

Lab 5. MẠNG NEURAL HỒI QUY (RNN)

MSSV: 20521196

Họ tên: Nguyễn Mạnh Đức

Lớp: DS201.N11.1

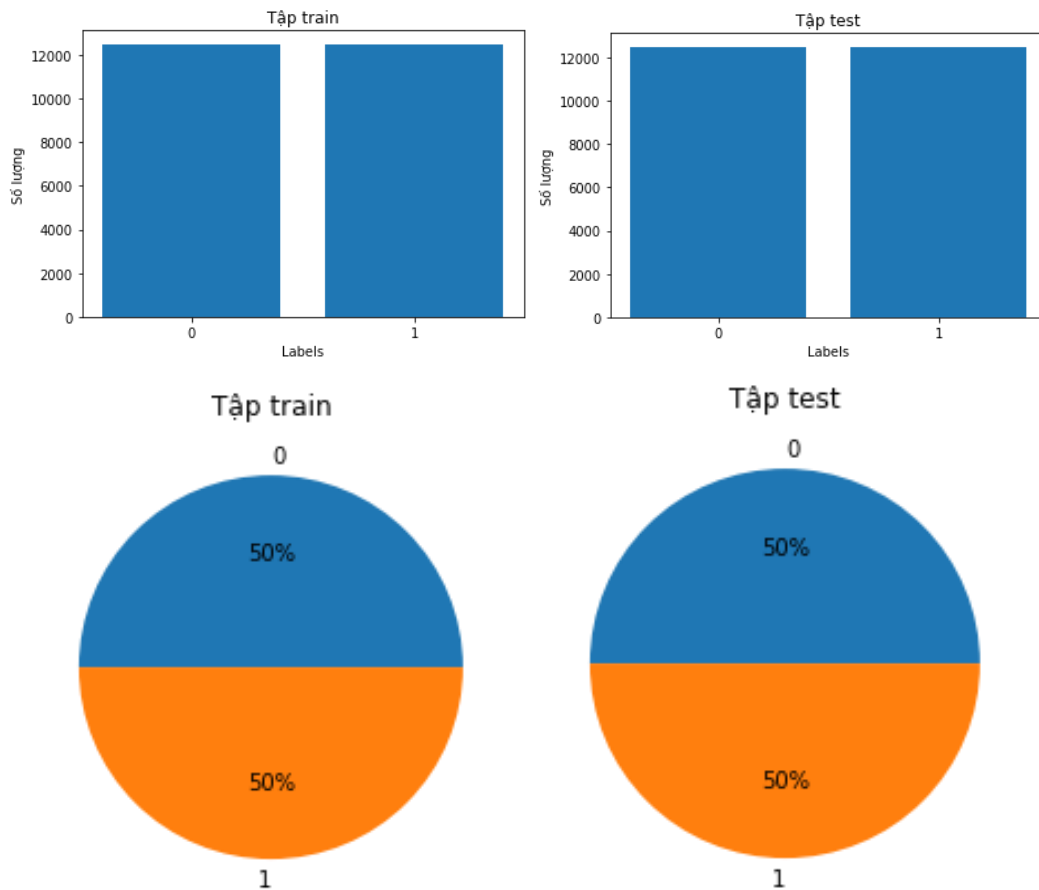
1. BỘ DỮ LIỆU

(?) Cho biết số lượng phần tử của tập train và tập test?

```
1 print("Tập train: " + str(len(training_targets)))  
2 print("Tập test: " + str(len(testing_targets)))
```

↳ Tập train: 25000
Tập test: 25000

(?) Vẽ biểu đồ cột thể hiện số lượng nhãn của mỗi tập dữ liệu và biểu đồ tròn thể hiện tỉ lệ phân bố nhãn của mỗi tập dữ liệu



