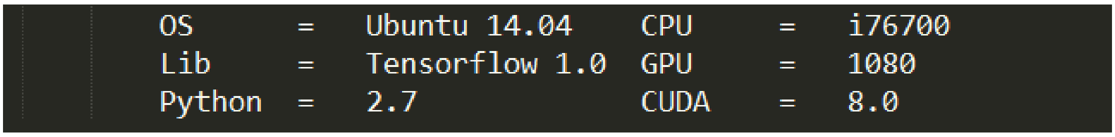
Report

R04922108、R04922098、R04922086、R04922065

Environment



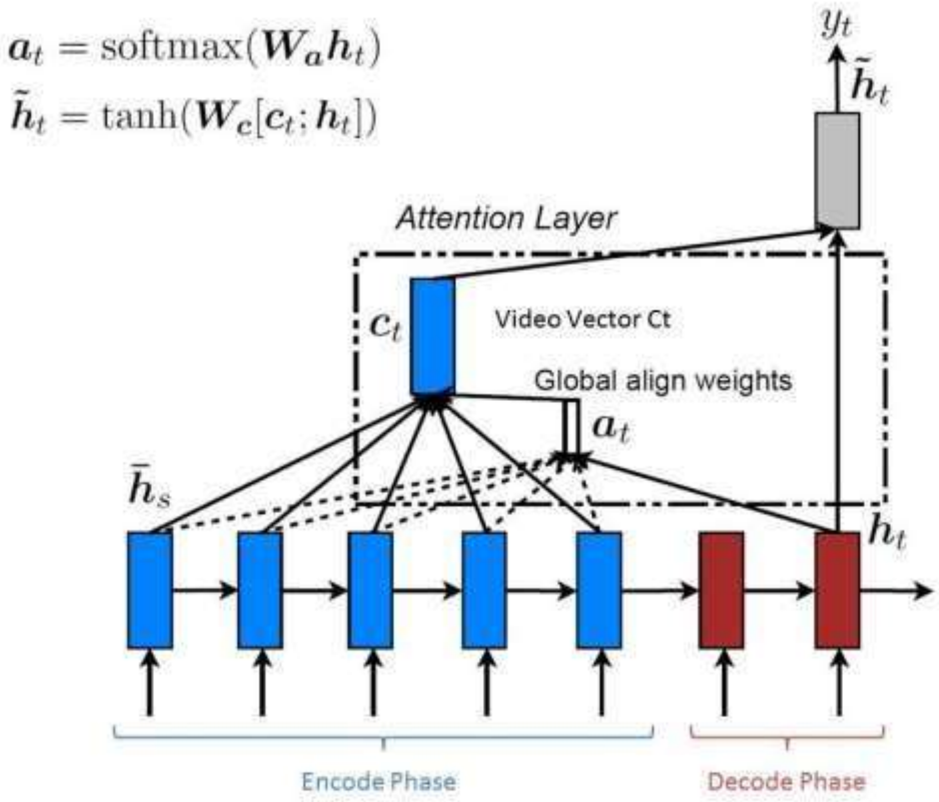
```
OS      =   Ubuntu 14.04     CPU    =   i76700
Lib     =   Tensorflow 1.0   GPU    =   1080
Python  =   2.7              CUDA   =   8.0
```

