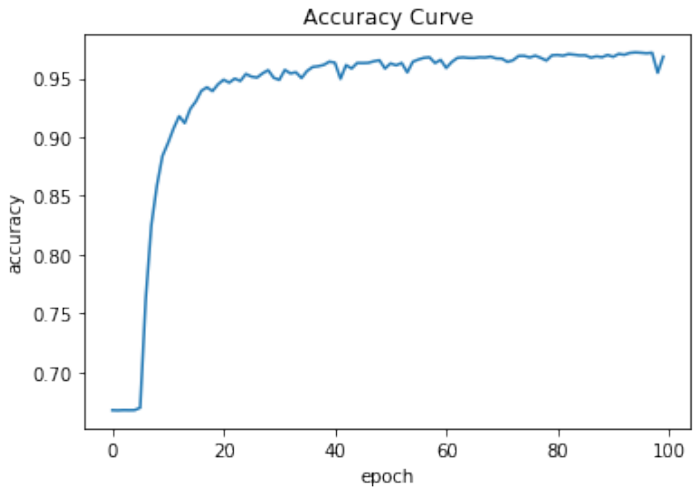
HW5 Report

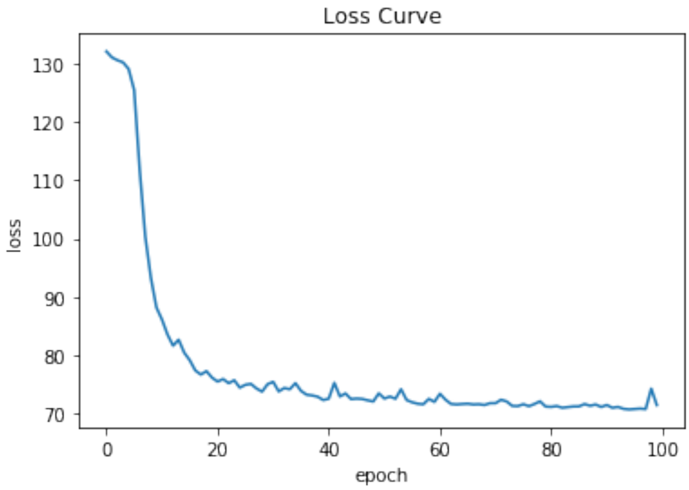
學號：B05902017 系級： 資工四 姓名：王盛立

1 請說明你實作之 RNN 模型架構及使用的 word embedding 方法，回報模型的正確率並繪出訓練曲線

```
class LSTM_Net(nn.Module):
    def __init__(self, embedding, embedding_dimension, hidden_dimension, layer_number,
dropout = 0.3):
        super(LSTM_Net, self).__init__()

        self.embedding = torch.nn.Embedding(embedding.size(0),embedding.size(1))
        self.embedding.weight = torch.nn.Parameter(embedding)

        self.lstm = nn.GRU(input_size=embedding.size(1),
hidden_size=hidden_dimension,num_layers=layer_number,dropout=dropout,batch_first=True)
        self.lstm2 = nn.GRU(input_size=100,hidden_size=100,num_layers=layer_number,
                            dropout=dropout,batch_first=True)

        self.classifier = nn.Sequential(
            nn.Linear(100 * 116 ,2048),
            nn.ReLU(),
            nn.Dropout(0.5),
            nn.Linear(2048, 256),
            nn.ReLU(),
            nn.Dropout(0.5),
            nn.Linear(256,32),
            nn.ReLU(),
            nn.Dropout(0.5),
            nn.Linear(32, 2),
            nn.Softmax(dim = 1)
        )
    def forward(self,inputs):
        inputs = self.embedding(inputs)

        x3, (_,_) = self.lstm(inputs)
        x2, (_,_) = self.lstm2(x3)
        x, (_,_) = self.lstm2(x2)

        x = x.reshape((-1,100 * 116))

        x = self.classifier(x)
        return x
```

我的RNN模型如上，我用了三層GRU加上Linear的部分，Linear每一層後面接著ReLu跟Dropout。在training上的準確率如下圖，不過如果有切validation一樣會看到在validation上的準確率是很低的大概只有六成。另外如果跟下一題的DNN做比較，可以發現他的訓練曲線是比較有波動的，不像DNN很簡單的就收斂到一個滿固定的地方就不太變了。而我使用的word embedding的方法就是先用word2vec的model當成初始化取兩百維，之後丟進embedding層。



