**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO ĐỒ ÁN**

KHAI THÁC DỮ LIỆU

Đề tài:

FAKE NEWS DETECTION ON SOCIAL MEDIA USING FUZZY DEEP HYBRID NETWORK

**Sinh viên thực hiện:**

Hồ Văn Vinh – 21520530

Trần Minh Hoàng - 21522101

Nguyễn Lý Đăng Khoa - 21522229

Trương Công Quốc Triệu - 21522714

*Thành phố Hồ Chí Minh, tháng 5 năm 2024*

# LỜI CẢM ƠN

Trước hết, nhóm chúng em xin gửi lời cảm ơn sâu sắc đến tập thể quý thầy cô trường Đại học Công nghệ Thông tin - Đại học Quốc gia TP.HCM và quý thầy cô khoa Hệ thống thông tin đã tạo điều kiện, giúp chúng em học tập và có được những kiến thức cơ bản làm tiền đề giúp chúng em hoàn thành được dự án này.

Đặc biệt, nhóm chúng em xin gửi lời cảm ơn chân thành và sâu sắc tới thầy Dương Phi Long (Giảng viên giảng dạy lý thuyết và thực hành môn Khai thác dữ liệu– IS252). Nhờ sự hướng dẫn tận tình và chu đáo của cô, nhóm chúng em đã học hỏi được nhiều kinh nghiệm và hoàn thành thuận lợi, đúng tiến độ cho dự án của mình.

Ngoài ra, chúng em cũng gửi lời cảm ơn đến tập thể lớp khoảng thời gian qua đã đồng hành cùng nhau. Cảm ơn sự đóng góp của tất cả các bạn cho những buổi học luôn sôi nổi, thú vị và dễ tiếp thu.

Trong quá trình thực hiện đồ án, nhóm chúng em luôn giữ một tinh thần cầu tiến, học hỏi và cải thiện từ những sai lầm, tham khảo từ nhiều nguồn tài liệu khác nhau và luôn mong tạo ra được sản phẩm chất lượng nhất có thể. Tuy nhiên, do vốn kiến thức còn hạn chế trong quá trình trau dồi từng ngày, nhóm chúng em không thể tránh được những sai sót, vì vậy chúng em mong rằng quý thầy cô sẽ đưa ra nhận xét một cách chân thành để chúng em học hỏi thêm kinh nghiệm nhằm mục đích phục vụ tốt các dự án khác trong tương lai. Xin chân thành cảm ơn quý thầy cô!

**Nhóm thực hiện**

# NHẬN XÉT CỦA GIẢNG VIÊN

*……., ngày……...tháng……năm 2023*

**Người nhận xét**

*(Ký tên và ghi rõ họ tên****)***

# MỤC LỤC

[LỜI CẢM ƠN 2](#_Toc169779880)

[NHẬN XÉT CỦA GIẢNG VIÊN 3](#_Toc169779881)

[MỤC LỤC 4](#_Toc169779882)

[DANH MỤC HÌNH ẢNH 6](#_Toc169779883)

[CHƯƠNG 1: GIỚI THIỆU BÀI BÁO 7](#_Toc169779884)

[1.1. Giới thiệu chung về bài báo 7](#_Toc169779885)

[1.2. Tổng quan đề tài 8](#_Toc169779886)

[1.3. Mục đích 9](#_Toc169779887)

[CHƯƠNG 2: GIỚI THIỆU CHUNG VỀ BÀI TOÁN, CÁC HƯỚNG TIẾP CẬN 10](#_Toc169779888)

[2.1. Tổng quan vấn đề 10](#_Toc169779889)

[2.2. Các hướng tiếp cận 11](#_Toc169779890)

[CHƯƠNG 3: GIỚI THIỆU VỀ BỘ DỮ LIỆU 13](#_Toc169779891)

[3.1. Giới thiệu chung về bộ dữ liệu 13](#_Toc169779892)

[3.2. Chi tiết bộ dữ liệu: 13](#_Toc169779893)

[3.3. Thống kê của bộ dữ liệu 15](#_Toc169779894)

[CHƯƠNG 4: MÔ HÌNH THUẬT TOÁN 17](#_Toc169779895)

[4.1. Giới thiệu chung về phương pháp, kỹ thuật áp dụng 17](#_Toc169779896)

[4.2. Mô hình TextCNN dùng ở kênh số 1 & 2 21](#_Toc169779897)

[4.2.1. Mô hình TextCNN 23](#_Toc169779898)

[4.2.2. Chính quy hoá mô hình TextCNN 24](#_Toc169779899)

[CHƯƠNG 5: THỰC NGHIỆM MÔ HÌNH VÀ ĐÁNH GIÁ KẾT QUẢ 26](#_Toc169779900)

[5.1. Cấu hình mô hình nhóm tác giả sử dụng 26](#_Toc169779901)

[5.2. Chuẩn bị công cụ 26](#_Toc169779902)

[5.3. Huấn luyện mô hình 27](#_Toc169779903)

[CHƯƠNG 6: ƯU ĐIỂM, HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN 49](#_Toc169779904)

[6.1. Ưu điểm 49](#_Toc169779905)

[6.2. Hạn chế 50](#_Toc169779906)

[6.3. Hướng phát triển 50](#_Toc169779907)

[PHÂN CÔNG CÔNG VIỆC 52](#_Toc169779908)

# DANH MỤC HÌNH ẢNH

# GIỚI THIỆU BÀI BÁO

## Giới thiệu chung về bài báo

Bài báo "Fuzzy Deep Hybrid Network for Fake News Detection" của Cheng Xu và M-Tahar Kechadi từ Đại học College Dublin, được trình bày tại Hội nghị Quốc tế về Công nghệ Thông tin và Truyền thông lần thứ 12 (SOICT 2023) tại TP. Hồ Chí Minh, đề xuất một mô hình kết hợp dựa trên logic mờ (fuzzy) và học sâu để cải thiện hiệu quả phát hiện tin giả. Mô hình này tận dụng sự kết hợp giữa nội dung các bài báo và thông tin ngữ cảnh của nó, bao gồm thông tin văn bản và số liệu, để nâng cao khả năng phân loại tin giả. Các tác giả đã thử nghiệm mô hình trên bộ dữ liệu kiểm chứng LIAR và đạt được kết quả vượt trội so với các mô hình hiện có. Bài báo cũng trình bày chi tiết về cấu trúc mô hình, cách xử lý thông tin văn bản và ngữ cảnh, cũng như lớp mờ được tích hợp vào mạng học sâu. Kết quả nghiên cứu cho thấy việc kết hợp logic mờ với học sâu không chỉ cải thiện độ chính xác trong việc phát hiện tin giả mà còn cung cấp một công cụ đáng tin cậy để chống lại thông tin sai lệch trong thời đại kỹ thuật số.

