Nhận dạng thực thể được đặt tên trên năm lĩnh vực của bộ dữ liệu CrossNER

Đỗ Hùng Dũng*, Dương Thị Nguyệt Minh*, Nguyễn Thiên Long*
Email: *{18520629, 18521095, 18521046}@gm.uit.edu.vn

***Tóm tắt nội dung*—Đối với một số mục đích trong Xử lý ngôn ngữ tự nhiên (NLP), chẳng hạn như trích xuất thông tin (Information extraction), Phân tích cảm xúc (Sentiment analysis), bài toán hỏi đáp (Question Answering - QA),.. Nhận dạng thực thể được đặt tên (Named Entity Recognition - NER) giữ một vai trò quan trọng vì nó giúp xác định và phân loại các thực thể trong văn bản thành các nhóm được xác định trước, chẳng hạn như tên người,vị trí, số lượng, tổ chức,... Trong bài viết này, nhóm chúng tôi có sử dụng một số mô hình học máy và học sâu để xử lý bài toán NER cho bộ dữ liệu CrossNER với năm lĩnh vực trong bộ dữ liệu. Kết quả đạt được trên các mô hình Bi-LSTM, Bi-Gru, Bi-LSTM + CNN, Bi-Gru + CNN, TF-XML-RoBERTa... Từ đó đưa ra các nhận xét và đánh giá cho bộ dữ liệu và các thuật toán được sử dụng.**

***Từ khóa*—**CrossNER, NER, Nhận dạng thực thể, TF-XML-RoBERTa, Bi-LSTM.

I. Giới thiệu

Nhận dạng thực thể có tên (Named Entity Recognition – NER) là một phần trong việc khai thác thông tin và xử lý các tài liệu có cấu trúc và phi cấu trúc. NER có hai nhiệm vụ, đầu tiên là xác định các cụm từ trong văn bản và thứ hai là nhận dạng các cụm này và phân loại chúng vào trong các nhóm đã được định trước như tên người, tổ chức, địa điểm, thời gian, loại sản phẩm, nhãn hiệu,... Ví dụ:

"[ORG U.N.] official [PER Ekeus] heads for [LOC Baghdad]"

Ở đây *Ekeus* là tên người, *U.N.* là tổ chức và *Baghdad* là địa điểm. Từ kết quả của NER có thể xử lý cho nhiều bài toán phức tạp hơn như Chatbot, Question Answering, Search,... NER là một nhiệm vụ cơ bản nhưng cũng là cốt lõi của hệ thống xử lý ngôn ngữ tự nhiên (NLP).

Trong bài báo này chúng tôi sử dụng các phương pháp học máy và mô hình học sâu để tiến hành phân tích và nhận dạng thực thể được đặt tên trên năm lĩnh vực Politics, Natural Science, Music, Literature, và Artificial Intelligence của bộ dữ liệu CrossNER.

II. Các công trình nghiên cứu liên quan

CrossNER: Evaluating Cross-Domain Named Entity Recognition[2]

Zihan Liu, Yan Xu, Tiezheng Yu, et al thấy rằng hầu hết các điểm chuẩn NER hiện tại đều thiếu các loại thực thể chuyên biệt về miền hoặc không tập trung vào một miền nhất định, dẫn đến việc đánh giá miền chéo kém hiệu qủa, Zihan Liu và các cộng sự đã tạo nên bộ dữ liệu CrossNER, bộ sưu tập dữ liệu NER trải dài trên năm miền với các danh mục thực thể chuyên biệt cho các miền khác nhau, sau đó tiến hành tiến hành các thử nghiệm toàn diện để khám phá hiệu quả của việc tận dụng các cấp độ khác nhau của kho ngữ liệu miền và các chiến lược đào tạo để thực hiện đào tạo về khả năng thích ứng miền cho nhiệm vụ miền chéo.

Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition [7]

Erik F. Tjong Kim Sang và Fien De Meulder, đã thực hiện đánh giá bộ dữ liệu The CoNLL-2003 shared task với hai loại ngôn ngữ là tiếng Anh và tiếng Đứ, công việc của họ tập trung chủ yếu và bốn loại thực thể được đặt tên: người, địa điểm, tổ chức và tên của các thực thể khác không thuộc ba nhóm trước. Họ đã sử dụng dữ liệu để phát triển hệ thống nhận dạng thực thể được đặt tên mà bao gồm thành phần của học máy.

III. Bộ dữ liệu CrossNER

A. Bộ dữ liệu

Bộ Dữ liệu:

Bộ dữ liệu CrossNER được lấy từ nguồn: https://github.com/zliucr/CrossNER/tree/main/ner_data.

Bộ dữ liệu CrossNER bao gồm năm lĩnh vực Politics, Natural Science, Music, Literature và Artificial Intelligence được trích xuất các câu từ kho ngữ liệu Wikipedia. Trong mỗi lĩnh vực được chia làm 3 tệp dữ liệu gồm một tập thử nghiệm, một tập huấn luyện và một tập phát triển. Trong mỗi tệp dữ liệu có hai thuộc tính Word thể hiện cho các từ và Tag thể hiện cho các nhãn.

Dưới đây là bản mô tả bộ dữ liệu theo số câu (hình 1) được trích từ bài báo CrossNER: Evaluating Cross-Domain Named Entity Recognition.

Domain	Unlabeled Corpus			Labeled NER			Entity Categories
	# paragraph	# sentence	# tokens	# Train	# Dev	# Test	
Reuters	-	-	-	14,987	3,466	3,684	person, organization, location, miscellaneous
Politics	2.76M	9.07M	176.56M	200	541	651	politician, person, organization, political party, event, election, country, location, miscellaneous
Natural Science	1.72M	5.32M	98.50M	200	450	543	scientist, person, university, organization, country, location, discipline, enzyme, protein, chemical compound, chemical element, event, astronomical object, academic journal, award, theory, miscellaneous
Music	3.49M	9.82M	194.62M	100	380	456	music genre, song, band, album, musical artist, musical instrument, award, event, country, location, organization, person, miscellaneous
Literature	2.69M	9.17M	177.33M	100	400	416	book, writer, award, poem, event, magazine, person, location, organization, country, miscellaneous
Artificial Intelligence	97.04K	287.62K	5.20M	100	350	431	field, task, product, algorithm, researcher, metrics, university, country, person, organization, location, miscellaneous

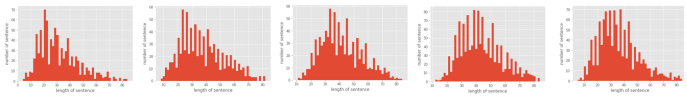
Hình 1: Mô tả bộ dữ liệu theo câu của 5 lĩnh vực [6]

Nhóm chúng tôi phân tích bộ dữ liệu theo từ cho ra được thống kê như hình 2 với Labled là số từ, Unique Words thể hiện số lượng từ độc nhất và Tags thể hiện số nhãn trên mỗi miền dữ liệu.

Domain	Labeled			Unique Words		
	#Dev	#Test	#Train	#Dev	#Test	#Train
Artificial Intelligence	11268	13421	3881	3054	3507	1441
Literature	14902	16572	3881	4401	4654	1613
Music	15970	20069	4008	4155	4884	1478
Natural Science	16588	20029	7299	4711	5556	2661
Politics	25164	28235	8583	4869	5835	2579

Tags	Artificial Intelligence	Literature	Music	Natural Science	Politics
	29	25	27	35	19

Hình 2: Mô tả bộ dữ liệu theo từ của 5 lĩnh vực



Hình 3: Phân bố độ dài câu theo thứ tự AI-Literature-Music-Police-Science

Hình 4: Tỉ lệ nhãn của năm lĩnh vực trên 3 tệp dữ liệu

Từ Hình 4 ta có thể thấy, tỉ lệ của nhãn O so với các nhãn còn lại có sự chênh lệch rất lớn, việc này dẫn đến nhiều khó khăn trong việc huấn luyện mô hình, đồng thời sẽ phát sinh ra lỗi trong quá trình dự đoán.

