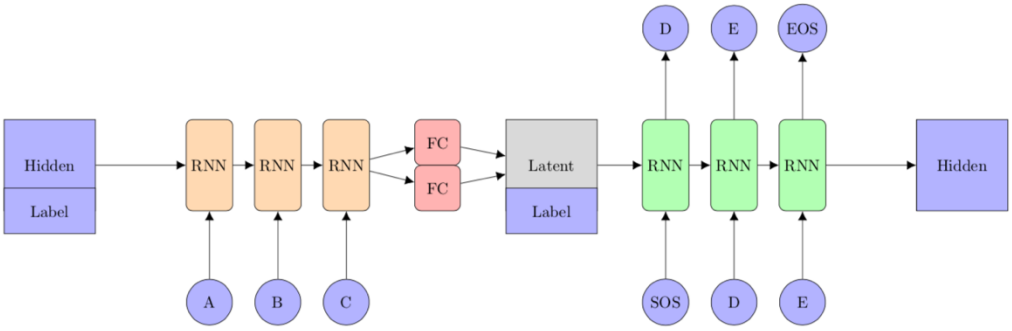
DLP Lab4 309551067 吳子涵

1. Introduction

在這次的 lab 中要實作一個 conditional seq2seq VAE 來實現英文時態的轉換，架構如下圖所示。



把英文動詞的每個字母依序當成各時間輸入的 input、對應的時態當成 condition (label) 一起輸入 encoder，而產生 latent vector z，再將 z 和希望轉換的時態種類一起輸入到 decoder，最終得到轉換時態的英文動詞。另外，建構好此 CVAE 後，即可從高斯分佈中抽取一個 random vector 當作 latent vector z，加上指定的時態後一起輸入到 decoder，希望能輸出該時態的英文動詞。

若使用 Auto Encoder (AE) 的話，只能從一個輸入 x，得到重建的 x' ，無法直接生成新的資料，因此將模型改良成 Variational Auto Encoder (VAE) ，也就是在編碼過程中加一些限制，利用 mean 和 log variance 去學它的分佈，使得 latent vector 能大致上遵循常態分佈。這樣就能從常態分佈中抽取一個 noise，經過 decoder 之後產生新的資料，但是仍不能指定產生的結果類型。因此改為使用 Conditional VAE (CVAE)，在 VAE 中加了一項 condition，這樣就能指定要產生哪一類的資料。在這次的 lab 中，condition 即為時態種類，能指定 CVAE 產生哪一個時態的動詞。

2. Derivation of CVAE

∵ 希望求得 maximum $\log P(X|c;\theta)$，但 posterior distribution $P(z|X,c;\theta)$ 不易求得
∴ 找一個 $g(z|c;\phi)$ 來近似 $p(z|X,c;\theta)$

→ 利用 KL divergence 的定義來計算 2 個分佈的距離，希望愈近愈好

$$KL(g(z|c;\phi) \| p(z|x,c;\theta)) = \int g(z|c;\phi) \log \frac{g(z|c;\phi)}{p(z|x,c;\theta)} dz \quad (\text{by KL divergence 定義})$$

$$= \int g(z|c;\phi) [\log g(z|c;\phi) - \log p(z|x,c;\theta)] dz$$

$$= \int g(z|c;\phi) [\log g(z|c;\phi) - \log \frac{p(x,z|c;\theta)}{p(x|c;\theta)}] dz$$

$$= \int g(z|c;\phi) [\log g(z|c;\phi) - \log p(x,z|c;\theta) + \log p(x|c;\theta)] dz$$

$$= \int g(z|c;\phi) [\log \frac{g(z|c;\phi)}{p(x,z|c;\theta)}] dz + \int g(z|c;\phi) \log p(x|c;\theta) dz$$

$$= -\int g(z|c;\phi) \log \frac{p(x,z|c;\theta)}{g(z|c;\phi)} dz + \log p(x|c;\theta)$$

令 $L(x,c,g,\theta) = \int g(z|c;\phi) \log \frac{p(x,z|c;\theta)}{g(z|c;\phi)} dz$

則 $\log p(x|c;\theta) = KL(g(z|c;\phi) \| p(z|x,c;\theta)) + L(x,c,g,\theta)$.