Các thông tin chung:

* Đường dẫn: [LINK](https://paperswithcode.com/paper/fuzzy-deep-hybrid-network-for-fake-news)
* Nhóm tác giả: Cheng Xu, M-Tahar Kechadi
* Công bố: tại HỘI THẢO QUỐC TẾ VỀ CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG SOICT 2023 (SOICT 2023), December 07–08, 2023, Ho Chi Minh, Vietnam. ACM, New York, NY, USA
* Thông số: 1 tệp 8 trang, định dạng pdf
* Bố cục bài báo:
  + Chương 1: Giới thiệu tổng quan
  + Chương 2: Nghiên cứu liên quan
  + Chương 3: Bộ dữ liệu và các vấn đề liên quan
  + Chương 4: Đề xuất và giới thiệu mô hình
  + Chương 5: Phân tích kết quả thử nghiệm
  + Chương 6: Tổng kết nghiên cứu và đưa ra định hướng trong tương lai
* Năm xuất bản: 07-08/12/2023
* Dataset: <https://paperswithcode.com/dataset/liar>
* Code: <https://github.com/chengxuphd/FDHN>

## Tổng quan đề tài

Tổng quan các thông tin chung:

* Tên: Phát hiện tin tức giả trên mạng xã hội sử dụng mô hình Fuzzy Deep Hybrid Network
* Sử dụng các mô hình, thuật toán deep learning được giới thiệu trong các bài báo trước nhưng được cải tiến nhờ Fuzzy logic bằng cách thêm một Fuzzy Layer có nhiệm vụ nhận đầu ra của mô hình học sâu trước đó và cho ra một biểu diễn dùng để tính toán kết quả cuối cùng. Các mô hình học sâu được tác giả sử dụng là:
  + TextCNN model được giới thiệu trong bài báo [*Convolutional Neural Networks for Sentence Classification*](https://doi.org/10.3115/v1/D14-1181) của tác giả Yoon Kim vào năm 2014
  + Bidirectional LSTM (Bi-LSTM) trong bài báo [*Bidirectional recurrent neural networks*](https://ieeexplore.ieee.org/document/650093) của nhóm tác giả Mike Schuster và Kuldip K Paliwal vào năm 1997
* Logic mờ (tiếng Anh: Fuzzy logic) được phát triển từ lý thuyết tập mờ để thực hiện lập luận một cách xấp xỉ thay vì lập luận chính xác theo logic vị từ cổ điển( tứ 0 đến 1 thay vì 0,1)
* Cho vd về fuzzy logic: các khái niệm như barely true, halfly true, mostly true, absolute true, absolute false, pants on fire
  + Barely true: Bầu trời màu tím
  + Hafly true: Ăn cà rốt giúp cải thiện tầm nhìn
  + Mostly true: Tập thể dục giúp cải thiện sức khỏe
  + Pants on fire: Mỹ xây dựng căn cứ quân sự trên mặt trăng
  + Absolute False: Trái đất phẳng
  + Absolute True: Nước sôi ở 100 độ C

## Mục đích

Đề xuất và kiểm chứng một phương pháp tiếp cận hybrid với sự kết hợp của fuzzy logic so với logic truyền thống về cách đánh giá mức độ đúng của 1 tin tức nhằm giải quyết thách thức của việc phát hiện tin giả trong thời đại số. Sự lan truyền của tin giả không chỉ đe dọa quá trình dân chủ mà còn làm suy giảm niềm tin vào các phương tiện truyền thông. Trong bối cảnh này, các kỹ thuật học máy và học sâu đã được áp dụng để tự động phát hiện tin giả, nhưng vẫn còn nhiều không gian để cải tiến.

Mô hình được đề xuất sẽ kết hợp dựa trên logic mờ để nâng cao hiệu suất phát hiện tin giả. Mô hình này kết hợp thông tin từ các bài báo tin tức cùng với ngữ cảnh văn bản và số liệu của chúng. Mục tiêu là cải thiện độ chính xác trong việc phân loại tin giả và cung cấp một công cụ đáng tin cậy để chống lại thông tin sai lệch. Các thí nghiệm được thực hiện trên một bộ dữ liệu kiểm chứng thực tế đã cho thấy mô hình này đạt được kết quả vượt trội, đề xuất rằng sự kết hợp giữa logic mờ và học sâu có thể tăng cường khả năng phát hiện tin giả.

# GIỚI THIỆU CHUNG VỀ BÀI TOÁN, CÁC HƯỚNG TIẾP CẬN

## Tổng quan vấn đề

Phát hiện tin tức giả là một nhiệm vụ đầy thách thức do tính chất phức tạp và nhiều sắc thái của ngôn ngữ tự nhiên. Những tiến bộ trong học máy những năm gần đây đã thúc đẩy việc ứng dụng các mô hình như text classification, neural network và propagation model để phát hiện tin giả.

Những nghiên cứu ban đầu tập trung phát triển các bộ phân lớp để phân loại tin thật/tin giả dựa trên đặc trưng văn bản như trong công trình nghiên cứu của nhóm tác giả Yang, [*Learning approaches for detecting and tracking news events*](https://doi.org/10.1109/5254.784083)năm 1999. Sau đó, các nghiên cứu bắt đầu tích hợp thông tin ngữ cảnh bổ sung liên quan đến tin tức như thông tin tác giả, các chuỗi truyền tin để nâng cao hiệu quả. Điển hình là nhóm tác giả Castillo đã đề xuất hướng tiếp cận kết hợp việc phân tích văn bản với các thông tin ngữ cảnh như số lần chia sẻ, trích dẫn, … trong bài báo [*Information Credibility on Twitter*](https://doi.org/10.1145/1963405.1963500) năm 2011, minh chứng cho tầm quan trọng của việc xét đến các thông tin ngữ cảnh liên quan đến tin tức trong bài toán nhận diện tin giả.

