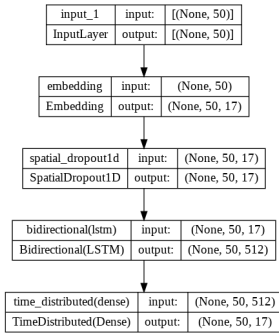
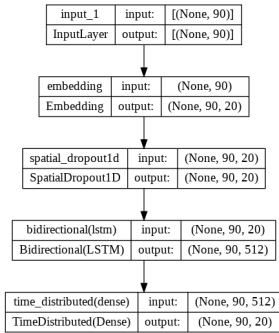
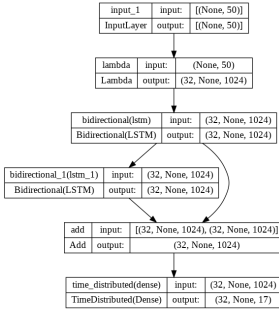


Tên: Trần Hoàng Anh  
MSSV: 20521079

Bài thực hành LAB2

## Tổng hợp

	Model	Visualize model structure	F1-macro
Câu 1	LSTM (biLSTM)		0.3465183
Câu 2	LSTM (biLSTM)		0.2455269
Câu 3 (bộ tiếng Anh)	ELMO(embedding) + LSTM (biLSTM)		0.5974857

Câu 3 (bộ tiếng Việt)	ELMO(embedding) + LSTM (biLSTM)		0.6607538
-----------------------	---------------------------------	--	-----------

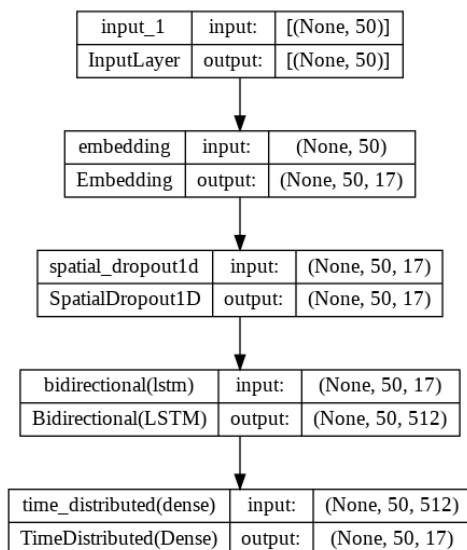
## Chi tiết

(\*) Chú thích:

- TP: True Positive
- FP: False Positive
- TN: True Negative
- FN: False Negative
- Precision =  $TP / (TP + FP)$
- Recall =  $TP / (TP + FN)$
- F1-score (micro) =  $2 * (Precision * Recall) / (Precision + Recall)$
- F1-score (macro) =  $Sum(F1\text{-score}) / number\_of\_samples$

Các độ đo trên được tính từ **confusion matrix**. Tuy nhiên, vì em chưa tìm được thư viện hay lệnh nào hỗ trợ cho loại **multilabel-multioutput** để tính trực tiếp được F1, nên em đã tự viết code để có thể tính được confusion matrix. Dựa vào confusion matrix của từng mẫu trong tập test, em cộng dồn toàn bộ confusion matrix để có thể tính được confusion matrix của cả bộ test. Từ đó có thể tính được độ đo F1.

### Câu 1:



-Train:

loss: 0.2161 - accuracy: 0.9478 - val\_loss: 0.1095 - val\_accuracy: 0.9688

-Test:

loss: 0.1130 - accuracy: 0.9679

O

TP = 446105.0, FP = 5754.0, TN = 26960.0, FN = 781.0  
Precision: 0.9872659391535855, Recall: 0.9982523507113671, **F1-score: 0.9927287495340724**

B-art

TP = 0.0, FP = 0.0, TN = 479518.0, FN = 82.0  
Precision: nan, Recall: 0.0, **F1-score: 0**

I-art

TP = 0.0, FP = 0.0 , TN = 479530.0, FN = 70.0

Precision: nan, Recall: 0.0, **F1-score: 0**

B-eve

TP = 0.0, FP = 0.0 , TN = 479539.0, FN = 61.0

Precision: nan, Recall: 0.0, **F1-score: 0**

I-eve

TP = 0.0, FP = 0.0 , TN = 479548.0, FN = 52.0

Precision: nan, Recall: 0.0, **F1-score: 0**

B-geo

TP = 6056.0, FP = 2631.0 , TN = 469438.0, FN = 1475.0

Precision: 0.6971336479797399, Recall: 0.8041428761120701, **F1-score: 0.74682451596991**