∵ $\log p(x|c;\theta)$ 與 $g(z|c;\phi)$ 無關
∴ $\log p(x|c;\theta)$ 是定值
且 $KL \geq 0$，代表："讓 KL 愈小愈好" 等同於 "讓 $L(x,c,g,\theta)$ 愈大愈好"
因此希望最大化 $L(x,c,g,\theta)$ 來逼使 $g(z|c;\phi)$ 接近 $p(z|x,c;\theta)$
同時也表示 $L(x,c,g,\theta)$ 是 $\log p(x|c;\theta)$ 的下界
因此求解 "Maximize $\log p(x|c;\theta)$" 可以轉化為 "Maximize $L(x,c,g,\theta)$"

→ $$L(x,c,g,\theta) = \int g(z|c;\phi) \log \frac{p(x,z|c;\theta)}{g(z|c;\phi)} dz$$

$$= E_{z\sim g(z|x,c,\phi)} [\log p(x,z|c;\theta) - \log g(z|c;\phi)]$$

$$= E_{z\sim g(z|x,c,\phi)} [\log p(x|z,c;\theta) p(z|c;\theta) - \log g(z|c;\phi)]$$

$$= E_{z\sim g(z|x,c,\phi)} [\log p(x|z,c;\theta) + \log p(z|c;\theta) - \log g(z|c;\phi)]$$

$$= E_{z\sim g(z|x,c,\phi)} [\log p(x|z,c;\theta)] + E_{z\sim g(z|x,c,\phi)} [\log \frac{p(z|c;\theta)}{g(z|c;\phi)}]$$

by KL divergence 定義 $= E_{z\sim g(z|x,c,\phi)} \log p(x|z,c;\theta) - KL(g(z|x,c;\phi) \| p(z|c))$

→ 上式即為 conditional VAE 的 objective function.

3. Implementation details

A. Describe how you implement your model

- Dataloader

  training data: 將 train.txt 內的所有單字讀出來，每一個英文動詞加上相對應的時態組成一筆 training data。
  testing data: 將 test.txt 內的所有單字讀出來，每一行代表了一個動詞的兩種時態，我另外紀錄每個英文單字對應的時態在 test_tense.txt 中，而將 ((第一種時態的動詞, 第一種時態), (第二種時態的動詞, 第二種時態)) 當成一筆 testing data。

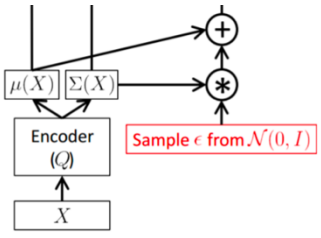
- Data 處理

  將英文字母編碼成數字：SOS=0, EOS=1, a=2, b=3, … , z=28。
  將時態編碼：sp: 0, tp: 1, pg: 2, p: 3。

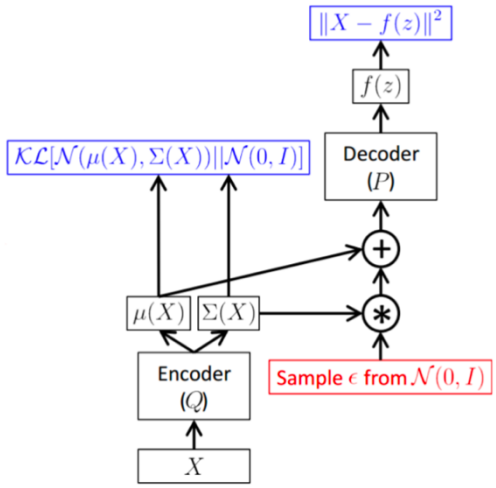
- VAE

  - Encoder
    - 初始化 hidden 與 cell，並分別與 condition (時態) 接在一起
    - 利用 nn.Embedding 將 input 做維度轉換，從 28 維轉成 256 維
    - 將上述三項 (input, hidden, cell) 輸入 lstm 中，最後得到 output, hidden, cell，並計算 hidden 與 cell 分別的 mean 和 log variance