Theo thời gian, lĩnh vực này ngày một đa dạng, dẫn đến xuất hiện nhiều lĩnh vực phụ và các phương pháp đã phát triển thành các ứng dụng cấp hệ thống tương đối độc lập. Trong bài khảo sát [*A Survey of Fake News: Fundamental Theories, Detection Methods, and Opportunities*](https://doi.org/10.1145/3395046), các tác giả đã chia thành nhiều hướng tiếp cận khác nhau dựa trên loại thông tin sử dụng cho bài toán nhận diện tin giả như knowledge-based, style-based, propagation-based và source-based. Các nghiên cứu sử dụng kết hợp nhiều luồng thông tin như [*news environment perception (NEP)*](https://aclanthology.org/2022.acl-long.311), [*a recurrent neural network with an attention mechanism (att-RNN)*](https://doi.org/10.1145/3123266.3123454)và [*event adversarial neural network (EANN)*](https://dl.acm.org/doi/10.1145/3219819.3219903)đã đạt được kết quả ấn tượng,

Nhìn chung đây là một vấn đề ngày càng trở nên quan trọng trong thời đại kỹ thuật số hiện nay khi tin giả lan truyền nhanh chóng trên các nền tảng truyền thông xã hội, gây ảnh hưởng đến quá trình dân chủ và làm suy giảm niềm tin của công chúng vào các nguồn tin tức. Các mô hình học máy cho phép phân loại tự động các bài báo tin tức thành tin thật hay tin giả dựa trên đặc trưng nội dung văn bản và các thông tin ngữ cảnh liên quan:

1. Text classification models: Các mô hình phân loại văn bản như CNN, RNN, LSTM đã được ứng dụng để trích xuất đặc trưng từ nội dung tin tức và phân loại tin giả/tin thật.
2. Sử dụng Neural network: Các mạng nơ-ron sâu như BERT, GPT đã được tiền huấn luyện trên dữ liệu lớn và có khả năng trích xuất đặc trưng cao cấp từ văn bản hiệu quả.
3. Propagation-based methods: Các phương pháp dựa trên cách tin tức lan truyền trên mạng xã hội để phát hiện tin giả dựa trên tốc độ, phạm vi lan truyền.

Tuy nhiên, ứng dụng fuzzy logic vào bài toàn phát hiện tin giả vẫn còn khá mới mẻ. Trong bài báo các tác giả sẽ tập trung khai thác về chủ đề này.

## Các hướng tiếp cận

Fuzzy logic là một phương pháp trong lĩnh vực trí tuệ nhân tạo, mô phỏng cách con người suy luận khi đối diện với sự không rõ ràng và mờ mịt trong thông tin. Nó cho phép xử lý dữ liệu không chắc chắn bằng cách sử dụng giá trị mờ (fuzzy values) thay vì chỉ có giá trị đúng/sai (true/false) như trong logic cổ điển. Điều này giúp giải quyết các vấn đề thực tế phức tạp, nơi thông tin luôn không hoàn toàn chắc chắn. Trong những năm gần đây, việc tích hợp fuzzy logic vào các kĩ thuật nhận diện tin giả nhận được nhiều sự quan tâm, nhiều nhà nghiên cứu đã cố gắng tích hợp nó vào các mô hình máy học và học sâu. Nhóm tác giả Das, trong khảo sát [*A Survey on Fuzzy Deep Neural Networks*](https://doi.org/10.1145/3369798)*,* đã chia các hướng tiếp cận học sâu với fuzzy logic thành hai loại như sau:

* Mô hình tích hợp - integrated model: fuzzy logic được nhúng vào trong mạng nơ-ron.
* Mô hình kết hợp - ensemble model: fuzzy logic được sử dụng ở các giai đoạn khác nhau trong quá trình xử lý của mạng nơ-ron, có thể ở thành phần upstream hoặc downstream.

Mô hình tác giả đề xuất sử dụng fuzzy logic như một module độc lập, thành phần riêng biệt với các mô hình học sâu được sử dụng. Fuzzy Layer được kết hợp cùng các biểu diễn nơ-ron để tạo thành một bộ phân lớp.

Mặc dù có nhiều nghiên cứu được đưa ra nhưng việc tích hợp fuzzy logic vào phát hiện tin tức giả vẫn chưa được khám phá rộng rãi. Lý do tác giả muốn tích hợp fuzzy logic vào việc nhận diện tin giả là vì các thông tin, manh mối dùng để đánh giá một tin tức có thật hay không đều mang tính chất chủ quan. Nói cách khác chúng không chắc chắn, không hoàn toàn đúng, điều này xảy ra do nhiều tác nhân như cách diễn giải, bối cảnh, sắc thái ngôn ngữ và thành kiến cá nhân. Do tính chất mờ của thông tin, ngôn ngữ tự nhiên nên thích hợp dùng fuzzy logic để mô phỏng hơn là phương pháp phân lớp nhị phân (đùng/sai) truyền thống. Thêm vào đó fuzzy logic có thể giảm dữ liệu nhiễu đến một mức nào đó.

Ví dụ, nghiên cứu [*A probabilistic approach toward evaluation of Internet rumor on COVID*](https://doi.org/10.1007/s00500-022-07064-1) đã áp dụng fuzzy logic để phát hiện tin đồn trên Internet về COVID-19 bằng cách sử dụng các thông tin đặc trưng liên quan của tin đồn làm đầu vào cho hệ thống fuzzy logic. Điểm không tốt của hệ thống này là chỉ sử dụng fuzzy logic truyền thống, các hệ thống này bị giới hạn tính linh hoạt và hiệu suất trong việc phát hiện tin giả.

Còn các kỹ thuật học máy và học sâu truyền thống tuy đã đạt được nhiều thành công trong việc phát hiện tin tức giả nhưng chúng thường gặp khó khăn trong việc mô hình hóa tính chất mờ và không chắc chắn ở ngôn ngữ tự nhiên. Lấy cảm hứng từ những lợi ích tiềm năng của việc kết hợp fuzzy logic và học sâu, mô hình FDHN được đề xuất trong bài báo nhằm mục đích tận dụng những lợi ích đó để giúp phát hiện tin giả hiệu quả hơn - một thách thức quan trọng trong xã hội kỹ thuật số ngày nay.