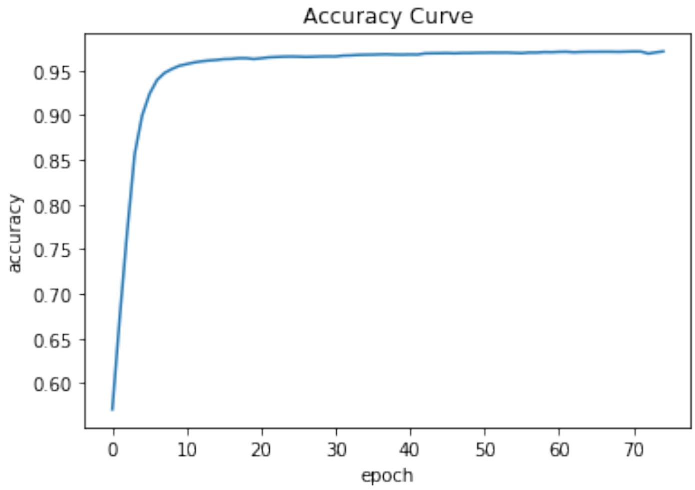


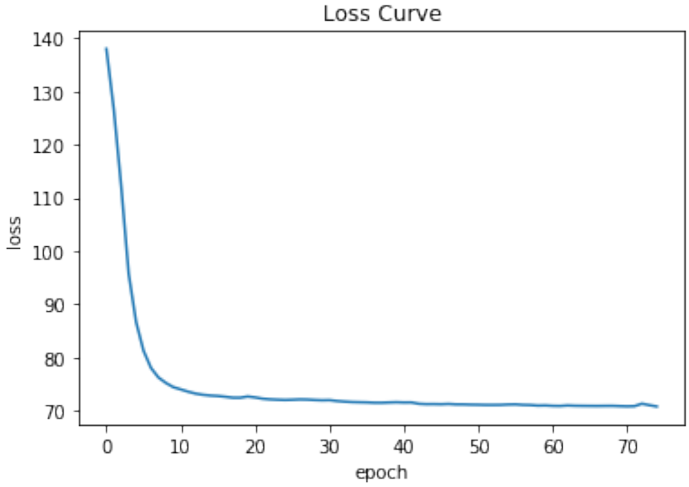
	Public	Private
RNN	0.77209	0.77674

2 請實作 BOW+DNN 模型，敘述你的模型架構，回報模型的正確率並繪出訓練曲線

```
class DNN(nn.Module):
    def __init__(self):
        super(DNN, self).__init__()
        self.dnn = nn.Sequential(
            nn.Linear(4880,1024),
            nn.ReLU(),
            nn.Dropout(0.5),
            nn.Linear(1024,256),
            nn.ReLU(),
            nn.Dropout(0.5),
            nn.Linear(256,32),
            nn.ReLU(),
            nn.Dropout(0.5),
            nn.Linear(32,2),
            nn.Softmax(dim = 1)
        )
    def forward(self, inputs):
        inputs = inputs.view(inputs.size()[0], -1)
        out = self.dnn(inputs)
        return out
```

我的DNN模型如上，是很簡單的多層Linear結構，每一層後面接著ReLu跟Dropout。在training上的準確率如下圖，不過如果有切validation會看到在validation上的準確率是很低的大概只有六成。





	Public	Private
BOW+DNN	0.78837	0.77906

3 請敘述你如何 improve performance（preprocess, embedding, 架構等），並解釋為何這些做法可以使模型進步

對於這次的資料我們可以先做一些preprocess的處理，例如把大小寫都先弄成小寫。避免其實是同樣的字但由於他出現在第一個而變成大寫讓我們的model以為他們是不同的。另外在丟進RNN時，要先把每一句話都弄成同樣的長度，但是在資料中其實每一句話的長度都是不一樣的，那我們就可以做一些補齊的部分讓他們成為同樣的程度。但是如果有句子特別長，就會導致其他資料都補了太多，所以我決定把過長的資料就拿掉，讓大部分的資料都是補齊到一個合理的長度。