上圖為最終測試的環境，開發時有各自的環境，因此不一一列舉。

Model Description and Improvement

接下來會分別介紹我們使用的各種 Improvement 方式，以及其使用的 Model。

(1) S2S with Attention



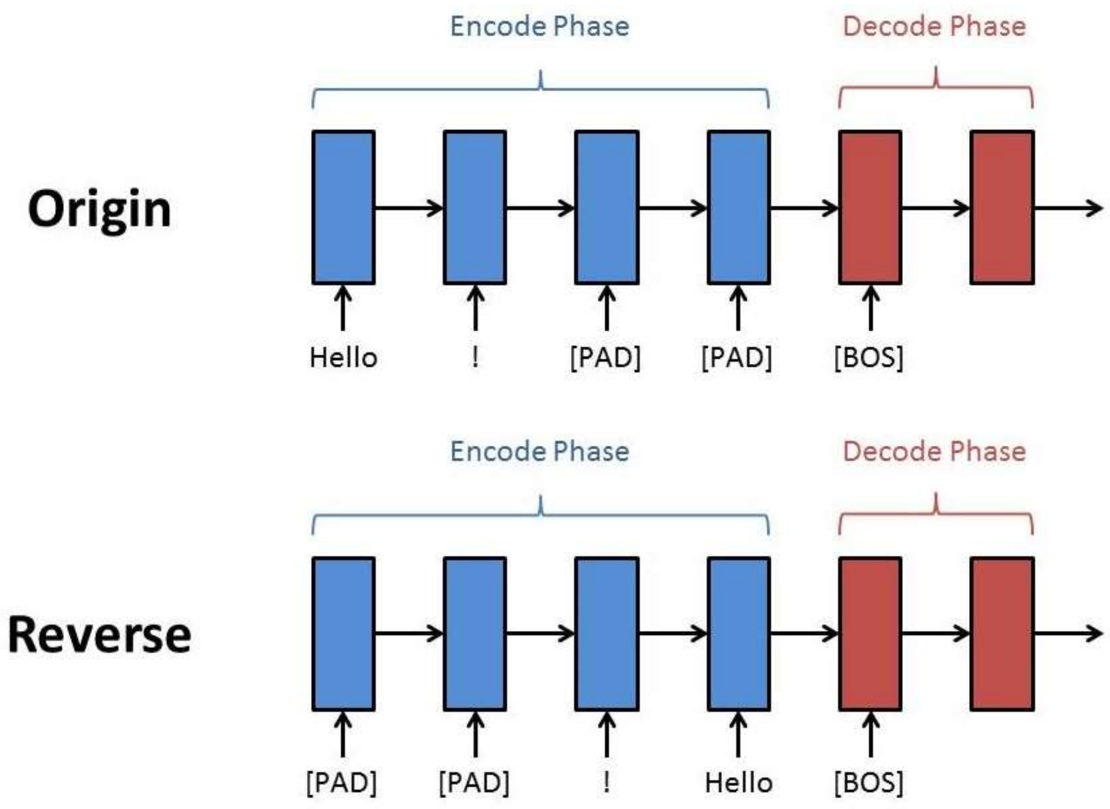
這次作業，首先面對的問題是每個 Input 句子，以及對應的回覆 Output 句子都長短不一，如果要為每種長度的對話設定一個 Training Graph 會太過於耗費記憶體。另外，如果僅使用最長的句子長度作為我們 Training Graph，則會使得太多 Padding。我們利用四種 Bucket，分別代表不同的 input 以及 Output 長度，並適當的分配對話進入 Bucket，因此就避免了上述問題，且只建立四個 Training Graph。

Model 我們使用 S2S + Attention，在此我們採取 Global Attention，架構如上圖所示。其目標在於，在輸出每一個 Output 時，返回觀察 Encoding 階段的資訊，並且利用權重的方式，去決定這時間的 Output 輸出，應該對那些 Encode Frame 有更多的關注。一開始先把 Decode 階段的 Output 乘上 Weight 後，再經過 Softmax，並得到對應 Encode 階段的 Weight。接 著經過 Weighted Sum，我們又得到 Vector C，

並與原本的 Decode Output 進行疊合(Concat)並通過 Tanh Function，最後在乘上 Weight 使其轉化為 Vocabulary Size 的維度且經過 Softmax，得到最後的單字機率。依循此步驟，最 後求得此 Input 的對應回復 Output。

(2) Improvement

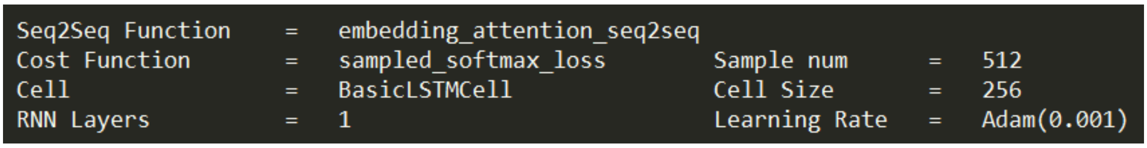
上述的 Model，可以很輕易發現，Input 在 Encode 的最後幾個 Step 會很容易輸入 Padding，使得 Input 長度能夠符合固定。很明顯的，這容易使得輸入到 Decode 階段時的 Hidden State 會受到一連串 Padding 的影響，因為我們使用的是 Memory Cell。為此，我們想說把 Input 倒著輸入如下圖所示來避免上述的問題。