(?) In ra 5 câu đầu tiên trong tập train và tập test kèm nhãn tương ứng?

```
Tập train: -----
train[0] : [1, 14, 22, 16, 43, 530, 973, 1622, 1385, 65, 458, 4468, 66, 3941, 4, 173, 36, 256, 5, 25, 100, 43, 838, 112, 50, 670, 2, 9, 35, 480,
=> label: 1

train[1] : [1, 194, 1153, 194, 8255, 78, 228, 5, 6, 1463, 4369, 5012, 134, 26, 4, 715, 8, 118, 1634, 14, 394, 20, 13, 119, 954, 189, 102, 5, 207,
=> label: 0

train[2] : [1, 14, 47, 8, 30, 31, 7, 4, 249, 108, 7, 4, 5974, 54, 61, 369, 13, 71, 149, 14, 22, 112, 4, 2401, 311, 12, 16, 3711, 33, 75, 43, 1825,
=> label: 0

train[3] : [1, 4, 2, 2, 33, 2804, 4, 2040, 432, 111, 153, 103, 4, 1494, 13, 70, 131, 67, 11, 61, 2, 744, 35, 3715, 761, 61, 5766, 452, 9214, 4, 5,
=> label: 1

train[4] : [1, 249, 1323, 7, 61, 113, 10, 10, 13, 1637, 14, 20, 56, 33, 2401, 18, 457, 88, 13, 2626, 1400, 45, 3171, 13, 70, 79, 49, 706, 919, 1,
=> label: 0

Tập test : -----
test[0] : [1, 591, 202, 14, 31, 6, 717, 10, 10, 2, 2, 5, 4, 360, 7, 4, 177, 5760, 394, 354, 4, 123, 9, 1035, 1035, 1035, 10, 10, 13, 92, 124, 89,
=> label: 0

test[1] : [1, 14, 22, 3443, 6, 176, 7, 5063, 88, 12, 2679, 23, 1310, 5, 109, 943, 4, 114, 9, 55, 606, 5, 111, 7, 4, 139, 193, 273, 23, 4, 172, 27,
=> label: 1

test[2] : [1, 111, 748, 4368, 1133, 2, 2, 4, 87, 1551, 1262, 7, 31, 318, 9459, 7, 4, 498, 5076, 748, 63, 29, 5161, 220, 686, 2, 5, 17, 12, 575, 2,
=> label: 1

test[3] : [1, 13, 1228, 119, 14, 552, 7, 20, 190, 14, 58, 13, 258, 546, 1786, 8, 1968, 4, 268, 237, 13, 191, 81, 15, 13, 80, 43, 3824, 44, 12, 14,
=> label: 0

test[4] : [1, 40, 49, 85, 84, 1040, 146, 6, 783, 254, 4386, 337, 5, 13, 447, 14, 500, 10, 10, 14, 500, 517, 1076, 357, 21, 1684, 72, 45, 290, 12,
=> label: 1
```

(?) Cho biết 5 từ đầu tiên của tập từ vựng?

```
['fawn', 'tsukino', 'nunnery', 'sonja', 'vani']
```

(?) Cho biết tập từ vựng của bộ dữ liệu có bao nhiêu từ?

Tập từ vựng có 88584 từ

(?) Cho biết độ dài của câu văn vừa được decode?

Độ dài câu văn vừa được decode: 218

(?) Decode 5 câu tiếp theo trong tập train?

```
the thought solid thought senator do making to is spot nomination assumed while he of jack in where picked as getting on was did hands fact chara
the as there in at by br of sure many br of proving no only women was than doesn't as you never of hat night that with ignored they bad out super
the of and and they halfway of identity went plot actors watch of share was well these can this only and ten so failing feels only novak killer t
the sure themes br only acting i i was favourite as on she they hat but already most was scares minor if flash was well also good 8 older was wit
the effort still been that usually makes for of finished sucking ended and an because before if just though something know novel female i i slowl
```

2. XÂY DỰNG MÔ HÌNH

(?) Compile mô hình với các thông số sau:

- Hàm tối ưu: Adam với learning_rate = 0.01.
- Hàm loss: Binary Cross-Entropy.
- Độ đo: Accuracy.

```
[ ] 1 from tensorflow.keras.optimizers import Adam
    2 from tensorflow.keras.losses import CategoricalCrossentropy
    3 optimizer = Adam(learning_rate=0.01)
    4 loss = CategoricalCrossentropy()
    5 model.compile(optimizer=optimizer, loss=loss, metrics=['accuracy'])
```

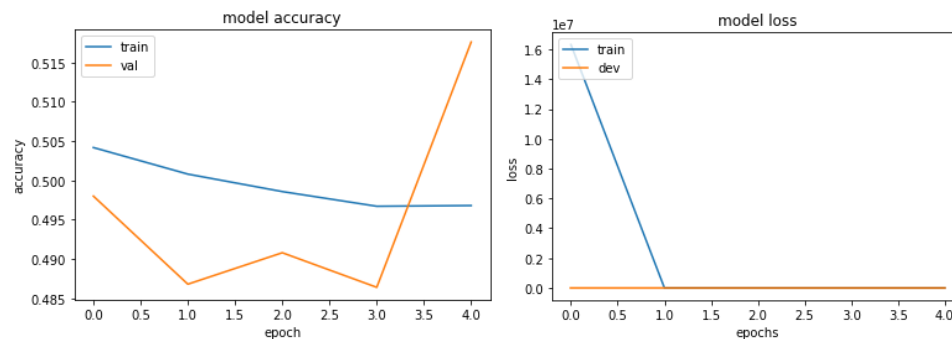
(?) Sử dụng hàm summary để xem cấu trúc mô hình đã xây dựng?