4 請比較不做斷詞 (e.g.,用空白分開) 與有做斷詞，兩種方法實作出來的效果差異，並解釋為何有此差別

我有使用套件做斷詞跟不做斷詞只用空白分開，最後出來的結果其實沒有差很多。做斷詞只有稍微好一點點，我覺得是因為英文本來就是用空白分開，所以其實用套件斷詞做出來的成效不大。如果是中文的資料應該就會有比較顯著的差距因為中文的斷詞無法輕易的做判斷。

5 請比較 RNN 與 BOW 兩種不同 model 對於 "Today is hot, but I am happy."與"I am happy, but today is hot." 這兩句話的分數（model output），並討論造成差異的原因

句1: "Today is hot, but I am happy."

句2: "I am happy, but today is hot."

	正面的分數	負面的分數
RNN + 句1	1.00	3.2556e-30
RNN + 句2	1.00	6.2299e-26
BOW+ 句1	1.00	7.2606e-09
BOW+ 句2	1.00	8.2115e-10

可以看到兩種方法最後都能知道這兩句話都是正面的，但是對於BOW而言這兩句話是一樣的，所以可以看到他們預測出來的結果更加相近。而由於考慮了前後文，所以RNN其實是比較知道整句話的意思的，BOW可能看到hot還會覺得有一點點負面的意思。同樣的硬要說的話第一句話是更加正面的，第二句話還有一點點的攻擊性，對於BOW而言兩者沒什麼差，但對RNN來說就可以反映這件事情。

6 Refer to math problem

1 LSTM Cell

t = 1:

$z = wx^1 + b = (0,0,0,1) \cdot (0,1,0,3) = 3$

$z_i = w_i x^1 + b_i = (100,100,0,0) \cdot (0,1,0,3) - 10 = 90$

$z_f = w_f x^1 + b_f = (-100,-100,0,0) \cdot (0,1,0,3) + 110 = 10$

$z_o = w_o x^2 + b_o = (0,0,100,0) \cdot (0,1,0,3) - 10 = -10$

$f(z) = \frac{1}{1+e^{-z}}$

$c' = f(z_i)g(z) + cf(z_f) = \frac{1}{1+e^{-90}} \cdot 3 + 0 \cdot \frac{1}{1+e^{-10}} \approx 3$

$y = f(z_o)\ h(c') = \frac{1}{1+e^{-(-10)}} \cdot 3 \approx 0$

t = 2:

$z = wx^2 + b = (0,0,0,1) \cdot (1,0,1,-2) = -2$

$z_i = w_i x^2 + b_i = (100,100,0,0) \cdot (1,0,1,-2) - 10 = 90$

$z_f = w_f x^2 + b_f = (-100,-100,0,0) \cdot (1,0,1,-2) + 110 = 10$

$z_o = w_o x^2 + b_o = (0,0,100,0) \cdot (1,0,1,-2) - 10 = 90$

$f(z) = \frac{1}{1+e^{-z}}$

$c' = f(z_i)g(z) + cf(z_f) = \frac{1}{1+e^{-(90)}} \cdot -2 + 3 \cdot \frac{1}{1+e^{-(10)}} \approx 1$

$y = f(z_o)\ h(c') = \frac{1}{1+e^{-(90)}} \cdot 1 \approx 1$

t = 3:

$z = wx^3 + b = (0,0,0,1) \cdot (1,1,1,4) = 4$

$z_i = w_i x^3 + b_i = (100,100,0,0) \cdot (1,1,1,4) - 10 = 190$

$z_f = w_f x^3 + b_f = (-100, -100, 0, 0) \cdot (1, 1, 1, 4) + 110 = -90$

$z_o = w_o x^3 + b_o = (0, 0, 100, 0) \cdot (1, 1, 1, 4) - 10 = 90$

$f(z) = \frac{1}{1+e^{-z}}$

$c' = f(z_i) g(z) + c f(z_f) = \frac{1}{1+e^{-(190)}} \cdot 4 + 1 \cdot \frac{1}{1+e^{-(-90)}} \approx 4$

$y = f(z_o)\ h(c') = \frac{1}{1+e^{-(90)}} \cdot 4 \approx 4$

t = 4:

$z = w x^4 + b = (0, 0, 0, 1) \cdot (0, 1, 1, 0) = 0$

$z_i = w_i x^4 + b_i = (100, 100, 0, 0) \cdot (0, 1, 1, 0) - 10 = 90$

$z_f = w_f x^4 + b_f = (-100, -100, 0, 0) \cdot (0, 1, 1, 0) + 110 = 10$

$z_o = w_o x^4 + b_o = (0, 0, 100, 0) \cdot (0, 1, 1, 0) - 10 = 90$

$f(z) = \frac{1}{1+e^{-z}}$

$c' = f(z_i) g(z) + c f(z_f) = \frac{1}{1+e^{-(90)}} \cdot 0 + 4 \cdot \frac{1}{1+e^{-(10)}} \approx 4$

$y = f(z_o)\ h(c') = \frac{1}{1+e^{-(90)}} \cdot 4 \approx 4$

t = 5:

$z = w x^5 + b = (0, 0, 0, 1) \cdot (0, 1, 0, 2) = 2$

$z_i = w_i x^5 + b_i = (100, 100, 0, 0) \cdot (0, 1, 0, 2) - 10 = 90$

$z_f = w_f x^5 + b_f = (-100, -100, 0, 0) \cdot (0, 1, 0, 2) + 110 = 10$

$z_o = w_o x^5 + b_o = (0, 0, 100, 0) \cdot (0, 1, 0, 2) - 10 = -10$

$f(z) = \frac{1}{1+e^{-z}}$

$c' = f(z_i) g(z) + c f(z_f) = \frac{1}{1+e^{-(90)}} \cdot 2 + 4 \cdot \frac{1}{1+e^{-(10)}} \approx 6$

$y = f(z_o)\ h(c') = \frac{1}{1+e^{-(-10)}} \cdot 6 \approx 0$

t = 6:

$z = w x^6 + b = (0, 0, 0, 1) \cdot (0, 0, 1, -4) = -4$

$z_i = w_i x^6 + b_i = (100, 100, 0, 0) \cdot (0, 0, 1, -4) - 10 = -10$

$z_f = w_f x^6 + b_f = (-100, -100, 0, 0) \cdot (0, 0, 1, -4) + 110 = 110$

$z_o = w_o x^6 + b_o = (0, 0, 100, 0) \cdot (0, 0, 1, -4) - 10 = 90$

$f(z) = \frac{1}{1+e^{-z}}$

$c' = f(z_i) g(z) + c f(z_f) = \frac{1}{1+e^{-(-10)}} \cdot -4 + 6 \cdot \frac{1}{1+e^{-(110)}} \approx 6$

$y = f(z_o)\ h(c') = \frac{1}{1+e^{-(90)}} \cdot 6 \approx 6$

t = 7:

$z = w x^7 + b = (0, 0, 0, 1) \cdot (1, 1, 1, 1) = 1$

$z_i = w_i x^7 + b_i = (100, 100, 0, 0) \cdot (1, 1, 1, 1) - 10 = 190$

$z_f = w_f x^7 + b_f = (-100, -100, 0, 0) \cdot (1, 1, 1, 1) + 110 = -90$

$z_o = w_o x^7 + b_o = (0, 0, 100, 0) \cdot (1, 1, 1, 1) - 10 = 90$

$f(z) = \frac{1}{1+e^{-z}}$

$c' = f(z_i) g(z) + c f(z_f) = \frac{1}{1+e^{-(190)}} \cdot 1 + 6 \cdot \frac{1}{1+e^{-(-90)}} \approx 1$

$y = f(z_o)\ h(c') = \frac{1}{1+e^{-(90)}} \cdot 1 \approx 1$

t = 8:

$z = w x^8 + b = (0, 0, 0, 1) \cdot (1, 0, 1, 2) = 2$

$z_i = w_i x^8 + b_i = (100, 100, 0, 0) \cdot (1, 0, 1, 2) - 10 = 90$

$z_f = w_f x^8 + b_f = (-100, -100, 0, 0) \cdot (1, 0, 1, 2) + 110 = 10$

$z_o = w_o x^8 + b_o = (0, 0, 100, 0) \cdot (1, 0, 1, 2) - 10 = 90$

$f(z) = \frac{1}{1+e^{-z}}$

$c' = f(z_i) g(z) + c f(z_f) = \frac{1}{1+e^{-(90)}} \cdot 2 + 1 \cdot \frac{1}{1+e^{-(10)}} \approx 3$