Experiment and Performance

(1) Experiment Setting

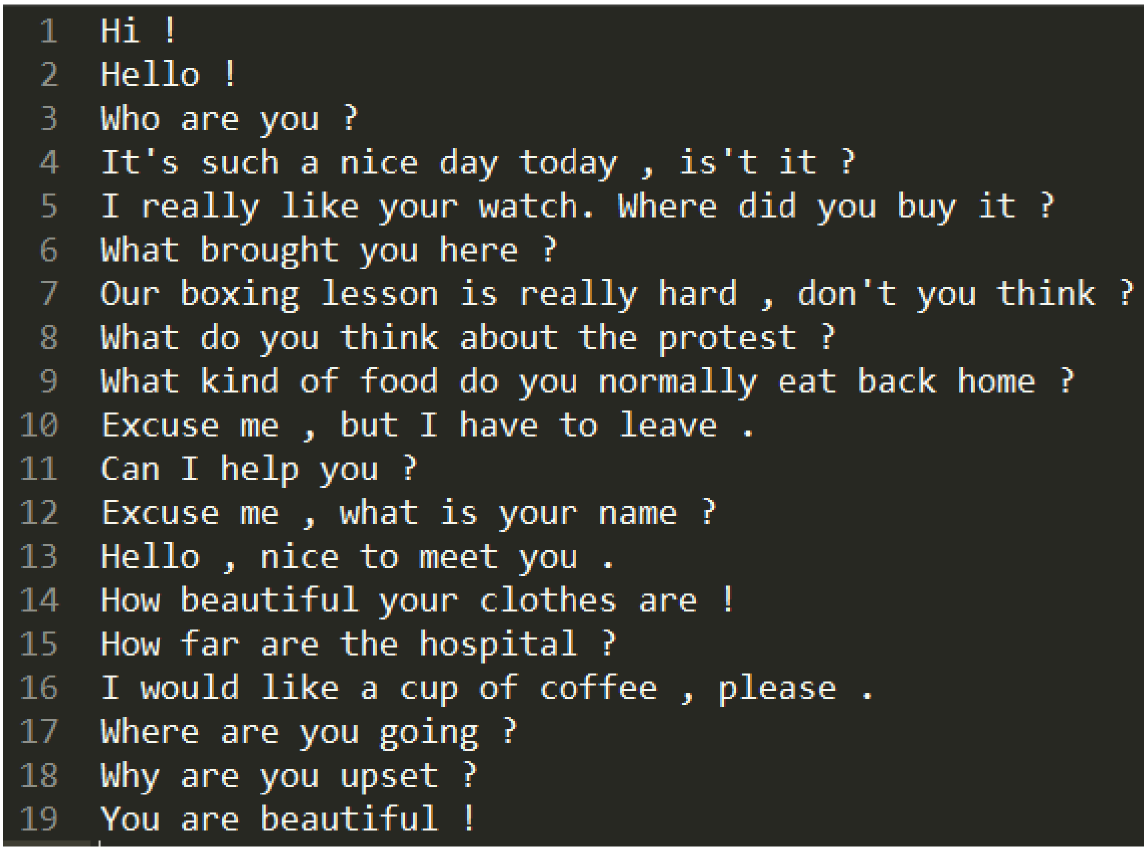
以下圖為固定的參數設定。



```
Seq2Seq Function     =   embedding_attention_seq2seq
Cost Function        =   sampled_softmax_loss        Sample num      =   512
Cell                 =   BasicLSTMCell               Cell Size       =   256
RNN Layers           =   1                           Learning Rate   =   Adam(0.001)
```