Định dạng dữ liệu:

Trong thuộc tính Word trên tất cả các bộ dữ liệu đều chứa một từ trên mỗi dòng với các dòng trống thể hiện ranh giới câu, và thuộc tính Tag cho biết từ hiện tại có bên trong một thực thể được đặt tên hay không.Thẻ cũng mã hóa loại thực thể được đặt tên. Đây là câu ví dụ:

U.N.	NNP	I-ORG
official	NN	O
Ekeus	NNP	I-PER
heads	VBZ	O
for	IN	O
Baghdad	NNP	I-LOC
.	.	O

Mỗi dòng chứa ba trường: Word, thẻ POP, và thẻ thực thể được đặt tên của nó. Các từ được gắng thẻ O nằm ngoài thực thể được đặt tên, thẻ I-XXX được dùng cho các từ nằm bên trong thực thể được đặt tên thuộc loại XXX. Bất cứ khi nào hai thực thể XXX ngay cạnh nhau, từ đầu tiên của thực thể thứ hai sẽ được đặt B-XXX để cho thấy rằng nó bắt đầu một thực thể khác. Dữ liệu chứa các từ thuộc loại người (PER), tổ chức (ORG), vị trí (LOC) và các tên khác (MISC). Sơ đồ gắng thẻ này là sơ đồ IOB được đưa ra lần đầu bởi Ramshaw và Marcus (1995).

IV. Phương pháp tiếp cận

Ở phần này, nhóm chúng tôi tiến hành theo nhiều mô hình học sâu thường dùng cho việc xử lý các bài toán về xử lý ngôn ngữ tự nhiên.

A. XLM-RoBERTa

XLM - RoBERTa là một mô hình Xử lý ngôn ngữ tự nhiên sử dụng các kỹ thuật đào tạo tự giám sát trong việc hiểu biết đa ngôn ngữ. Đây là mô hình được phát triển bởi Facebook AI nên có thể xử lý tới 100 ngôn ngữ, bao gồm cả những ngôn ngữ ít tài nguyên như tiếng Urdu, Miến Điện hay Swahili

1) BERT: BERT (Bidirection Encoder Reprsentations from Transfomers) là một mô hình học sẵn (pre-train model) được phát triển bởi Google AI (nhóm tác giả gồm Jacob Devlin cùng các cộng sự) và công bố vào ngày 2/11/2018. BERT cũng hoạt động dựa trên phương pháp tìm ra đại diện của từ thông qua ngữ cảnh của chúng như các kỹ thuật đã có trước đó như Word2Vec, FastText,... Nhưng điều làm cho BERT trở nên khác biệt là ở chỗ nó có thể biểu diễn ngữ cảnh 2 chiều thay vì chỉ 1 vector đại diện cho mỗi từ của hai kĩ thuật đã nêu. Điều đó tạo ra khả năng làm tăng độ phong phú của ngữ nghĩa trong mô hình ngôn ngữ.[4]

System	Dev F1	Test F1
ELMo (Peters et al., 2018a)	95.7	92.2
CVT (Clark et al., 2018)	-	92.6
CSE (Akbik et al., 2018)	-	**93.1**
Fine-tuning approach		
$\text{BERT}_{\text{LARGE}}$	96.6	92.8
$\text{BERT}_{\text{BASE}}$	96.4	92.4
Feature-based approach ($\text{BERT}_{\text{BASE}}$)		
Embeddings	91.0	-
Second-to-Last Hidden	95.6	-
Last Hidden	94.9	-
Weighted Sum Last Four Hidden	95.9	-
Concat Last Four Hidden	96.1	-
Weighted Sum All 12 Layers	95.5	-

Hình 5: Kết quả của mô hình BERT trên tập CoNLL2003, được trình bày trong bài báo công bố[4]