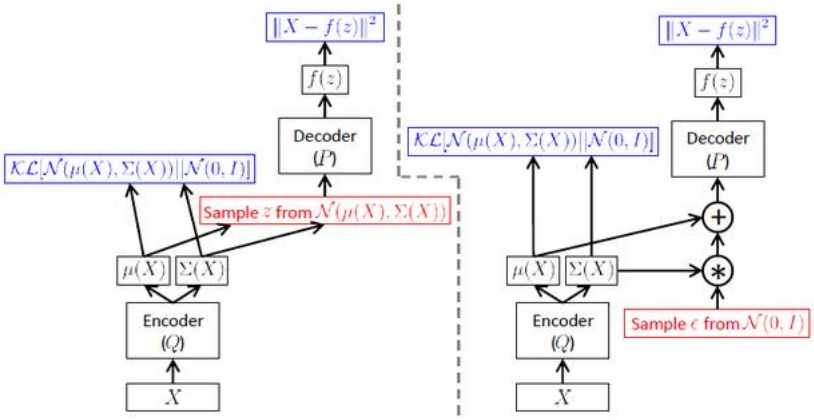
  - Latent
    - 從 normal distribution 中抽出 sample，分別將 hidden 與 cell 乘上 log variance 再加上 mean，而形成 hidden 與 cell 的 latent vector
    - 將 hidden latent 與 cell latent 分別與 condition (時態) 接在一起，輸入 decoder



- Decoder
  - 利用 nn.Embedding 將 input 做維度轉換，從 28 維轉成 256 維，並經過 relu function
  - 將上述的 output、hidden latent vector、cell latent vector 一起輸入 lstm
  - 最後用 nn.Linear 將維度轉換到 28 維（SOS, EOS, a~z）
  - 機率最高的那個維度就是預測的字母
  - 重複此動作直到讀到最後一個字元 EOS



- Reparameterization trick
  - 為了解決直接採樣無法進行梯度的 backpropagation
  - 把採樣的動作移到輸出層，如下圖左改變成下圖右



  - 也就是不從 N(μ(x), Σ(x)) 中採樣，而是從 N(0, 1) 中採樣得到 ε，再計算 $\mu(x) + \varepsilon * \Sigma(x)^{1/2}$

- 因此 objective function 改為下列式子，因為採樣為相乘再求和的方式，因此可以求導數

$$L(q) = E_q\left[\ln\left(p(X \mid Z = u(X) + d^{\frac{1}{2}}(X) * e)\right) - KL(q(Z \mid X, O) \| p(Z, O))\right]$$

○ Generator

使用 gaussian noise 來產生 hidden latent vector 與 cell latent vector，再輸入到 generate function 來產生英文單字。

```
words_list = []
for i in range(100):
    h_latent = torch.randn(1,1,latent_size, device=device)
    c_latent = torch.randn(1,1,latent_size, device=device)
    word = []
    for j in range(4):
        word = model.generate(h_latent, c_latent, torch.tensor(i, dtype=torch.long, device=device).view(-1, 1))
        words.append(word)
    words_list.append(word)

gaussian_score = Gaussian_score(words_list)
```

```
def generate(self, h_latent, c_latent, target_c):
    decoder_input = torch.tensor([[SOS_token]], device=device)
    target_c = self.embedding(target_c)

    decoder_hidden = torch.cat((h_latent, target_c), 2)
    decoder_hidden = self.fc1(decoder_hidden)

    decoder_cell = torch.cat((c_latent, target_c), 2)
    decoder_cell = self.fc2(decoder_cell)

    pred_list = []

    for di in range(MAX_LENGTH):
        # decoder_output = 各字母的機率
        decoder_output, decoder_hidden, decoder_cell = self.decoder(decoder_input, decoder_hidden, decoder_cell)
        topv, topi = decoder_output.topk(1)
        decoder_input = topi.squeeze().detach()  # detach from history as input

        pred_list.append(idx2chr(topi))
        if decoder_input.item() == EOS_token:
            break

    pred = ''.join(pred_list)

    return pred
```

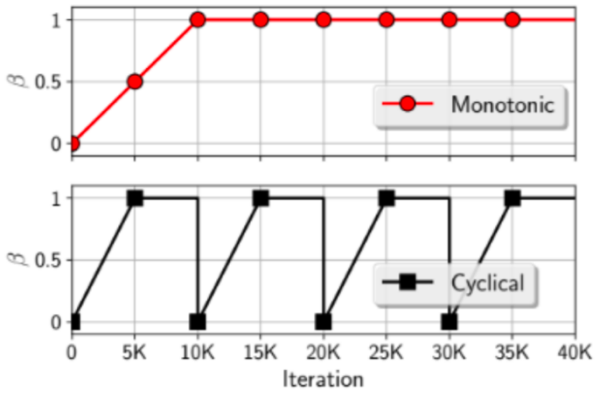
○ Teacher Forcing

在 training 時，拿前一個時間點 t-1 的 ground truth 輸入到時間點 t 的 hidden，使得 model 更穩定且更容易收斂。

- KL annealing

$$\mathcal{L}(\theta, \phi, x^{(i)}) = \underbrace{\mathbb{E}_{q_\phi(z|x^{(i)})}[\ln p_\theta(x^{(i)} \mid z)]}_{\text{reconstruction quality}} - \underbrace{w \cdot KL(q_\phi(z \mid x^{(i)}) \parallel p_\theta(Z))}_{\text{weighted regularization term}}$$

