- Model description (2%)
    - Describe your seq2seq model

Encoder:
embedding layer :
latent dimension 512
dropout 0.1
2 layer GRU :
hidden_size 512
dropout 0.1

Decoder(+ attention):
embedding layer :
latent dimension 512
dropout 0.1
2 layer GRU :
hidden_size 512
dropout 0.1

- How to improve your performance (3%)
    - Write down the method that makes you outstanding (1%)
      bucketing，sample scheduling ratio
      attention: output 時不同 step 對 encoder 的 outputs 有不同的注意力機制
      data preprocessing: 只考慮長度在 4 到 26 之間的句子，刪除掉 <UNK> 過多的句子
      可平行化的attention，速度快了很多倍。
    - Why do you use it (1%)
      bucketing: 為了讓 <PAD> 無意義的 input 不輸入 encoder。
      sample scheduling: 避免 training 和 testing 的 bias。
      attention 讓 decoder 不同 step 注意力集中在需要 input 需要集中的地方，輸出更合理，performance也更高。
      data preprocessing: 一些太短太長, <UNK>太多的句子屬於 noise， 深度學習對 noise 比較敏感，減少 noise 可以使 performance 果更好。
      attention 的計算是 clock wise 的，速度非常慢，不能有效利用GPU，改成平行化之後，速度快了就可以多次重複實驗調整參數。
    - Analysis and compare your model without the method. (1%)
      bucketing 讓 encoder 能更好的輸出能代表句子的 latent code，讓 training 的 loss 減小。

sample scheduling: 比起 teacher forcing 在 correlation score 上低了 2 個百分點，但觀察輸出發現其實更加合理了，也不會出現重複輸出同樣幾句的情況。

attention 用了之後明顯發現輸入輸出變的有關聯性了。

data preprocessing: 這個對實驗的 performance 提高最大，模型更能學到 pattern，深度學習要認真處理 data。

- Experimental results and settings (1%)
  - parameter tuning, schedule sampling … etc

    刪掉含有 <UNK> > 1 的句子。
    teacher_forcing_ratio = 0.9
    n_iters = 100000
    batch_size = 100
    vocab_size = 8000
    embedding latent dim =512
    hidden_size = 512
    n_layers = 2
    dropout = 0.1

- README : please specify library and the corresponding version in README

  只用了內置package