# GIỚI THIỆU VỀ BỘ DỮ LIỆU

## Giới thiệu chung về bộ dữ liệu

* Tên: LIAR dataset
* Đường dẫn: <https://www.cs.ucsb.edu/~william/data/liar_dataset.zip>
* Là bộ dữ liệu chuẩn thực tế được dùng rộng rãi cho vấn đề phát hiện tin giả. Nó được giới thiệu trong bài báo [*Liar, Liar Pants on Fire: A New Benchmark Dataset for Fake News Detection*](https://doi.org/10.18653/v1/P17-2067) bởi tác giả William Yang Wang vào năm 2017. Bộ dữ liệu gồm 12.8K tuyên bố ngắn về các chủ đề liên quan đến chính trị nước Mỹ được gán nhãn thủ công bởi các chuyên gia kiểm định sự thật từ tổ chức PolitiFact cùng với các thông tin ngữ cảnh ứng với tuyên bố đó (như chủ đề, người tuyên bố, hoàn cảnh tuyên bố, … ).
* Kích thước
  + Số dòng: 12.836
  + Số cột: 14
  + Độ lớn: 2.9 MB

## Chi tiết bộ dữ liệu:

|  |  |  |  |
| --- | --- | --- | --- |
|  | **#** | **Cột** | **Ý nghĩa** |
|  | - | id | Định danh duy nhất cho mỗi tuyên bố |
|  | - | label | Nhãn phân lớp tuyên bố.  **Miền giá trị:** [ “Pants on fire”, “False”, “Barely-true”, “Half-true“, “Mostly-true”, “True”] |
| **Thông tin chính của tin tức** | *f1* | statement | Nội dung tuyên bố |
| **Ngữ cảnh chữ** | *f2* | subject | Chủ đề, lĩnh vực của tuyên bố |
| *f3* | speaker | Người hoặc tổ chức đưa ra tuyên bố |
| *f4* | job\_title | Chức danh của người phát ngôn |
| *f5* | state\_info | Bang mà người phát ngôn đến từ |
| *f6* | party\_affiliation | Đảng phái chính trị của người phát ngôn |
| **Ngữ cảnh số**  Số liệu mức độ sự thật các tuyên bố trước của người phát ngôn, phản ánh độ tin cậy của người nói | *f7* | barely\_true\_counts | Số lượng tuyên bố trước của người phát ngôn được gán nhãn tương ứng |
| *f8* | false\_counts |
| *f9* | half\_true\_counts |
| *f10* | mostly\_true\_counts |
| *f11* | pants\_on\_fire\_counts |
| **Ngữ cảnh chữ** | *f12* | context | Ngữ cảnh bổ sung, cung cấp thêm thông tin về hoàn cảnh của tuyên bố |

Ví dụ một mẫu dữ liệu:

A white text with black text

Description automatically generated

## Thống kê của bộ dữ liệu

Tác giả Wang chia bộ dữ liệu thành 3 phần theo tỉ lệ 8:1:1 lần lượt ứng với các tập train, tập validation và tập test.

A screenshot of a report

Description automatically generated

Lý do tác giả chọn bộ dữ liệu

* Dữ liệu phân bổ đều cho các mức độ sự thật của tuyên bố, mức độ này được phân thành 6 lớp: “Pants on fire”, “False”, “Barely-true”, “Half-true“, “Mostly-true”, “True”.
* Do có phân thành nhiều lớp nên nó phù hợp với các tình huống thực tế hơn so với các bộ dữ liệu chỉ phân thành 2 lớp duy nhất (đúng và sai)

# MÔ HÌNH THUẬT TOÁN

## Giới thiệu chung về phương pháp, kỹ thuật áp dụng

* Fuzzy Logic: là phương pháp đưa giá trị số về các khả năng trong khoảng đúng – sai, biến nó thành gần đúng, hơi đúng,.. thay vì chỉ có 0 – 1, đúng – sai tuyệt đối.

A diagram of fuzzy logic

Description automatically generated

* Mô hình Fuzzy Deep Hybrid Network (FDHN): là mô hình học giám sát dựa trên Fuzzy Logic. Cấu trúc mô hình gồm 3 phần: 1 kênh nội dung tin tức (News Text); 1 kênh ngữ cảnh chữ (Textual Context); 1 kênh ngữ cảnh số (Numerical Context). Fuzzy Layer, thành phần cốt lõi của mô hình FDHN, được đặt ở kênh ngữ cảnh số để áp dụng fuzzy logic lên biểu diễn đặc trưng (feature representation) có được từ các mô hình học sâu trong kênh này. Kết quả đầu ra ở tầng kế cuối trong mô hình FDHN là bốn biểu diễn đặc trưng khác nhau, chúng sẽ được nối tiếp lại với nhau trước khi đi vào tầng cuối cùng để cho kết quả phân lớp của dữ liệu tin tức đầu vào. Tóm tắt thông tin từng kênh:
  + **News Text:** kênh này nhận đầu vào là nội dung chính của tin tức (thuộc tính statement - *f1*) và thực hiện phân lớp văn bản bằng cách tách các các đặc trưng quan trọng để phân biệt tin thật/giả. Tác giả dùng mô hình [TextCNN](https://aclanthology.org/D14-1181/) cho việc này. Kết quả biểu diễn đặc trưng có được sẽ tiếp tục được xử lý ở thành phần downstream của mô hình FDHN
  + **Textual Conext:** kênh này nhận đầu vào là các thông tin ngữ cảnh chữ liên quan đến tin tức (các thuộc tính *f2, …, f6, f12*). Việc tách để xử lý riêng dữ liệu ngữ cảnh chữ và số giúp hạn chế việc mất mát thông tin trong lúc huấn luyện. Tác giả dùng một mô hình TextCNN riêng biệt giống trong kênh News Text để xử lý dữ liệu ngữ cảnh chữ. Kết quả biểu diễn đặc trưng của kênh này cũng sẽ được kết hợp với kết quả từ các kênh khác để tiếp tục xử lý ở thành phần downstream
  + **Numerical Context:** dùng module CNNBi-LSTM (một mô hình CNN theo sau bởi một [Bidrectional LSTM (Bi-LSTM)](https://doi.org/10.1109/78.650093)) để xử lý dữ liệu ngữ cảnh số (các thuộc tính *f7, …, f11*). Ngoài ra tác giả còn cài đặt một lớp Fuzzy Layer sau đó để tiếp tục xử lý kết quả biểu diễn đặc trưng đầu ra từ module CNNBi-LSTM. Kết quả cuối cùng của kênh ngữ cảnh số này bao gồm 2 biểu diễn đặc trưng từ CNNBi-LSTM và từ Fuzzy Layer. Các biểu diễn đặc trưng này sẽ được kết hợp cùng các biểu diễn đặc trưng ở các kênh khác để tiếp tục xử lý ở tầng cuối cùng của mô hình FDHN