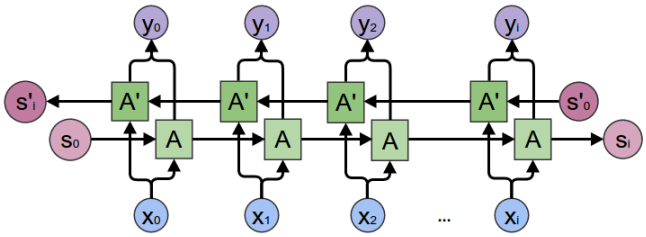
2) RoBERTa: RoBERTa là phiên bản tinh chỉnh mô hình BERT ban đầu cùng với thao tác dữ liệu và đầu vào do Facebook AI. Cụ thể, các tác giả của RoBERTa (gồm Yinhan Liu và các cộng sự) phát hiện thấy tiềm năng phát triển của BERT khi được đào tạo trên các bộ dữ liệu lớn hơn nên đã đưa vào một dữ liệu tới 160GB các văn bản không nén (Gồm các tài liệu như BookCorpus và wikipedia tiếng Anh, CC-News, OpenWebText và một tập tập hợp con dữ liệu CommonCrawl có phong cách giống lược đồ Winograd). Ngoài ra, trong quá trình đào tạo của RoBERTa cũng thay đổi cách thức che giấu đối tượng bằng mặt nạ động. Sự cải tiến này cho kết quả gần như tương tự và có phần nhỉnh hơn so với phương pháp tĩnh. Thêm vào đó, như đã nói ở trên thì các tác giả của RoBERTa cũng thay đổi cách trình bày đầu vào, dự đoán câu tiếp theo của mô hình tiền nhiệm và đặc biệt là tối ưu hoá nhanh hơn, có thể cải thiện hiệu suất tác vụ cuối vì có có kích thước lô lớn hơn. Cuối cùng, về mặt mã hoá thì RoBERTa sử dụng lược đồ mã hoá Byte-pair Encoding cấp byte với từng vựng chứa 50 nghìn đơn vị từ khoá đơn trong khi BERT sử dụng BPE cấp ký tự với 30 nghìn từ vựng.[5]

Model	**Accuracy**	**Middle**	**High**
Single models on test (as of July 25, 2019)			
$\text{BERT}_{\text{LARGE}}$	72.0	76.6	70.1
$\text{XLNet}_{\text{LARGE}}$	81.7	85.4	80.2
RoBERTa	**83.2**	**86.5**	**81.3**

Hình 6: Kết quả so sánh mô hình RoBERTa Large và BERT Large trong bài báo công bố RoBERTa[5]

B. Bi-directional Long Short-Term Memory - BiLSTM

BiLSTM là LSTM hai chiều. Cấu trúc này cho phép các mạng có cả thông tin trình tự chuyển tiếp và lùi về ở mọi bước thời gian. BiLSTm sẽ chạy thông tin đầu vào theo hai cách: Từ quá khứ sang tương lai và từ tương lai đến quá khứ. phương pháp này có thể bảo toàn thông tin từ tương lai và sử dụng hai trạng thái ẩn kết hợp lại với nhau.Có thể từ bất kỳ thời điểm nào lưu trữ thông tin từ cả quá khứ và tương lai.[3]



Hình 7: Mô hình BiLSTM

Ví dụ trong câu:

He said: "Teddy bears are one sale!"

He said: "Teddy Roosevelt was a great Prisident!"

Ở đây, đối với từ "Teddy", không thể nói từ tiếp theo nó sẽ là "bears" hay "Roosevelt", nó sẽ phụ thuộc vào ngữ cảnh của câu. BiLSTM là kiến trúc có thể sử dụng bất kỳ mô hình RNN nào.

C. Bi-directional Gate Recurrent Unit - Bi-GRU

Bi-GRU Dựa trên mạng neural hai chiều BiLSTM, mạng neural GRU có cấu trúc mạng hình tròn, xác định thông tin đầu ra hiện tại thông qua thông tin đầu vào tại thời điểm hiện tại và thông tin đầu ra tại thời điểm trước đó. Do đó thông tin đầu ra tại mỗi thời điểm trong mạng neural Gru phụ thuộc vào thông tin trong quá khứ. Vì vậy,thuộc tính chuỗi của Gru liên quan chặc chẽ đến vấn đề ghi nhãn tuần tự và được áp dụng cho nhiệm vụ phân đoạn từ.