I-geo

TP = 662.0, FP = 296.0 , TN = 477888.0, FN = 754.0

Precision: 0.6910229645093946, Recall: 0.4675141242937853, **F1-score: 0.5577085088458299**

B-gpe

TP = 2517.0, FP = 182.0 , TN = 476271.0, FN = 630.0

Precision: 0.9325676176361616, Recall: 0.7998093422306959, **F1-score: 0.8611016079370509**

I-gpe

TP = 0.0, FP = 0.0 , TN = 479562.0, FN = 38.0

Precision: nan, Recall: 0.0, **F1-score: 0**

B-nat

TP = 0.0, FP = 0.0 , TN = 479562.0, FN = 38.0

Precision: nan, Recall: 0.0, **F1-score: 0**

I-nat

TP = 0.0, FP = 0.0 , TN = 479590.0, FN = 10.0

Precision: nan, Recall: 0.0, **F1-score: 0**

B-org

TP = 1714.0, FP = 2024.0 , TN = 473423.0, FN = 2439.0

Precision: 0.45853397538790797, Recall: 0.41271370093908016, **F1-score: 0.434418958306932**

I-org

TP = 1573.0, FP = 2675.0 , TN = 473372.0, FN = 1980.0

Precision: 0.3702919020715631, Recall: 0.44272445820433437, **F1-score: 0.40328163056018457**

B-per

TP = 1485.0, FP = 830.0, TN = 475357.0, FN = 1928.0

Precision: 0.6414686825053996, Recall: 0.4351010840902432, **F1-score: 0.5185055865921787**

I-per

TP = 1459.0, FP = 652.0, TN = 475455.0, FN = 2034.0

Precision: 0.6911416390336333, Recall: 0.4176925279129688, **F1-score: 0.520699500356888**

B-tim

TP = 2507.0, FP = 203.0, TN = 475128.0, FN = 1762.0

Precision: 0.9250922509225092, Recall: 0.5872569688451628, **F1-score: 0.718441037397908**

I-tim

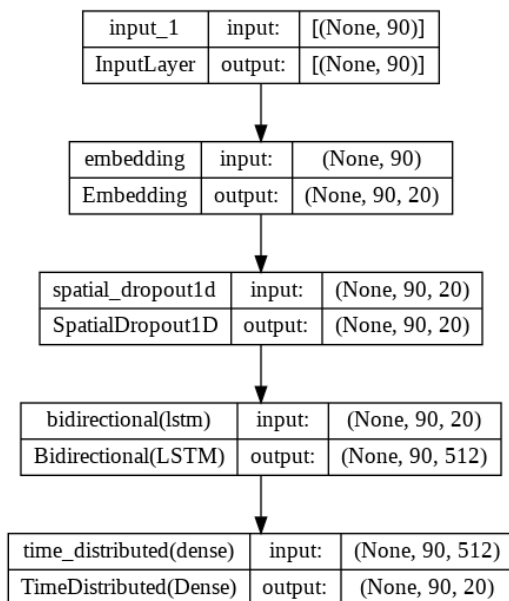
TP = 114.0, FP = 161.0, TN = 478051.0, FN = 1274.0

Precision: 0.4145454545454545, Recall: 0.08213256484149856, **F1-score: 0.13710162357185807**

**F1-macro: 0.34651833641604785**

**-Nhận xét:**

- + Tuy accuracy ở tập train và test được rất cao, nhưng F1-macro cho thấy được mô hình hoạt động không thật sự đạt được hiệu quả (0.3465).
- + Nguyên nhân do tag O được dự đoán đúng rất nhiều, vì trong bộ dữ liệu tag O chiếm phần lớn khiến cho mô hình dự đoán thiên về tag O, làm tăng accuracy của tag O, từ đó dẫn đến accuracy của mô hình cao, dễ gây nhầm lẫn khi đánh giá độ hiệu quả của mô hình.
- + Ngoài ra nguyên nhân cũng do quy trình embedding không mang ý nghĩa về mặt ngữ nghĩa, vì embedding layer của keras chỉ phân tách index là số nguyên thành vector nhiều chiều hơn chứa các số thập phân, dùng để fit với đầu vào của LSTM layer.
- + Phương pháp giải quyết vấn đề: Sử dụng các Language model, các embedding matrix hoặc các phương pháp embedding, ... giúp cho đầu vào mang được nhiều ý nghĩa hơn, từ đó giúp mô hình có thể học tốt hơn.