$y = f(z_o)\ h(c') = \frac{1}{1+e^{-(90)}} \cdot 3 \approx 3$

The output sequence$(y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8) = (0, 1, 4, 4, 0, 6, 1, 3)$

2 Word Embedding

$$h = w^T x = \begin{bmatrix} w_{11} & w_{21} & \cdots & w_{v1} \\ w_{12} & w_{22} & \cdots & w_{v2} \\ & \vdots & & \\ w_{1n} & w_{2n} & \cdots & w_{vn} \end{bmatrix} \begin{bmatrix} x1 \\ x2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{v} w_{i1} x_i \\ \sum_{i=1}^{v} w_{i2} x_i \\ \vdots \\ \sum_{i=1}^{v} w_{in} x_i \end{bmatrix}$$

$$u = w'^T h = \begin{bmatrix} w'_{11} & w'_{21} & \cdots & w'_{n1} \\ w'_{12} & w'_{22} & \cdots & w'_{n2} \\ & \vdots & & \\ w'_{1v} & w'_{2v} & \cdots & w'_{nv} \end{bmatrix} \begin{bmatrix} \sum_{i=1}^{v} w_{i1} x_i \\ \sum_{i=1}^{v} w_{i2} x_i \\ \vdots \\ \sum_{i=1}^{v} w_{in} x_i \end{bmatrix} = \begin{bmatrix} \sum_{q=1}^{N} w'_{q1} \sum_{i=1}^{v} w_{iq} x_i \\ \sum_{q=1}^{N} w'_{q2} \sum_{i=1}^{v} w_{iq} x_i \\ \vdots \\ \sum_{q=1}^{N} w'_{qv} \sum_{i=1}^{v} w_{iq} x_i \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{v} \sum_{q=1}^{N} w_{iq} w'_{q1} x_i \\ sum_{i=1}^{v} \sum_{q=1}^{N} w_{iq} w'_{q2} x_i \\ \vdots \\ \sum_{i=1}^{v} \sum_{q=1}^{N} w_{iq} w'_{qv} x_i \end{bmatrix}$$

$$Loss = -log \prod_{c \in C} \frac{exp(u_c)}{\sum_{i \in V} exp(u_i)} = -\sum_{c \in C} log \frac{exp(u_c)}{\sum_{i \in V} exp(u_i)} = -\sum_{c \in C} log(exp(u_c)) - log(\sum_{i \in V} exp(u_i))$$

$$= -\sum_{c \in C} log(u_c - log(\sum_{i \in V} exp(u_i))) = -\sum_{c \in C} (\sum_{p=1}^{v} \sum_{q=1}^{N} w_{pq} w'_{qc} x_p - log(\sum_{i \in V} exp(\sum_{p=1}^{v} \sum_{q=1}^{N} w_{pq} w'_{qi} x_p)))$$

$$\frac{\partial L}{\partial W_{ij}^{T}} = \frac{\partial L}{\partial W_{ji}} = -\sum_{c \in C} (w'_{ic} x_j - \frac{\sum_{r \in V} w'_{ir} x_j exp(\sum_{p=1}^{v} \sum_{q=1}^{N} w_{pq} w'_{qr} x_p)}{\sum_{r \in V} exp(\sum_{p=1}^{v} \sum_{q=1}^{N} w_{pq} w'_{qr} x_p)})$$

$$\frac{\partial L}{\partial W_{ij}^{T'}} = \frac{\partial L}{\partial W'_{ji}} = [i \in c](-\sum_{p=1}^{v} w_{pj} x_p) + \sum_{c \in C} \frac{(\sum_{p=1}^{v} w_{pj} x_p) exp(\sum_{p=1}^{v} \sum_{q=1}^{N} w_{pq} w'_{qi} x_p)}{\sum_{r \in v} (\sum_{p=1}^{v} \sum_{q=1}^{N} w_{pq} w'_{qr} x_p)}$$