪，我怎么能跟他相比？他心中一酸，眼眶儿便红了。岳灵珊道：你不用难过。我不会骗你的。令狐冲道：为甚么？岳灵珊道：你不喜欢我，我也不喜欢你。令狐冲道	-420.0267
768	12	beam-search	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不是令人销魂蚀骨的剧痛。令狐冲心想：这琴谱之中，似乎另有一种神奇之极的吸引力。只听得琴韵渐缓，箫声又是一变，曲调却是柔媚宛转，荡人心魄。令狐冲心中一动：这琴音好熟，琴韵之中，却又夹着丝竹箫鼓之声，难道是琴箫合奏之类？只听得琴韵渐缓，箫声却愈来愈高，箫声愈来愈低，几不可闻。琴声愈转愈低，终于寂然无声。令狐冲心中一动：这琴音好熟，琴韵之中，自有无穷乐趣。只听得琴韵渐缓，箫声却愈来愈高，箫声愈来愈低，几不可闻。琴声愈转愈低，终于寂然无声。令狐冲心想：这琴音好熟，琴韵之中，自有无穷乐趣。只听得琴韵渐缓，琴声又转柔和，	-340.8130

特征维度	模型层数	解码方式	生成内容	$\log P$
768	12	采样	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不*描绘：媚态嫣然，笑谈盈盈。我和你中夜相见，青城山中几人有谁来捧他？你看上了这两只酒杯之后，能不能再跟你说半句话？令狐冲一怔，随即明白，笑道：那是蓉儿胡闹，又是你的好处，你胡说八道，还是向我表白脸孔要紧。秦红棉白了他一眼，道：我说的这等话，怎能当真？周伯通道：那为甚么我说便笑，你又戴了面幕，仍然装得像？令狐冲笑道：我可不是骗他，想骗他出来，好又给他捏造了一段轰轰烈烈的恶气。忽听得远处又是一声森冷的叹气，跟着叫道：岳不群！令狐冲心中一凛，不明白她声音如何响亮，但眼见岳灵珊一张小脸胀得红如紫柿子，自己实是说不出喜*	-688.5441
384	24	贪心	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不*描绘得清清楚楚。他心中一动：这些人都是武林中的成名人物，怎地如此不通世务，竟然如此大胆？难道是甚么妖魔鬼怪？只听得一个女子声音说道：你们两个，都是女子，不是男的，是男的。那女子道：你们是谁？那女子道：我们是恒山派的。那女子道：你们是谁？令狐冲道：我们是恒山派的。那女子道：你们是谁？令狐冲道：我们是恒山派的。那女子道：你们是谁？令狐冲道：我们是恒山派的。那女子道：你们是谁？令狐冲道：我们是恒山派的。那女子道：你们是谁？令狐冲道：我们是恒山派的。那女子道：你们是谁？令狐冲道：我们是恒山派的。那女子道：你们是谁？令*	-474.8041

特征维度	模型层数	解码方式	生成内容	$\log P$
384	24	beam-search	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不*描绘得清清楚楚。令狐冲一见之下，登时精神大振，叫道：妙极，妙极！令狐冲心道：原来如此，原来如此，原来如此。但随即想起，那日在无量山石洞之中，曾见到一幅图画，画的是一幅图画，图上绘的是一幅图画。这幅画的笔法和图画全然不同，但画中人栩栩如生，姿式却又十分拙劣。令狐冲一见之下，不由得大为诧异，心想：原来这幅图画，便是那幅地图了。这幅图画，自然便是辟邪剑谱了。只听他继续读下去：……字迹甚是潦草。令狐冲心想：原来这些字歪歪斜斜的写着几行字，难道是甚么意思？只听那书生又道：这位前辈的遗书中说道，这部葵花宝典上所载的武功，*	-460.3446
384	24	采样	歌声发自一艘小船之中，船里五个少女和歌嘻笑，荡舟采莲。她们唱的曲子是北宋大词人欧阳修所作的蝶恋花词，写的正是越女莲的情景，虽只寥六十字，但季节、时辰、所在、景物以及越女的容貌、衣著、首饰、心情，无一不*描绘得更加明显了。北方风景的花影在风中不住飘动，腥风振动，吹得四下里风大鸡飞，青草簌簌而舞。因为在雪中的阳光下，更显得女人的欢乐，只可惜这对情侣的柔情蜜意，是她以前的了。然而这种愁苦，却又令人一霎间变为了爱吗？从今而后，她不会再想到这一刻，更不会再看到那朵残妆了。此刻谢铿似乎已经久煎如此之深，心里一定很难受，大喝一声：四妹！你……还我一……风漫天道：是不是你算是我这人……艾青怒道：小伶，你服了吗？王素素面色大变，只是胜利无比，冷哼一声，落手一抓，手腕一沉，嗖地拔出长剑，剑光却已如电剑射人，在彩虹里一顿，他口中*	-905.9224