**Câu 2:**

-Train:

loss: 0.1919 - accuracy: 0.9460 - val\_loss: 0.1426 - val\_accuracy: 0.9597

-Test:

loss: 0.1448 - accuracy: 0.9591

O

TP = 167113.0, FP = 3404.0, TN = 9297.0, FN = 726.0

Precision: 0.9800371810435323, Recall: 0.9956744260869047,

**F1-score: 0.9877939211954273**

#### B-AGE

TP = 0.0, FP = 0.0 , TN = 180225.0, FN = 315.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### I-AGE

TP = 0.0, FP = 0.0 , TN = 180539.0, FN = 1.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### B-DATE

TP = 258.0, FP = 83.0 , TN = 179371.0, FN = 828.0

Precision: 0.7565982404692082, Recall: 0.23756906077348067, **F1-score: 0.36159775753328666**

#### I-DATE

TP = 768.0, FP = 419.0 , TN = 179121.0, FN = 232.0

Precision: 0.6470092670598147, Recall: 0.768, **F1-score: 0.7023319615912209**

#### B-GENDER

TP = 0.0, FP = 0.0 , TN = 180277.0, FN = 263.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### B-JOB

TP = 0.0, FP = 0.0 , TN = 180446.0, FN = 94.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### I-JOB

TP = 0.0, FP = 0.0 , TN = 180495.0, FN = 45.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### B-LOCATION

TP = 2011.0, FP = 1498.0 , TN = 176484.0, FN = 547.0

Precision: 0.573097748646338, Recall: 0.7861610633307271, **F1-score: 0.6629306082083402**

#### I-LOCATION

TP = 1921.0, FP = 821.0 , TN = 177093.0, FN = 705.0

Precision: 0.700583515681984, Recall: 0.7315308453922316, **F1-score: 0.7157228017883756**

#### B-NAME

TP = 0.0, FP = 0.0 , TN = 180361.0, FN = 179.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### I-NAME

TP = 0.0, FP = 0.0 , TN = 180533.0, FN = 7.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### B-ORGANIZATION

TP = 115.0, FP = 44.0 , TN = 180021.0, FN = 360.0

Precision: 0.7232704402515723, Recall: 0.24210526315789474, **F1-score: 0.362776025236593**

#### I-ORGANIZATION

TP = 400.0, FP = 221.0, TN = 179213.0, FN = 706.0

Precision: 0.644122383252818, Recall: 0.3616636528028933, **F1-score: 0.4632310364794441**

#### B-PATIENT\_ID

TP = 374.0, FP = 742.0, TN = 178501.0, FN = 923.0

Precision: 0.33512544802867383, Recall: 0.28835774865073244, **F1-score: 0.3099875673435557**

#### I-PATIENT\_ID

TP = 0.0, FP = 0.0, TN = 180527.0, FN = 13.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### B-SYMPTOM\_AND\_DISEASE

TP = 2.0, FP = 3.0, TN = 179872.0, FN = 663.0

Precision: 0.4, Recall: 0.0030075187969924814, **F1-score: 0.005970149253731344**

#### I-SYMPTOM\_AND\_DISEASE

TP = 197.0, FP = 146.0, TN = 179572.0, FN = 625.0

Precision: 0.5743440233236151, Recall: 0.2396593673965937, **F1-score: 0.3381974248927039**