A table of text with black text

Description automatically generated with medium confidence

Hình 1. Hình lấy 1 tin ngẫu nhiên từ LIAR để minh họa

A diagram of a structure

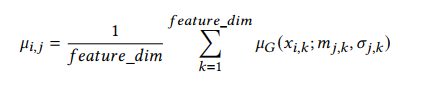
Description automatically generated

Hình 2. Hình minh họa hoạt động của mô hình với 1 tin

* Chú thích: conv1d là phép tích chập 1 chiều, Max pooling giảm kích thước, độ phức tạp mô hình, giữ đặc trưng quan trọng trong mạng nơ-ron học sâu.
* Đầu vào: các đặc trưng f1,...,f12 của bộ dữ liệu LIAR chia vào 3 kênh:
  + **News Text:** *f1* nội dung chính của tin tức.
  + **Textual Context:** *{f2, f3, f4, f5, f6, f12}* thông tin ngữ cảnh chữ liên quan đến tin tức: lĩnh vực, người phát biểu, nguồn, vị trí,...
  + **Numerical Context:** *{f7, f8, f9, f10, f11}* là 5 nhãn chân lý: đúng 1 nửa, gần đúng, ,... ghi lại lịch sử số lần được dán nhãn tương ứng của người đưa ra tin tức đó.
  + **Fuzzy Layer:** nhận và xử lý kết quả đầu ra của mô hình CNN cùng Bi-LSTM ở kênh số 3.
  + **Fuse Layer:** nhận 4 output từ 3 kênh (output 1 từ kênh số 1, output 2 từ kênh số 2 và ouput 3, 4 từ kênh số 3) để cho ra kết quả phân lớp cuối cùng.
* Đầu ra:
* Output 1: ma trận đối tượng x lớp
* Output 2: ma trận đối tượng x lớp
* Output 3: ma trận đối tượng x lớp
* Output 4: ma trận đối tượng x lớp
  + Fuse layer: mảng 2 chiều có 6 lớp (nhãn) có trọng số của đối tượng
* Công thức:
  + Dùng hàm thành viên Gaussian để định hình hàm, số lượng hàm thành viên = số chiều đặc trưng x số lượng lớp (số lượng tập mờ)



* + Như vậy, để tính ra giá trị hàm thành viên của mỗi đối tượng i ở mỗi lớp j, ta tính giá trị trung bình của hàm thành viên theo các chiều của đối tượng đó ở lớp j.



* Chú thích: (ma trận đối tượng x chiều) + (ma trận chiều x lớp) = mảng 3D (cube) => tính trung bình chiều = (ma trận đối tượng x lớp).

## Mô hình TextCNN

Việc xử lý hiệu quả dữ liệu văn bản là một thách thức lớn, các hướng tiếp cận truyền thống trong việc tiền xử lý dữ liệu chữ có thể kể đến là:

* **Stemming:** Quá trình loại bỏ tiền tố và hậu tố của từ để đưa nó về dạng gốc. Ví dụ từ “running” hay “runs” sẽ được chuyển về “run”. Kĩ thuật này làm giảm số lượng từ trong văn bản, giúp đơn giản việc xử lý và phân tích.
* **Removing punctuation and stop words:** Loại bỏ dấu câu như dấu chấm, dấu phẩy, … và các từ dừng như “and”, “the”, “is” vì chúng thường không mang nhiều ý nghĩa đặc biệt. Việc loại bỏ những thành phần này giúp giảm kích thước bộ dữ liệu cần xử lí và tập trung vào những từ quan trọng hơn trong văn bản.

Sau khi đã tiền xử lí thì tiếp theo là thực hiện **tokenisation**, là quá trình chia văn bản thành các đơn vị nhỏ hơn, thường là từ hoặc cụm từ. Mỗi đơn vị nhỏ này được gọi là một "token".

Tuy nhiên xu hướng của các phương pháp NLP dựa trên học sâu hiện nay là giữ nguyên vẹn thông tin văn bản, bao gồm cả ngữ cảnh và cấu trúc ngữ pháp để hiểu và xử lý văn bản một cách chính xác hơn và xây dựng các mô hình máy học từ đầu đến cuối. Cụ thể các mô hình này có thể được huấn luyện để thực hiện trực tiếp từ đầu vào đến đầu ra mà không cần các bước tiền xử lý và có thể được tùy chỉnh cho từng tác vụ cụ thể, như mô hình BERT hay GPT. Các mô hình này dùng các kĩ thuật NLP như mạng nơ-ron tích chập (CNN), cơ chế attention (attention mechanism) và mạng nơ-ron hồi quy (RNN) để nắm bắt được các quan hệ ngữ nghĩa phức tạp trong văn bản.

* **CNN:** Thường được biết đến với ứng dụng trong xử lý ảnh, nhưng cũng rất hiệu quả trong việc xử lý văn bản. Trong NLP, CNN có thể nhận diện được các mẫu cục bộ trong văn bản (ví dụ, các cụm từ hoặc cấu trúc ngữ pháp cụ thể) bằng cách sử dụng các bộ lọc để quét qua văn bản.
* **Attention Mechanisms:** Cho phép mô hình tập trung vào các phần quan trọng của văn bản khi xử lý, bỏ qua những thông tin nhiễu hoặc không quan trọng, từ đó nâng cao hiệu quả và độ chính xác của model.
* **RNN:** Được thiết kế để xử lý dữ liệu chuỗi, làm cho chúng thích hợp đối với dữ liệu văn bản, vốn là một chuỗi của các từ hoặc ký tự. RNN có khả năng nhớ thông tin từ các bước trước đó, giúp nó hiểu được ngữ cảnh và cấu trúc của văn bản theo thời gian.

Trong mô hình đề xuất, để thực hiện phân lớp văn bản bằng việc trích xuất các đặc trưng quan trọng dùng cho diện tin giả thì tác giả tận dụng mô hình [TextCNN](https://doi.org/10.3115/v1/D14-1181) được giới thiệu vào năm 2014, mô hình đã áp dụng CNN từ lĩnh vực thị giác máy tính vào việc xử lý ngôn ngữ tự nhiên. Hướng tiếp cận này đã cho thấy hiệu quả trong việc nắm bắt các đặc trưng, các mẫu quan trọng của dữ liệu chữ, dẫn đến tăng độ chính xác cho các ứng dụng thuộc NLP, bao gồm bài toán phân lớp văn bản.