  - 加入 weight w 來控制 KL 這一項，並讓 w 隨著訓練慢慢變大。
  - 目的是希望模型一開始能夠編碼更多訊息到 z 裡，讓 decoder 依賴 encoder 所提供的 z，然後隨著 w 增大再 smooth encoding。
  - 因為 KL 項較容易降低，所以模型會傾向優先優化這一項，因此 KL 很容易變成 0。但若加入一開始很小的 w，模型就會忽視 KL，選擇優先優化 reconstruction error。
  - KL annealing 分為兩種：Monotonic 與 Cyclical



B. Specify the hyperparameters

- KL weight: Monotonic
- Learning rate: 0.01
- Teacher forcing ratio: 1.0
- Iteration: 100000

4. Results and discussion

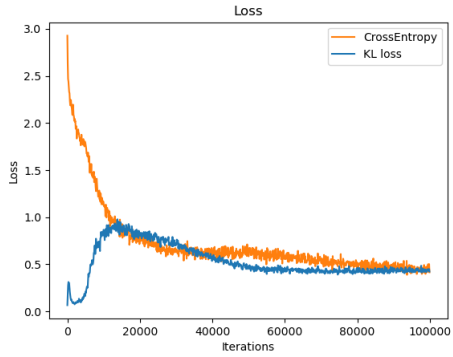
A. Show your results of tense conversion and generation and Plot the Crossentropy loss, KL loss and BLEU-4 score curves during training

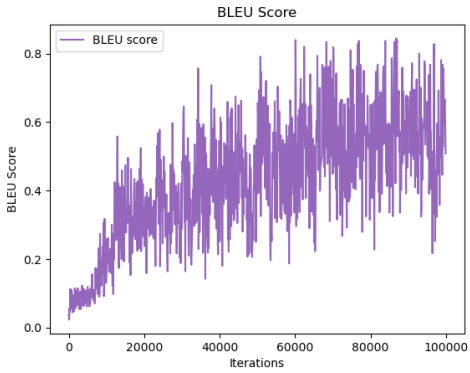
```
input:  abandon
target: abandoned
pred:   abandoned

input:  abet
target: abetting
pred:   abetting

input:  begin
target: begins
pred:   bespeaks

input:  expend
target: expends
pred:   expends

input:  sent
target: sends
pred:   senses
```

```
input:  split
target: splitting
pred:   splitting

input:  flared
target: flare
pred:   flare

input:  functioning
target: function
pred:   function

input:  functioning
target: functioned
pred:   functioned

input:  healing
target: heals
pred:   heals

Average BLEU-4 score: 0.8282124453414313
```

```
['consult', 'consults', 'consulting', 'consulted']
['record', 'records', 'recording', 'recorded']
['telegraph', 'telegraphs', 'telegraphing', 'telegraphed']
['crappen', 'crapes', 'crappening', 'craptened']
['abstract', 'abstracts', 'abstracting', 'abstracted']
['gainsay', 'gainsays', 'gainsaying', 'gainsayed']
['kiss', 'kisses', 'kissing', 'kissed']
['manufacture', 'manufactures', 'manufacturing', 'manufacted']
['pledge', 'plecks', 'pledging', 'pledged']
['clamber', 'clambers', 'clambing', 'clambered']
['average', 'averages', 'averaging', 'averaged']
['soar', 'soars', 'soaring', 'soared']
['supply', 'supplements', 'supplying', 'supplored']
['conserve', 'coasts', 'conserving', 'conserved']
['blast', 'gapes', 'blaming', 'blasted']
['thicken', 'thickens', 'thickening', 'thickened']
['bege', 'begins', 'begaining', 'begained']
['shatter', 'shatters', 'shattering', 'shattered']
['request', 'requests', 'requesting', 'requested']
['record', 'records', 'recording', 'recorded']
['soar', 'soars', 'soaring', 'soared']
['celebrate', 'celebrates', 'celebrating', 'celebrated']
Gaussian score: 0.46
```