Một mạng neural GRU có hai cổng điều khiển, một cổng đặt lại và một cổng cập nhật như ở công thức số (1), Cổng đặt lại xác định lượng thông tin cần được quên trong trạng thái ẩn của thời điểm trước đó. Khi giá trị của cổng đặt lại gần 0, thông tin của thời điểm trước đó sẽ bị quên. Khi giá trị gần bằng 1, thông tin ẩn của thời điểm trước đó được giữ lại trong thông tin bộ nhớ hiện tại. Cổng cập nhật xác định có bao nhiêu thông tin ở trạng thái ẩn tại thời điểm trước đó sẽ được đưa vào trạng thái ẩn hiện tại. Khi giá trị của cổng cập nhật gần bằng 0, thông tin ở trạng thái ẩn tại thời điểm trước đó sẽ bị quên. Khi giá trị gần hơn 1, thông tin được giữ lại ở trạng thái ẩn hiện tại.[1]

Trong Hình 8, zt là cổng cập nhật, *rt* là cổng đặt lại, *(~ht)* là trạng thái ẩn ứng viên của nút hiện đang ẩn, *ht* là trạng thái ẩn hiện tại,*xt* là đầu vào của mạng neural hiện tại và *(ht-1)* là trạng thái ẩn ở thời điểm trước đó. Công thức tính toán chi tiết như sau:

$$z_t = \sigma(w_{zx}x_t + u_{zh}h_{t-1}) \quad (1)$$

$$r_t = \sigma(w_{rx}x_t + u_{rh}h_{t-1}) \quad (2)$$

$$\tilde{h}_t = \tan(w_{hx}x_t + r_t \odot u_{hh}h_{t-1}) \quad (3)$$

$$h_t = (1 - z_t) \odot \tilde{h} + z_t \odot h_{t-1} \quad (4)$$

Hình 8: Công thức GRU

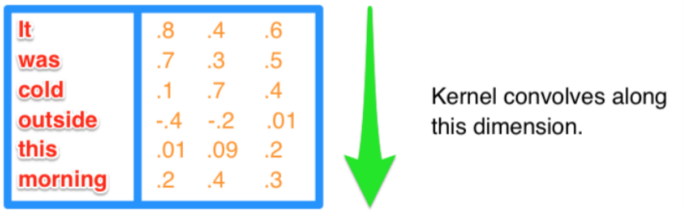
Trong đó là hàm kích hoạt sigmoid, nằm trong khoảng từ 0 đến 1, là kết quả Hadamard của ma trận, *w* và *u* là các ma trận trọng số cần phải học, và *zt* và *rt* nằm trong khoảng từ 0 đến 1 . Như được hiển thị trong Công thức (3), cổng đặt lại bao gồm các vector từ 0 đến 1. Do đó, sau khi nhận được kết quả Hadamard, cổng đặt lại xác định có bao nhiêu trạng thái ẩn trong thời gian trước đó sẽ bị quên trong bộ nhớ hiện tại. Thông tin đầu vào hiện tại sau đó được thêm vào và đặt trong chức năng kích hoạt. Do đó, *(~ht)* ghi lại tất cả thông tin quan trọng thông qua cổng đặt lại và thông tin đầu vào. Cổng cập nhật xác định trạng thái hiện đang ẩn ht bằng cách tác động lên *h(t 1)* và *(~ht)* và chuyển nó đến đơn vị tiếp theo. Như được trình bày trong Công thức (4), số hạng đầu tiên đến *(1 - zt)* xác định thông tin nào cần được quên và thông tin tương ứng trong nội dung bộ nhớ được cập nhật tại thời điểm này. Số hạng thứ hai của công thức xác định có bao nhiêu thông tin *h(t 1)* được giữ lại ở trạng thái ẩn hiện tại. Do đó, ht quyết định thu thập thông tin cần thiết ở *(~ht)* và *h(t 1)* thông qua cổng cập nhật.