Model: "sequential"

Layer (type)	Output Shape	Param #
simple_rnn (SimpleRNN)	(None, None, 200)	40400
simple_rnn_1 (SimpleRNN)	(None, 200)	80200
dense (Dense)	(None, 2)	402

=====
Total params: 121,002
Trainable params: 121,002
Non-trainable params: 0
=====

(?) Vẽ đồ thị học với Accuracy và Loss?



(?) Tính độ chính xác đánh giá của mô hình bằng độ đo Accuracy?

```
1 from sklearn.metrics import accuracy_score
2 accuracy_score(y_test, y_pred_label)*100
```

50.036

3. LSTM, BILSTM, EMBEDDING

Lớp LSTM

(?) Compile mô hình với các thông số sau:

- Hàm tối ưu: Adam với learning_rate = 0.01.
- Hàm loss: Binary Cross-Entropy.
- Độ đo: Accuracy.

```
1 optimizer = Adam(learning_rate=0.01)
2 loss = CategoricalCrossentropy()
3 model1.compile(optimizer=optimizer, loss=loss, metrics=['accuracy'])
```

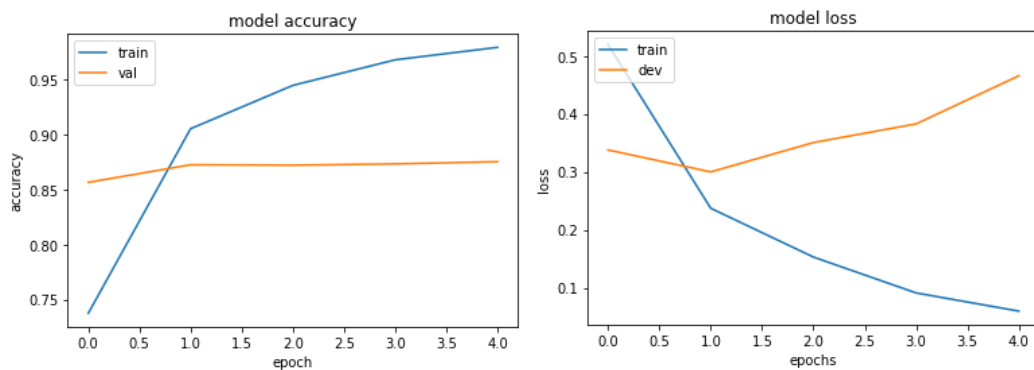
(?) Sử dụng hàm summary để xem cấu trúc mô hình đã xây dựng?

. Model: "sequential_1"

Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, None, 128)	11338752
lstm (LSTM)	(None, 200)	263200
dense_1 (Dense)	(None, 2)	402

=====
Total params: 11,602,354
Trainable params: 11,602,354
Non-trainable params: 0
=====

(?) Vẽ đồ thị học với Accuracy và Loss?



(?) Đánh giá độ chính xác mô hình bằng độ đo Accuracy?

782/782 [=====] - 4s 5ms/step
86.084

Lớp Bidirectional

(?) Compile mô hình với các thông số sau:

- Hàm tối ưu: Adam với learning_rate = 0.01.
- Hàm loss: Binary Cross-Entropy.
- Độ đo: Accuracy.

```
1 optimizer = Adam(learning_rate = 0.01)
2 loss = BinaryCrossentropy()
3 model2.compile(optimizer = optimizer, loss = loss, metrics = ['accuracy'])
```