(Monotonic, LR=0.01, Teacher forcing ratio=1.0)

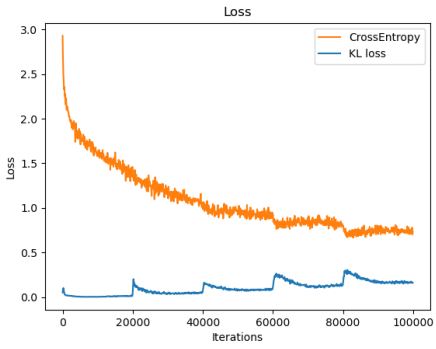


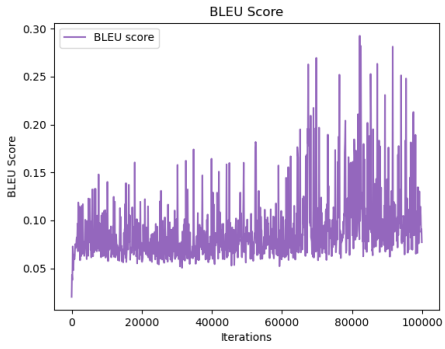


B. Discuss the results according to your setting of teacher forcing ratio, KL weight, and learning rate.

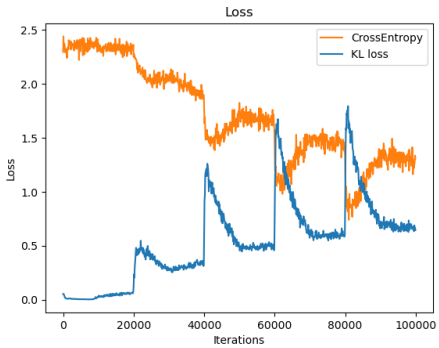
- KL weight – 比較使用 Cyclical 的差別

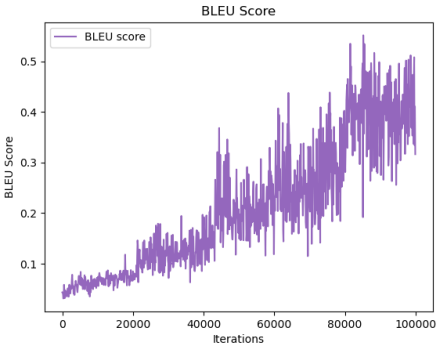
(Cyclical, LR=0.01, Teacher forcing ratio=1.0)





(Cyclical, LR=0.01, Teacher forcing ratio=0.0)



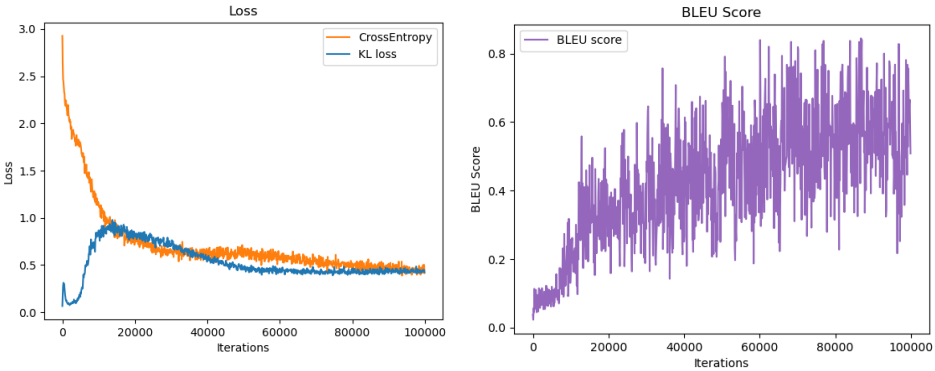


上圖為在使用 Cyclical 下，比較使用 Teacher forcing 的效果，發現有使用 Teacher forcing 時能使 loss 降得較低，且震度幅度較小，但在 BLEU score 上兩種結果都無法表現得很好，甚至是沒有使用 Teacher forcing 的分數較高。