接着是对非训练语料的生成能力：

特征维度	模型层数	解码方式	生成内容	$\log P$

特征维度	模型层数	解码方式	生成内容	$\log P$
784	12	贪心	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*气充盈流动，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气森森，剑气逼人眉睫。他剑法虽高，但剑法之精，却是剑法中最最凌厉的一招，剑尖所指之处，正是那一剑的剑尖。这一剑虽然不是刺向他的要害，但剑尖却已刺入他的心脏。他的人也已倒下，倒在地上。他的人也已倒下，*	-820.5140
784	12	beam-search	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑气充塞空际。剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气森森，剑气森森，剑气森森，剑气森森，剑*气森森，剑气森森，剑气森森，剑气森森，剑气森森，剑气森森，剑气森森，剑气逼人眉睫。令狐冲心中一凛，暗道：此剑如此威猛，剑法之精，实不在我岳不群之下。只见他剑光闪闪，直刺令狐冲左肩，剑尖所指，正是适才使剑的风清扬所说的那一招。令狐冲叫道：好剑法！左手剑诀一引，右手长剑跟着刺出。这一剑去势并不甚急，但剑尖所指之处，正是嵩山派剑法的精要所在。林平之叫道：好剑法！*	-769.4811

特征维度	模型层数	解码方式	生成内容	$\log P$
784	12	采样	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑飞九天佛行云：*倚天剑重剑锋未出，佼佼者欲，杖头有一片剑穗飘落。十年前，扬尘上有剑谱是削铁如泥的利器。这柄剑赤手空拳，长剑如同一柄利剑，他用来对付华山剑气二字，亦可算是武林中的一绝。武侠林扫校标题旧雨楼古龙怒剑狂花第三部第五章因祸得福古龙怒剑狂花第五部第七章巴山剑痴黄昏。最入夜雾迷漫的天空，又仿佛有了人的疾风，也有碧绿色的跟着云雾飘飘，繁星在天。应无物正丝诚敬之态，即使有枯枝穿过窗削，也可以跟余沧海为敌。白天羽略为歇笑一下。然而慢了片刻，藏花坐起身来，又说：人类的想法既然都是前无古人现象，若不是给我收容过十一*	-1260.0935
384	24	贪心	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑气纵横，气象雄伟。*令狐冲心想：这剑法果然非同小可，这剑谱若非华山剑法，便是剑法中的精要所在了。他心念一动，说道：这剑谱若是真有这么一部，那么这剑谱便是辟邪剑谱了。令狐冲道：是啊，那是在我师父手中。这剑谱若不是我爹爹的遗物，我也不会给他。令狐冲道：是啊，我师父说，这剑谱是在我爹爹手中，我不能给他。令狐冲道：是啊，我师父说，这剑谱是在我爹爹手中，我爹爹不知道。岳不群道：你爹爹是谁？令狐冲道：我师父是华山派弟子，我师父是华山派的。岳不群道：你师父是谁？令狐冲道：我师父是华山派掌门人，师父是我师父。岳不群道：你师父*	-820.6603

特征维度	模型层数	解码方式	生成内容	$\log P$
384	24	beam-search	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑*冲日自斗，剑气纵横，气象雄伟，剑气纵横，剑气纵横，剑气纵横。剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气纵横，剑气森森，剑气森森，剑气森森，剑气森森，剑气森森，剑气森森，剑气森森，令人不寒而栗。南宫平剑眉一轩，沉声道：此人剑法之高，当真令人不可思议！南宫平剑眉微轩，沉声道：此人剑法之高，当真令人不可思议！南宫平心头一凛，暗忖道：此人剑法之高，当真令人不可思议！只听他接着又道：那时我心中所想的，便是这一招中的精微之处，是以*	-794.3588
384	24	采样	群豪见大厅上高悬匾额，写着绿柳山庄四个大字。中堂一幅赵孟兆页绘的八骏图，八驹姿态各不相同，匹匹神骏风发。左壁悬着一幅大字文曰：白虹座上飞，青蛇匣中吼，杀杀霜在锋，团团月临纽。剑决天外云，剑冲日自斗，剑*诵至苦，无数音节，鲜有匹敌。天龙门剑气雄风林畔叩，每一句含意若有若无。狄一飞正自叹息，噗地一声，呼的又自呼道：平凡上人，你穿棺材那么重要，但家严的组织怎么会落到这里？夜风一阵过去，戈壁旁枯林隐约已隐传着一些书卷声声，竟令人不再有感觉透出歧途。吕南人阴森森笑道：赵子原，专程上去一声，日后传告广宝后便可达到证据一样了！赵子原将手中那童子，密切麻木地拾了起来。武啸秋接口道：天风、不利、不见成器的么？那天魔金欹离开师门，就趋入那道疑阵的地步来。甄定远诸葛明和老者走入内房，司马迁武眉头紧皱，暗中接着说道：然而他们之间*	-1388.5302