D. CNN (Conv1D)

Convolutional Neural Network (CNN hoặc ConvNet) được tạm dịch là: Mạng nơ ron tích tụ. Đây được xem là một trong những mô hình của Deep Learning – tập hợp các thuật toán để có mô hình dữ liệu trừu tượng hóa ở mức cao bằng cách sử dụng nhiều lớp xử lý cấu trúc phức tạp. Hiểu đơn giản, CNN là một lớp của mạng nơ-ron sâu.[8]

Thông thường, trong lĩnh vực thị giác máy tính, các vấn đề phân loại hình ảnh thường được giải quyết bằng cách sử dụng lớp Conv2D. Về mặt lý thuyết, Conv2D hoạt động bằng cách áp dụng các hạt nhân tiến dọc theo không gian 2 chiều. Trong trường hợp của hình ảnh, các bộ lọc của lớp tích chập 2 chiều này đang dịch chuyển dọc theo chiều cao và chiều rộng của nó. Mặt khác, Conv1D chỉ di chuyển dọc theo một trục duy nhất, do đó, hoàn toàn hợp lý khi áp dụng loại lớp tích chập này cho dữ liệu tuần tự như văn bản hoặc tín hiệu.

Đối với Conv1D, các câu được đệm / cắt đến một độ dài tối đa nhất định; các từ được mã hóa dưới dạng vectơ của một thứ nguyên đầu vào nhất định; hạt nhân biến đổi dọc theo các kích thước được cho bởi chiều dài tối đa và kích thước đầu vào.



Hình 9: Chuyển đổi Conv1D

V. Thực nghiệm và phân tích

Trong phần này, chúng tôi sẽ giới thiệu cách thức đánh giá cho bài toán nhận diện thực thể được đặt tên trên bộ dữ liệu CrossNER.

A. Các thông số đánh giá

Chỉ số đánh giá: Trong bài báo cáo này, nhóm chúng tôi sẽ tiến hành đánh giá các mô hình bằng chỉ số F1-score vì các lý do như sau:

Trước tiên, nhóm sử dụng F1-score là trung bình điều hoà của 2 chỉ số Precision và Recall nên có thể có cái nhìn tổng quát hơn về kết quả thu được.

Mặc khác, do bài báo cáo khoa học về CrossNER sử dụng độ đo là F1-score nên việc dùng cùng một thang đo có thể dễ dàng so sánh được các mô hình nhóm đã thực hiện với các mô hình của nhóm tác giả.

B. Kết quả đánh giá thực nghiệm

Model	AI	Literature	Music	Politics	Science
Jia and Zhang (2020) + DAPT (Spanlevel & Integrated)	0.69	0.69	0.74	0.71	0.68
Best BERT*	0.63	0.69	0.76	0.72	0.69
BiLSTM	0.89	0.84	0.57	0.67	0.64
BiLSTM_CNN	0.89	0.81	0.63	0.69	0.71
BiGru	0.87	0.79	0.73	0.71	0.73
BiGru_CNN	0.82	0.73	0.73	0.69	0.70
XLM-RoBERTa	0.79	0.9	0.88	0.9	0.82

Hình 10: So sánh kết quả các mô hình dựa trên độ đo f1-score (*: Mô hình BERT-base Pre-train on the Source Domain then Fine-tune on Target Domains (Pre-train then Fine-tune với Span-level và Intergrated)

Bảng kết quả trên cho thấy các mô hình do nhóm sử dụng cho kết quả cao hơn mô hình tiêu biểu được lấy ra từ bài báo cáo bộ dữ liệu CrossNER. Trong đó, mô hình cho kết quả f1-score cao nhất là mô hình XLM-RoBERTa.

C. Phân tích lỗi

Như đã nói ở mục giới thiệu bộ dữ liệu, bộ dữ liệu được gán quá ít nhãn dẫn đến quy mô của bộ dữ liệu là khá nhỏ. Ngoài ra sự phân bố nhãn của các tập dữ liệu là không phù hợp, khi tập train chiếm số lượng rất ít so với 2 tập dev là Test. Và đặc biệt là sự phân bố giữa các nhãn cũng có sự chênh lệch lớn. Điển hình là nhãn O luôn có số lượng lớn hơn so với các nhãn khá gây ra tình trạng overfitting.