#### B-TRANSPORTATION

TP = 0.0, FP = 0.0, TN = 180424.0, FN = 116.0

Precision: nan, Recall: 0.0, **F1-score: 0**

#### I-TRANSPORTATION

TP = 0.0, FP = 0.0, TN = 180507.0, FN = 33.0

Precision: nan, Recall: 0.0, **F1-score: 0**

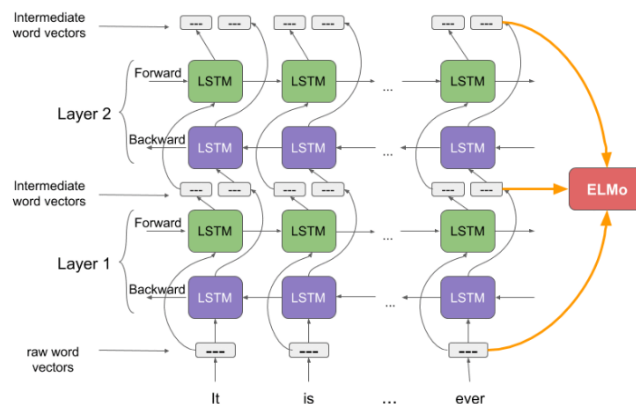
**F1-macro: 0.24552696267613397**

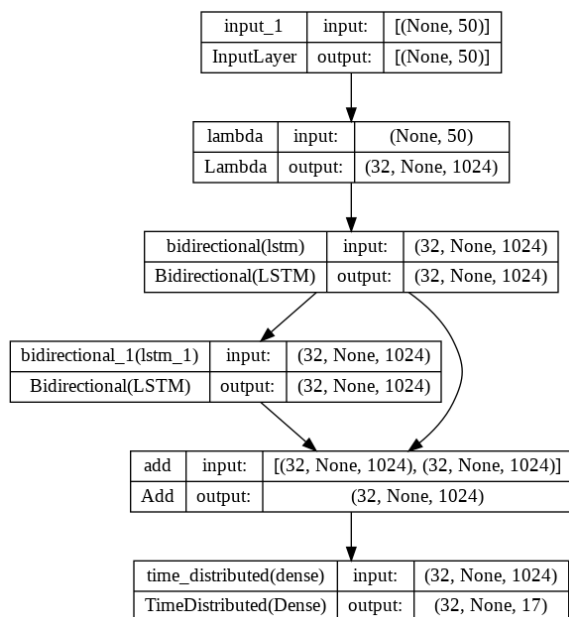
-Nhận xét: Tương tự như bài 1

### Câu 3:

(\*) Ở câu 3, em chọn sử dụng mô hình ngôn ngữ ELMo để trích xuất feature vector từ text, vì ELMo là pre-trained model cho bài toán text classification và các vector từ ELMo được tính toán trên đầu mô hình ngôn ngữ hai chiều hai lớp (biLM) giống như mô hình em đang xây dựng. Mô hình biLM này có hai lớp xếp chồng lên nhau. Mỗi lớp có 2 đường chuyển - forward pass và backward pass.

Em sử dụng ELMo với mục đích trích xuất feature vector, nên sẽ không fine-tune mô hình này.





## Bộ dữ liệu tiếng Anh

-Train:

loss: 0.0612 - acc: 0.9821 - val\_loss: 0.0422 - val\_acc: 0.9864

-Test:

O

TP = 221185.0, FP = 697.0, TN = 15629.0, FN = 889.0

Precision: 0.9968586906553935, Recall: 0.9959968298855336, F1-score: 0.9964275739037202

B-art

TP = 7.0, FP = 10.0, TN = 238341.0, FN = 42.0

Precision: 0.4117647058823529, Recall: 0.14285714285714285, F1-score: 0.2121212121212107

I-art

TP = 0.0, FP = 1.0, TN = 238357.0, FN = 42.0

Precision: 0.0, Recall: 0.0, F1-score: 0

B-eve

TP = 12.0, FP = 14.0, TN = 238353.0, FN = 21.0

Precision: 0.46153846153846156, Recall: 0.36363636363636365, F1-score: 0.4067796610169492

I-eve

TP = 9.0, FP = 15.0, TN = 238352.0, FN = 24.0

Precision: 0.375, Recall: 0.2727272727272727, F1-score: 0.3157894736842105

B-geo

TP = 3350.0, FP = 626.0, TN = 234054.0, FN = 370.0

Precision: 0.8425553319919518, Recall: 0.9005376344086021, F1-score: 0.8705821205821206

I-geo

TP = 428.0, FP = 45.0, TN = 237648.0, FN = 279.0

Precision: 0.904862579281184, Recall: 0.6053748231966054, F1-score: 0.7254237288135593