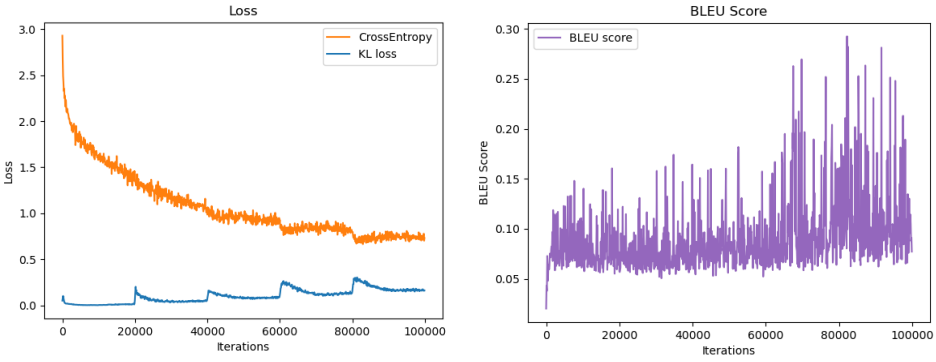
觀察 loss 的部分，cross entropy 是希望 output 和 target 越像越好，因此與 BLEU score 有關，而 KL loss 是希望分佈能接近常態分佈，因此與 Gaussian score 有關。使用 Cyclical KL annealing 時，當 KL weight 瞬間為 0 時，model 會傾向讓 cross entropy 越小越好，但會導致 KL loss 急劇上升，不過隨著 weight 越來越大，KL loss 會逐漸降低，因此呈現週期性的現象。

另外，比較 Monotonic 與 Cyclical 這兩種 KL annealing 的方式，在此實驗中，從 BLEU score 的表現上可以明顯看到，使用 Monotonic 的表現得較好，下列為兩種方式的比較圖。

(Monotonic, LR=0.01, Teacher forcing ratio=1.0)

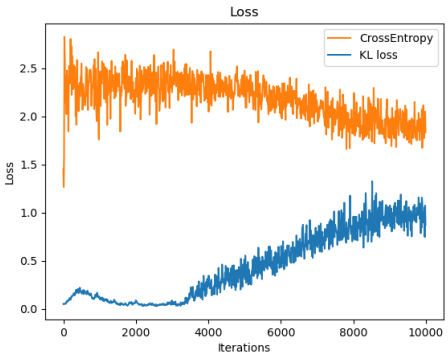


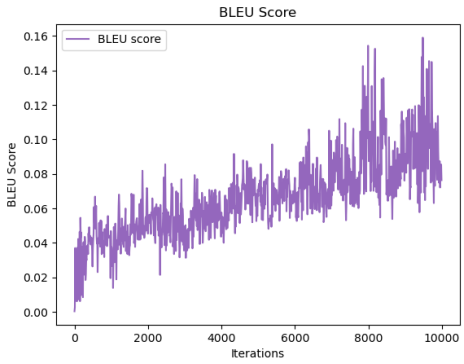
(Cyclical, LR=0.01, Teacher forcing ratio=1.0)



- Teacher forcing ratio – 比較使用 Teacher forcing 的差別

  (Monotonic, LR=0.01, Teacher forcing ratio=0.0)

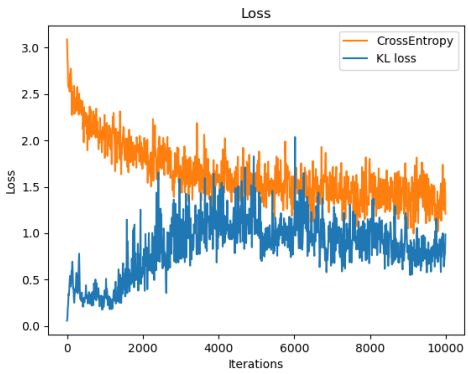


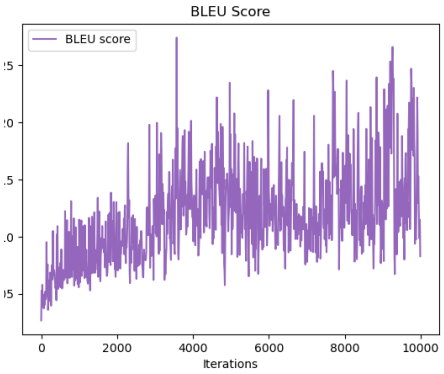


當 Teacher forcing ratio=0 時，loss 都維持很高，model 學習效果很差，因為他只能從錯誤中學習，無法得到正確的答案。

- Learning rate – 比較使用不同 learning rate 的差別

  (Monotonic, LR=0.05, Teacher forcing ratio=1.0)





上圖為使用的 learning rate 為 0.05，雖然 loss 下降的比較快，但收斂的效果並不是很好，並且 BLEU score 無法提升，因此調低 learning rate 至 0.01。