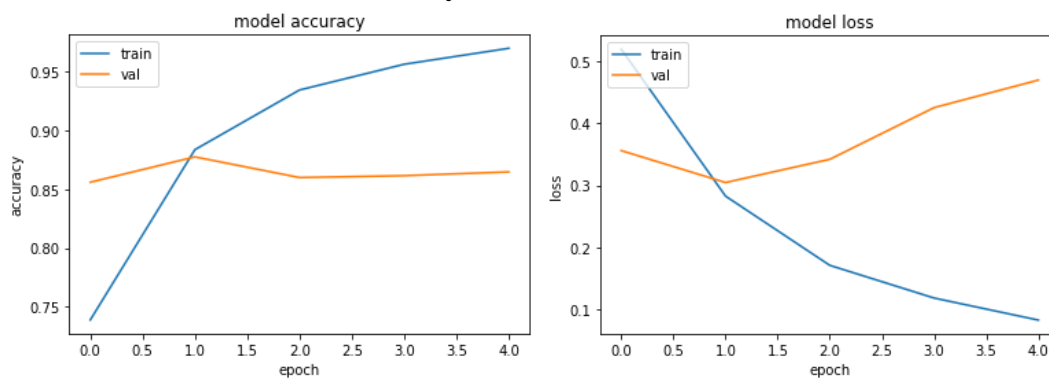
(?) Sử dụng hàm summary để xem cấu trúc mô hình đã xây dựng?

Model: "sequential_2"

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, None, 128)	11338752
bidirectional (Bidirectional)	(None, 400)	526400
dense_2 (Dense)	(None, 2)	802

=====
Total params: 11,865,954
Trainable params: 11,865,954
Non-trainable params: 0
=====

(?) Vẽ đồ thị học với Accuracy và Loss?



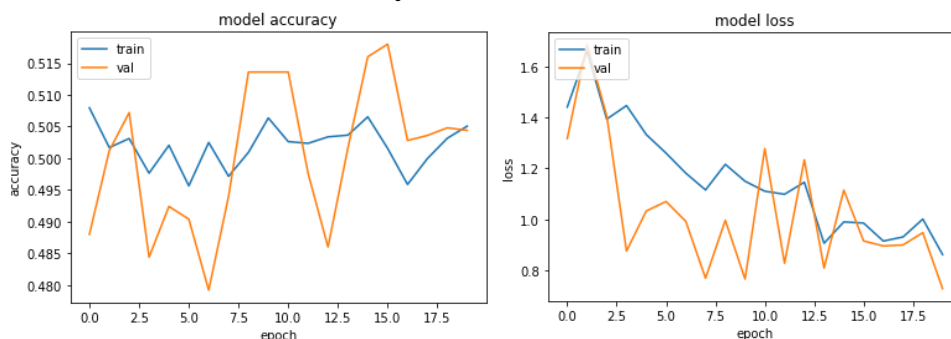
(?) Đánh giá độ chính xác mô hình bằng độ đo Accuracy?

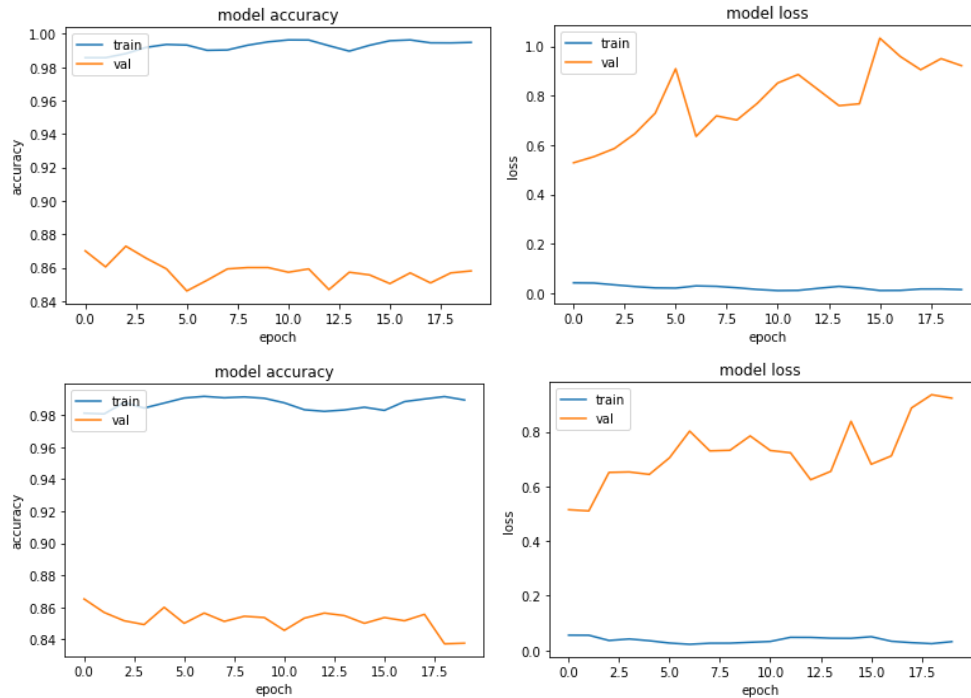
782/782 [=====] - 8s 9ms/step
86.188

BÀI TẬP

Bài 1

a. Vẽ đồ thị học với Accuracy và Loss.





b. Cho biết độ chính xác đánh giá của mô hình?