B-gpe

TP = 1465.0, FP = 56.0, TN = 236753.0, FN = 126.0

Precision: 0.9631821170282708, Recall: 0.9208045254556882, F1-score: 0.9415167095115681

I-gpe

TP = 6.0, FP = 4.0 , TN = 238380.0, FN = 10.0

Precision: 0.6, Recall: 0.375, **F1-score: 0.4615384615384615**

B-nat

TP = 11.0, FP = 21.0 , TN = 238357.0, FN = 11.0

Precision: 0.34375, Recall: 0.5, **F1-score: 0.4074074074074074**

I-nat

TP = 0.0, FP = 0.0 , TN = 238395.0, FN = 5.0

Precision: nan, Recall: 0.0, **F1-score: 0**

B-org

TP = 1526.0, FP = 601.0 , TN = 235738.0, FN = 535.0

Precision: 0.7174424071462153, Recall: 0.7404172731683649, **F1-score: 0.728748806112703**

I-org

TP = 1457.0, FP = 693.0 , TN = 235935.0, FN = 315.0

Precision: 0.6776744186046512, Recall: 0.822234762979684, **F1-score: 0.742988271290158**

B-per

TP = 1387.0, FP = 207.0 , TN = 236516.0, FN = 290.0

Precision: 0.8701380175658721, Recall: 0.827072152653548, **F1-score: 0.8480586976459799**

I-per

TP = 1581.0, FP = 333.0 , TN = 236340.0, FN = 146.0

Precision: 0.8260188087774295, Recall: 0.9154603358425014, **F1-score: 0.868442735512222**

B-tim

TP = 1796.0, FP = 128.0 , TN = 236124.0, FN = 352.0

Precision: 0.9334719334719335, Recall: 0.8361266294227188, **F1-score: 0.8821218074656189**

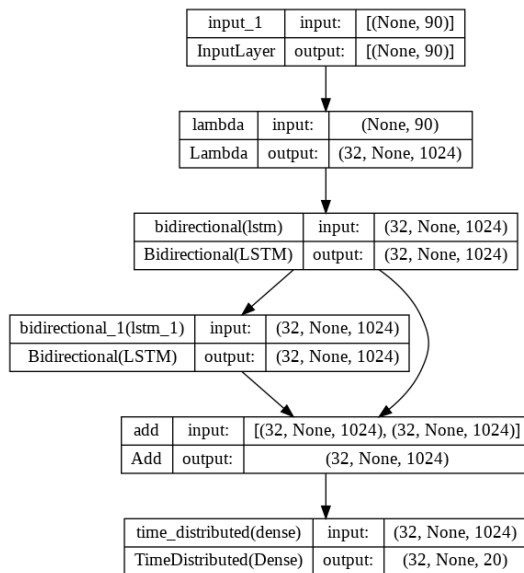
I-tim

TP = 544.0, FP = 185.0 , TN = 237492.0, FN = 179.0

Precision: 0.7462277091906722, Recall: 0.7524204702627939, **F1-score: 0.7493112947658402**

**F1-macro: 0.5974857624336313**





## Bộ dữ liệu tiếng Việt:

-Train:

loss: 0.0616 - acc: 0.9823 - val\_loss: 0.0465 - val\_acc: 0.9864

-Test:

O

TP = 82570.0, FP = 623.0, TN = 5641.0, FN = 446.0

Precision: 0.9925113891793781, Recall: 0.9946275416787126, F1-score: 0.9935683386579548

B-AGE

TP = 136.0, FP = 19.0, TN = 89108.0, FN = 17.0

Precision: 0.8774193548387097, Recall: 0.8888888888888888, F1-score: 0.8831168831168831

I-AGE

TP = 0.0, FP = 0.0, TN = 89280.0, FN = 0.0

Precision: nan, Recall: nan, F1-score: 0

B-DATE

TP = 528.0, FP = 27.0, TN = 88721.0, FN = 4.0

Precision: 0.9513513513513514, Recall: 0.9924812030075187, F1-score: 0.9714811407543699

I-DATE

TP = 487.0, FP = 28.0, TN = 88763.0, FN = 2.0

Precision: 0.945631067961165, Recall: 0.9959100204498977, F1-score: 0.9701195219123505

B-GENDER

TP = 106.0, FP = 16.0, TN = 89138.0, FN = 20.0

Precision: 0.8688524590163934, Recall: 0.8412698412698413, F1-score: 0.8548387096774194