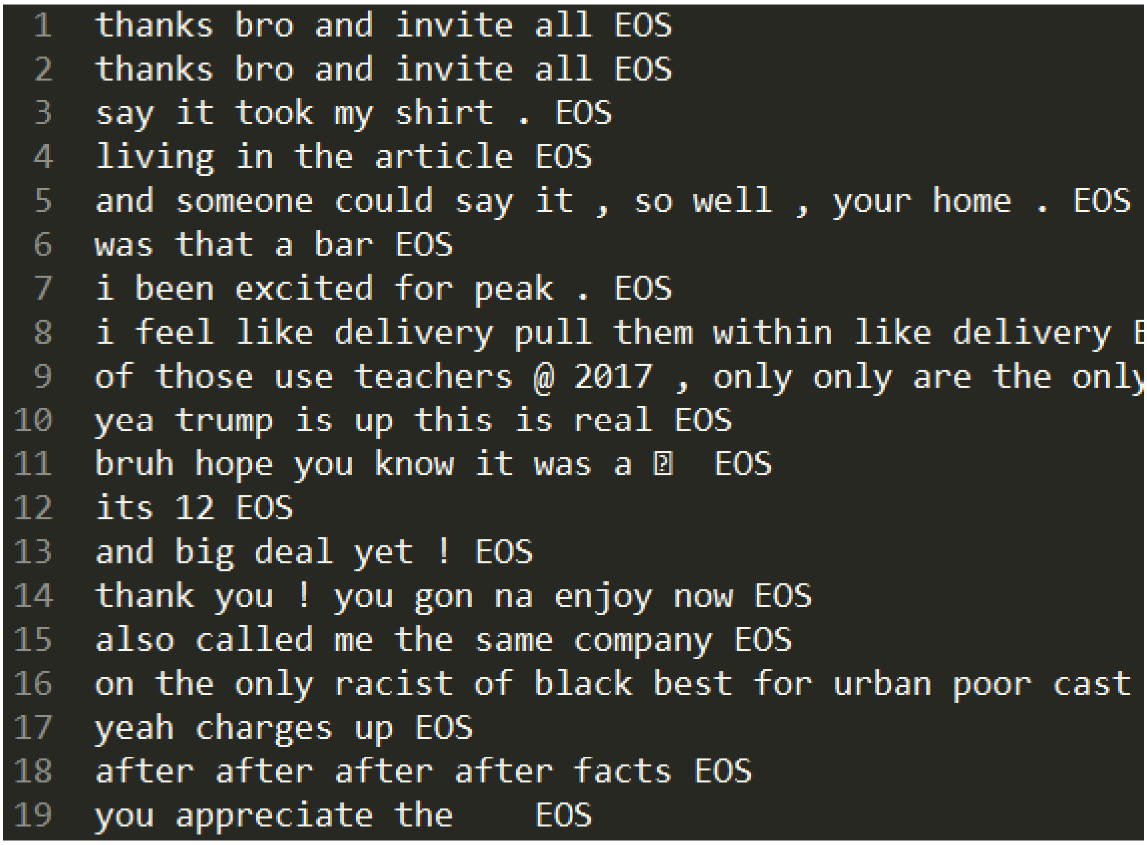
(2) Performance

接下來我們分別比較倒著 Input，跟照順序的差別，以下為 Input 句子。



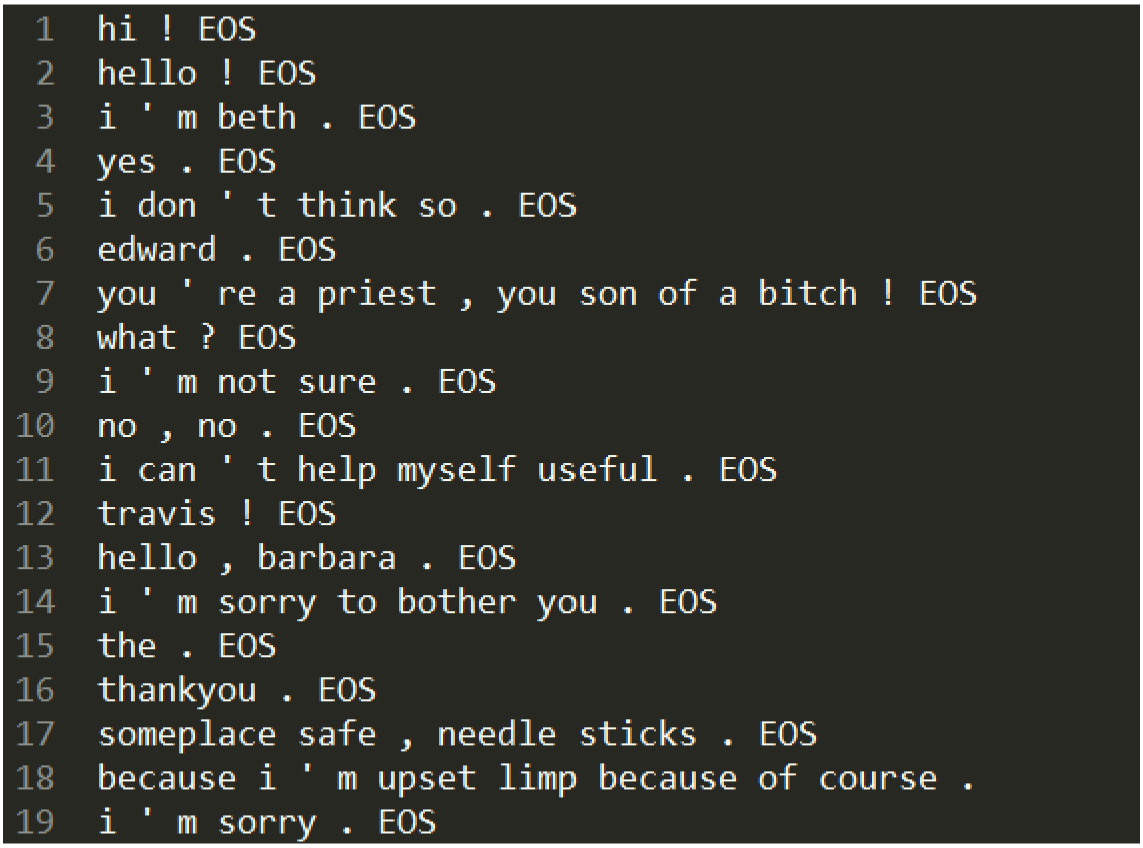
```
Hi !
Hello !
Who are you ?
It's such a nice day today , is't it ?
I really like your watch. Where did you buy it ?
What brought you here ?
Our boxing lesson is really hard , don't you think ?
What do you think about the protest ?
What kind of food do you normally eat back home ?
Excuse me , but I have to leave .
Can I help you ?
Excuse me , what is your name ?
Hello , nice to meet you .
How beautiful your clothes are !
How far are the hospital ?
I would like a cup of coffee , please .
Where are you going ?
Why are you upset ?
You are beautiful !
```

接著下圖是 Reverse 的結果。



```
thanks bro and invite all EOS
thanks bro and invite all EOS
say it took my shirt . EOS
living in the article EOS
and someone could say it , so well , your home . EOS
was that a bar EOS
i been excited for peak . EOS
i feel like delivery pull them within like delivery E