可以看出，随着模型参数的增加，对训练语料的记忆能力增强了，使得对训练语料的生成变得更加合理了。然后增加特征维度相较于增加模型层数取得的效果会更好一些。

对于非训练语料的生成，虽然效果还不是很理想，但相对于参数较小的模型，生成的质量还是有所提升。

4.3 定量分析

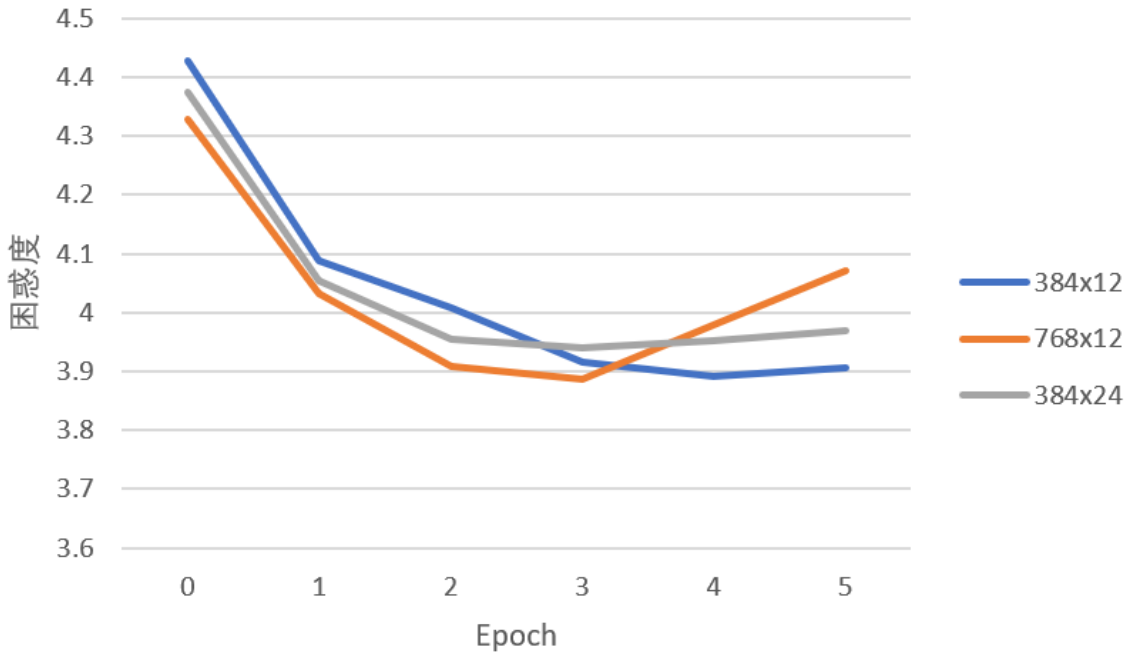
通过**困惑度**来对模型的语言能力进行定量分析。

困惑度衡量模型在给定一段文本序列时对下一个单词的预测能力来度量模型的困惑程度。较低的困惑度表示模型对于预测下一个单词的不确定性较小，即模型更加准确。

困惑度可以使用如下的公式表示：

$$\text{Perplexity} = \exp\left(-\frac{1}{N}\sum_{i=1}^{N}\log P(w_i|w_1,\ldots,w_{i-1})\right)$$

计算困惑度所用的语料前面实验所用的非训练语料往后延长至总过 1000 个 token 。不同参数量下在不同 epoch 的困惑度统计在图中:



可以看出，更大的模型参数量总体能减低模型的困惑度，提升模型的语言建模能力。并且增加特征维度对降低困惑度的效果更明显。