B-JOB

TP = 10.0, FP = 2.0, TN = 89231.0, FN = 37.0

Precision: 0.8333333333333334, Recall: 0.2127659574468085, F1-score: 0.3389830508474576

I-JOB

TP = 0.0, FP = 0.0, TN = 89262.0, FN = 18.0

Precision: nan, Recall: 0.0, F1-score: 0

B-LOCATION

TP = 1127.0, FP = 99.0 , TN = 87904.0, FN = 150.0  
Precision: 0.9192495921696574, Recall: 0.8825371965544244, **F1-score:**  
**0.9005193767479024**

#### I-LOCATION

TP = 1156.0, FP = 205.0 , TN = 87749.0, FN = 170.0  
Precision: 0.8493754592211609, Recall: 0.8717948717948718, **F1-score:**  
**0.8604391514700409**

#### B-NAME

TP = 74.0, FP = 5.0 , TN = 89187.0, FN = 14.0  
Precision: 0.9367088607594937, Recall: 0.8409090909090909, **F1-score:**  
**0.8862275449101796**

#### I-NAME

TP = 0.0, FP = 0.0 , TN = 89275.0, FN = 5.0  
Precision: nan, Recall: 0.0, **F1-score: 0**

#### B-ORGANIZATION

TP = 169.0, FP = 31.0 , TN = 89033.0, FN = 47.0  
Precision: 0.845, Recall: 0.7824074074074074, **F1-score: 0.8125**

#### I-ORGANIZATION

TP = 390.0, FP = 124.0 , TN = 88673.0, FN = 93.0  
Precision: 0.7587548638132295, Recall: 0.8074534161490683, **F1-score:**  
**0.7823470411233701**

#### B-PATIENT\_ID

TP = 615.0, FP = 119.0 , TN = 88496.0, FN = 50.0  
Precision: 0.837874659400545, Recall: 0.924812030075188, **F1-score:**  
**0.8791994281629736**

#### I-PATIENT\_ID

TP = 0.0, FP = 0.0 , TN = 89269.0, FN = 11.0  
Precision: nan, Recall: 0.0, **F1-score: 0**

#### B-SYMPTOM\_AND\_DISEASE

TP = 240.0, FP = 17.0 , TN = 88921.0, FN = 102.0  
Precision: 0.933852140077821, Recall: 0.7017543859649122, **F1-score:**  
**0.8013355592654423**

#### I-SYMPTOM\_AND\_DISEASE

TP = 267.0, FP = 26.0 , TN = 88831.0, FN = 156.0  
Precision: 0.9112627986348123, Recall: 0.6312056737588653, **F1-score:**  
**0.7458100558659219**

#### B-TRANSPORTATION

TP = 46.0, FP = 8.0, TN = 89220.0, FN = 6.0

Precision: 0.8518518518518519, Recall: 0.8846153846153846, **F1-score: 0.8679245283018868**

#### I-TRANSPORTATION

TP = 7.0, FP = 3.0, TN = 89266.0, FN = 4.0

Precision: 0.7, Recall: 0.6363636363636364, **F1-score: 0.6666666666666666**

**F1-macro: 0.660753849874041**

#### -Nhận xét:

- + Sau khi sử dụng ELMo và chỉnh sửa kiến trúc mô hình thì chúng ta cải thiện được hiệu suất 1 cách rõ rệt (tăng từ 0.15 đến 0.22 cho F1-macro). Các tag khác cũng được dự đoán đúng nhiều hơn, không bị trường hợp bị lệch về tag O quá nhiều.
- + Tuy nhiên hiệu suất đạt ở mức ổn, 1 phần do sử dụng mô hình pre-trained để encode, không thể chắc chắn được mô hình đó có thực sự phù hợp với dữ liệu của chúng ta hay không.
- + Hiện tại vì để đạt được hiệu suất chấp nhận được, chi phí và thời gian tính toán ít hơn em chỉ sử dụng ở mức trích xuất vector feature. Để đạt được hiệu suất cao hơn, nếu có thể thì em sẽ fine-tune cả ELMo để mô hình được fit hơn. Tuy nhiên, vì đây là bài thực hành nên em sẽ ưu tiên cho thời gian tính toán hơn.

---

**Kết thúc báo cáo**