of those use teachers @ 2017 , only only are the only
yea trump is up this is real EOS
bruh hope you know it was a □  EOS
its 12 EOS
and big deal yet ! EOS
thank you ! you gon na enjoy now EOS
also called me the same company EOS
on the only racist of black best for urban poor cast
yeah charges up EOS
after after after after facts EOS
you appreciate the    EOS
```

最後則是照順序的回應結果。



```
hi ! EOS
hello ! EOS
i ' m beth . EOS
yes . EOS
i don ' t think so . EOS
edward . EOS
you ' re a priest , you son of a bitch ! EOS
what ? EOS
i ' m not sure . EOS
no , no . EOS
i can ' t help myself useful . EOS
travis ! EOS
hello , barbara . EOS
i ' m sorry to bother you . EOS
the . EOS
thankyou . EOS
someplace safe , needle sticks . EOS
because i ' m upset limp because of course .
i ' m sorry . EOS
```

可以發現一個很有趣的結果，Reverse過後雖然有點文不達意，但是卻能夠生出較長的句子，而照著原始順序，雖然看起來稍微切中回應，但是通常都處於回應短短的句子。我們猜測如果給予Reverse更多的訓練資料，以及訓練時間，或許能夠有更好的成果。

Team Division

R04922108	R04922098	R04922086	R04922065
S2S RL	S2S	Report	Report