Vì đang tiến hành đo thực nghiệm trên bộ dữ liệu đa nhãn nên nhóm chúng tôi đề xuất đến độ đo F1 micro và F1 macro

để có cái nhìn tổng quan hơn về kết quả huấn luyện các mô hình đã thực hiện.

Model	F1	AI	Literature	Music	Politics	Science
BiLSTM	F1-macro	0.03	0.03	0.02	0.05	0.02
	F1-micro	0.88	0.84	0.56	0.67	0.64
BiLSTM_CNN	F1-macro	0.03	0.04	0.03	0.05	0.03
	F1-micro	0.89	0.80	0.62	0.69	0.71
BiGru	F1-macro	0.03	0.04	0.03	0.06	0.03
	F1-micro	0.87	0.79	0.73	0.71	0.73
BiGru_CNN	F1-macro	0.04	0.05	0.04	0.06	0.03
	F1-micro	0.82	0.73	0.73	0.69	0.75
XLM-RoBERTa	F1-macro	0.27	0.65	0.5	0.61	0.34
	F1-micro	0.79	0.90	0.88	0.9	0.82

Hình 11: Các độ đo f1-micro và f1-macro của từng mô hình phản ánh sự mất cân bằng dữ liệu

Như đã thấy ở Hình 11, dựa vào các kết quả đã được nhóm chúng tôi thống kê cho thấy các mô hình được thực hiện bởi nhóm cho ra kết quả F1 macro khá thấp và cực kì thấp ở một số mô hình. Tuy nhiên vì đây là bộ dữ liệu ngôn ngữ đa nhãn nên chúng ta sẽ chú trọng vào ý nghĩa của F1 micro.

Ngoài ra nhóm chúng tôi còn có thêm nhận xét về kết quả thực nghiệm. Năm bộ dữ liệu cho kết quả khá chênh lệch

VI. Kết luận

Trong bài báo cáo này chúng tôi đã tìm hiểu về bộ dữ liệu CrossNER - một bộ dữ liệu khá lớn với năm lĩnh vực Politics, Natural, Science, Music, Literature và Artificial Intelligence.

Bên cạnh đó, chúng tôi cũng đã thử nghiệm các mô hình học sâu trên bộ dữ liệu như Bi-LSTM, Bi-Gru, Bi-LSTM + CNN, Bi-Gru + CNN, XML-RoBERTa. Tiến hành so sánh kết quả F1 với các mô hình do nhóm tác giả công bố bộ dữ liệu và đạt được kết quả tốt. Ngoài ra, nhóm còn đề xuất độ đo mới cho bộ dữ liệu để mang nhiều ý nghĩa hơn và cũng đã tiến hành đo thử với kết quả thu được là f1-micro cao trên cả năm lĩnh vực nhưng đối với f1-macro lại cho ra kết quả tương đối thấp.

Cuối cùng, nhóm cũng nhận ra được một số mặt hạn chế trong việc xây dựng bộ dữ liệu của nhóm tác giả về việc gán nhãn cũng như phân bố tài nguyên cho từng tập dữ liệu. Đồng thời còn cho thấy được kết quả chưa được tổng quát của các tác giả khi chỉ đưa ra 1 độ đo để nhận xét về tập dữ liệu.

Tài liệu

[1] Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014.

[2] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale. *CoRR*, abs/1911.02116, 2019.

[3] Zhiyong Cui, Ruimin Ke, and Yinhai Wang. Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction. *CoRR*, abs/1801.02143, 2018.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[5] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.

[6] Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. Crossner: Evaluating cross-domain named entity recognition. *CoRR*, abs/2012.04373, 2020.

[7] Erik F. Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. *CoRR*, cs.CL/0306050, 2003.

[8] Wei Wang and Jianxun Gang. Application of convolutional neural network in natural language processing. In *2018 International Conference on Information Systems and Computer Aided Education (ICISCAE)*, pages 64–70, 2018.