782/782 [=====] - 23s 29ms/step
49.756

782/782 [=====] - 4s 5ms/step
84.636

782/782 [=====] - 7s 9ms/step
83.14399999999999

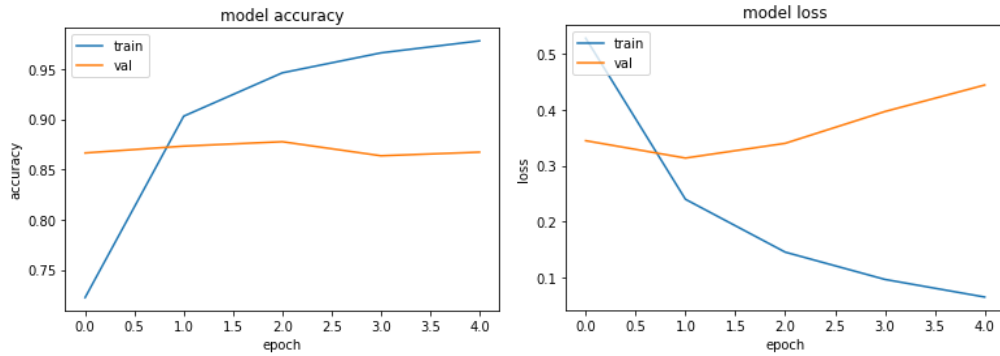
Bài 2

Thử thêm 1 lớp Simple RNN nữa vào mô hình và cho biết kết quả độ chính xác đánh giá của mô hình?

782/782 [=====] - 33s 42ms/step
49.996

Bài 3

a. Vẽ đồ thị học với Accuracy và Loss.

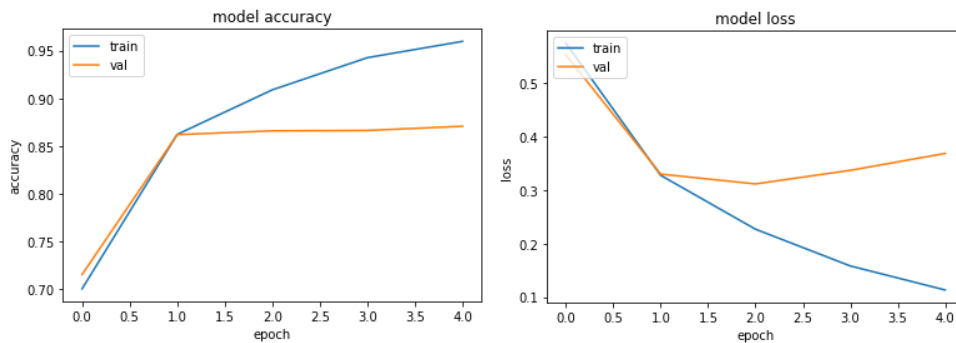


b. Cho biết độ chính xác đánh giá của mô hình?

782/782 [=====] - 4s 5ms/step
86.288

Bài 4

a. Vẽ đồ thị học với Accuracy và Loss.



b. Cho biết độ chính xác đánh giá của mô hình?

782/782 [=====] - 8s 9ms/step
86.60799999999999

Bài 5

a. Tìm hiểu GRU và xây dựng mô hình với lớp GRU (thực hiện mô hình tương tự Bài tập 3, nhưng thay lớp LSTM thành GRU).

Model: "sequential_6"

Layer (type)	Output Shape	Param #
embedding_4 (Embedding)	(None, None, 128)	11338752
gru (GRU)	(None, 200)	198000
dense_6 (Dense)	(None, 2)	402
Total params: 11,537,154		
Trainable params: 11,537,154		
Non-trainable params: 0		

b. So sánh độ chính xác đánh giá của 2 mô hình.

Sử dụng lớp GRU

782/782 [=====] - 4s 5ms/step
83.43199999999999

Sử dụng lớp LSTM

782/782 [=====] - 4s 5ms/step
86.288

Nhận xét: Sử dụng lớp LSTM trong model bài tập 3 có khả năng dự đoán tốt hơn so với sử dụng lớp GRU nhưng không đáng kể (~ 3%)