### Mô hình TextCNN

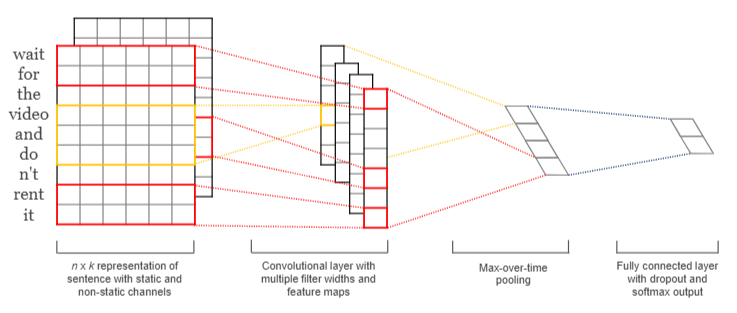
TextCNN là mô hình CNN đơn giản với một lớp tích chập, mô hình sử dụng các vector từ có được từ mô hình Word2vec. Cho là một vector từ k chiều ứng với từ thứ trong câu. Một câu có chiều dài là *n* (có thể có padding để cho câu có độ dài đúng với yêu cầu) được biểu diễn như sau:

Trong đó, ⊕ đơn giản là phép nối các vetor từ với nhau thành một ma trận kích thước , trong đó hàng thứ là vector từ . Tóm lại, ta có là kết quả việc nối các từ liên tiếp nhau . Phép toán tích chập được thực hiện bằng cách sử dùng một nhân lên một cửa sổ từ có độ rộng là , cho ra một kết quả feature. Cụ thể, kết quả feature có được từ cửa sổ từ biểu diễn bởi biểu thức sau:

Trong đó, là hệ số bias và là một hàm phi tuyến tính như hàm tanh (hyperbolic tangen). Nhân được thực hiện lên lần lượt các cửa sổ từ độ rộng trong câu có từ cho ra kết quả là một feature map

Với . Tiếp đó cho feature map qua một tầng gộp cực đại theo thời gian (max-over-time pooling), giới thiệu trong bởi Collobert vào năm 2011, để lấy được giá trị feature lớn nhất trong feature map, cụ thể là , là kết quả feature có được ứng với một nhân cụ thể.

Trên là mô tả cho quy trình lấy một feature ứng với một nhân cụ thể. Mô hình TextCNN sử dùng nhiều nhân có kích thước khác nhau để lấy được nhiều giá trị feature tương ứng. Cuối cùng các giá trị feature này đi qua một tầng softmax kết nối đầy đủ để cho ra các phân phối xác suất ứng với nhãn, lớp để quyết định kết quả phân lớp cho câu (sentence).



Kiến trúc mô hình với 2 kênh và sử dụng 2 nhân có kích thước 2 và 3

Đối với kiến trúc đa kênh thì mỗi nhân trong mô hình đều được thực hiện trên tất cả các kênh.

### Chính quy hoá mô hình TextCNN

Để chính quy hoá mô hình, thực hiện dropout trên lớp kế cuối của mô hình (penultimate layer), là một vector chứa các kết quả feature ứng với các nhân, kèm theo ràng buộc về l2-norm của các vector trọng số. Cho lớp kế cuối được biểu diễn bởi (giả sử sử dụng nhân) thì dropout sử dụng biểu thức sau để tính toán kết quả trong quá trình lan truyền (forward propagation):

Trong đó:

* là vector trọng số
* là đầu vào từ lớp penultimate
* là phép nhân element-wise
* là vector mask (vector chứa các biến ngẫu nhiên Bernoulli) với xác suất p để mỗi phần tử là 1

Ở thời điểm huấn luyện, gradient chỉ được lan truyền ngược qua các đơn vị không được mask (unmasked units). Tại thời điểm kiểm tra, các vector trọng số học được sẽ được điều chỉnh bằng cách nhân với sao cho , được sử dụng (không dropout) để đánh giá các câu chưa từng thấy. Thêm ràng buộc l2-norm của các vector trọng số bằng cách điều chỉnh lại sao cho khi sau mỗi bước tối ưu gradient descent.

# THỰC NGHIỆM MÔ HÌNH VÀ ĐÁNH GIÁ KẾT QUẢ

## Cấu hình mô hình nhóm tác giả sử dụng

Độ dài chuỗi đầu ra của mỗi module được đặt là 6, tỷ lệ dropout được đặt là 0.5, và số chiều của embedding là 128, với zero-padding khi cần thiết.

Module TextCNN, có trách nhiệm xử lý văn bản tin tức và ngữ cảnh văn bản của tin tức, bao gồm một lớp embedding tiếp theo là ba lớp CNN song song. Các lớp CNN có kích thước nhân là (3, 4, 5) với số chiều là 128. Đầu ra của mỗi lớp CNN sau đó được đi qua một lớp MaxPooling để lấy giá trị lớn nhất cho mỗi feature map. Các feature map kết quả sau đó được nối lại (concatenated) và đưa vào một lớp linear có dropout.

Module CNNBiLSTM có trách nhiệm xử lý ngữ cảnh số. Trước tiên, nó đi qua một lớp linear có dropout, tiếp theo là một lớp CNN có số kênh đầu ra là 32 và kích thước kernel là 1. Đầu ra sau đó được đưa vào mạng LSTM hai chiều (Bi-LSTM) có dropout và cuối cùng được đưa vào một lớp linear khác để tính toán đầu ra.

## Chuẩn bị công cụ

* Sao chép code từ trang github của bài báo về.
* Cài đặt lại phiên bản ngôn ngữ Python là 3.10.0a1 để phù hợp và thực hiện chính xác lại thực nghiệm của bài báo.
* Cài đặt các thư viện theo như yêu cầu bài báo trong tệp requirements.txt là:

+ torch

+ pandas

+ numpy

+ time

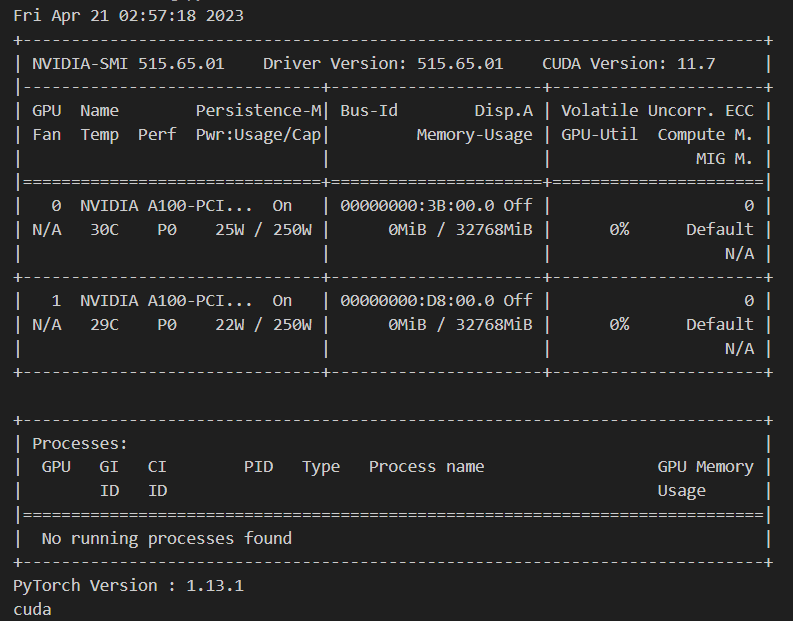
+ sklearn

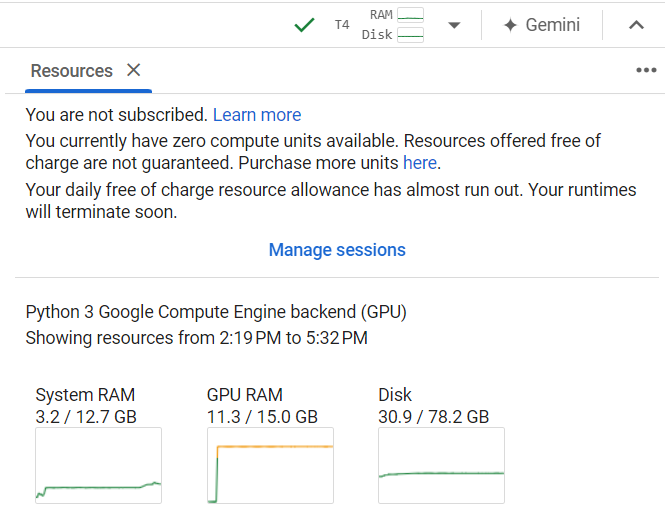
+ transformers

* 3 tệp train.tsv, valid.tsv, test.tsv của bộ dữ liệu LIAR.
* Máy tính thực nghiệm khuyến nghị có GPU và CPU chạy tốt.

## Huấn luyện mô hình

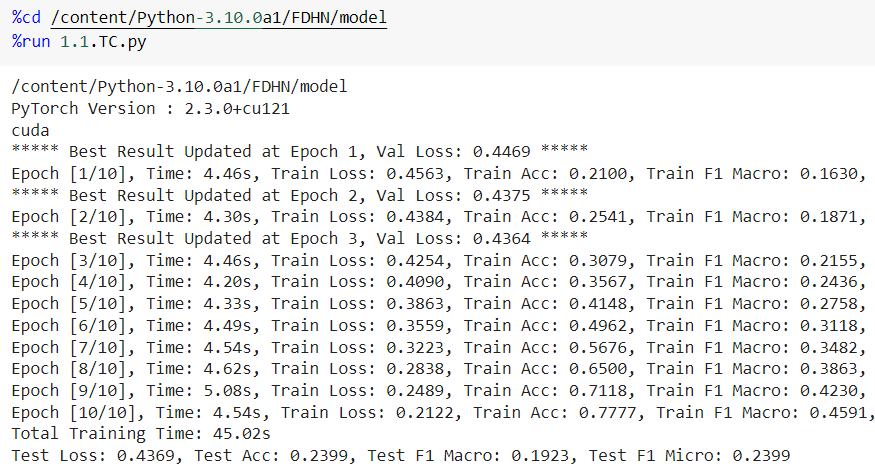
* Do cấu hình máy tính nhóm yếu hơn so với cấu hình máy tính của bài báo nên thời gian thực nghiệm các mô hình sẽ dài hơn so với bài báo:

Cấu hình máy tính của nhóm tác giả bài báo

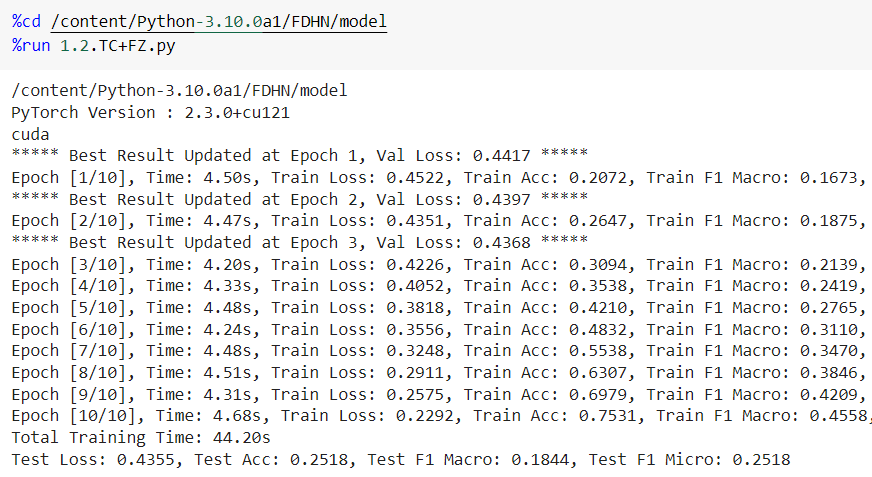
Cấu hình máy tính của nhóm chạy với Google Colab

* Hình ảnh các quá trình và kết quả thực hiện lại các mô hình:

+ 1.1. TextCNN



+ 1.2. TextCNN + Fuzzy



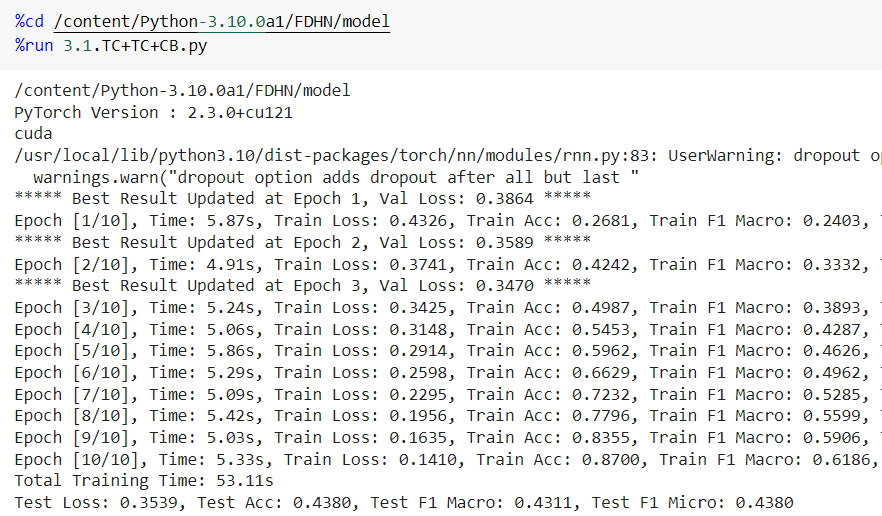
+ 2.1. BERT + TextCNN + CNNBiLSTM



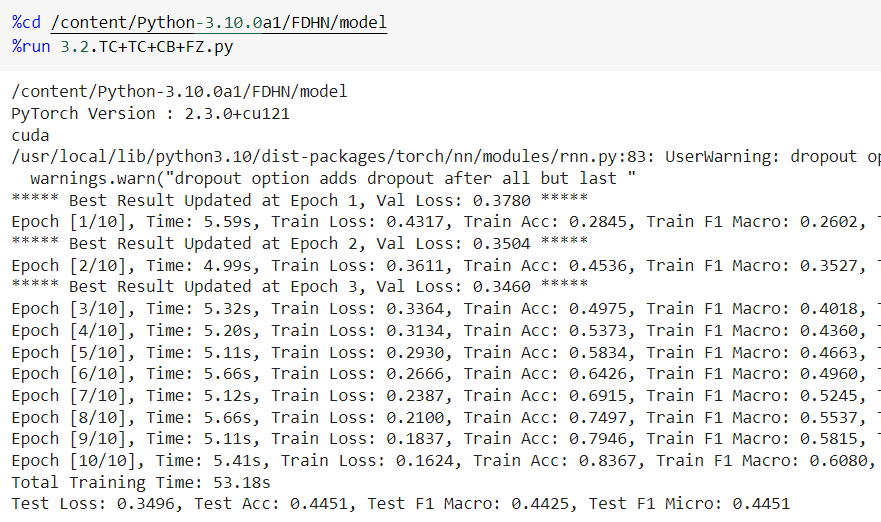
+ 2.2. BERT + TextCNN + CNNBiLSTM + Fuzzy



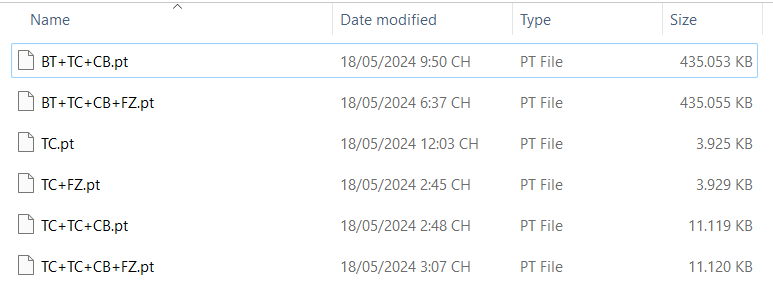
+ 3.1. TextCNN + TextCNN + CNNBiLSTM



+ 3.2. TextCNN + TextCNN + CNNBiLSTM + Fuzzy



+ Các tệp pytorch lưu lại các mô hình đã được huấn luyện và dùng cho dự đoán hoặc phát triển thêm sau này:



* Tổng kết thời gian chạy các mô hình:

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên mô hình** | **Thời gian chạy** |
| **1** | 1.1. TC | 45.02 giây |
| **2** | 1.2. TC+FZ | 44.20 giây |
| **3** | 2.1. BT+TC+CB | 2 giờ 34 phút 33.27 giây |
| **4** | 2.2. BT+TC+CB+FZ | 2 giờ 37 phút 12.59 giây |
| **5** | 3.1. TC+TC+CB | 53.11 giây |
| **6** | 3.2. TC+TC+CB+FZ | 53.18 giây |

* So sánh kết quả chạy lại mô hình so với bài báo (epoch tốt nhất sẽ được chọn)

+ Kết quả của bài báo:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Mô hình** | **Validation** | | | **Test** | | | **Trung bình thời gian chạy 1 epoch (Giây)** |
| **Accuracy** | **F1-Macro** | **F1-Micro** | **Accuracy** | **F1-Macro** | **F1-Micro** |
| **TC** | 0.2516 | 0.1774 | 0.2321 | 0.2431 | 0.1933 | 0.2431 | 3.096 |
| **TC + FZ** | 2.5223 | 0.1548 | 0.2481 | 0.2463 | 0.1446 | 0.2463 | 3.179 |
| **BT + TC + CB** | 0.4338 | 0.4214 | 0.4208 | 0.4081 | 0.4152 | 0.4081 | 233.941 |
| **BT + TC + CB + FZ** | 0.4439 | 0.4293 | 0.4383 | 0.4238 | 0.4253 | 0.4238 | 234.190 |
| **TC + TC + CB** | 0.4587 | 0.4416 | 0.4410 | 0.4365 | 0.4443 | 0.4365 | 3.779 |
| **TC + TC + CB + FZ** | 0.4673 | 0.4490 | 0.4577 | 0.4649 | 0.4701 | 0.4649 | 3.944 |

+ Kết quả của mô hình được chạy lại của nhóm:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Mô hình** | **Validation** | | | **Test** | | | **Trung bình thời gian chạy 1 epoch (Giây)** |
| **Accuracy** | **F1-Macro** | **F1-Micro** | **Accuracy** | **F1-Macro** | **F1-Micro** |
| **TC** | 0.2438 | 0.1765 | 0.2274 | 0.2399 | 0.1923 | 0.2399 | 4.502 |
| **TC + FZ** | 0.2484 | 0.1756 | 0.2422 | 0.2518 | 0.1844 | 0.2518 | 4.420 |
| **BT + TC + CB** | 0.4260 | 0.3965 | 0.4024 | 0.4199 | 0.4269 | 0.4199 | 927.327 |
| **BT + TC + CB + FZ** | 0.4525 | 0.4098 | 0.4171 | 0.4294 | 0.4393 | 0.4294 | 943.259 |
| **TC + TC + CB** | 0.4603 | 0.4202 | 0.4372 | 0.4380 | 0.4311 | 0.4380 | 5.311 |
| **TC + TC + CB + FZ** | 0.4572 | 0.4327 | 0.4463 | 0.4451 | 0.4425 | 0.4451 | 5.318 |

+ Kết quả sau cùng sau khi thực hiện lại các mô hình thì tuy nhóm chúng em khác với nhóm tác giả bài báo khi mà mô hình TextCNN + TextCNN + CNNBiLSTM với Validation Accuracy = 0.4603 (là cao nhất trong toàn bộ mô hình) nhưng xét đến hiệu suất mô hình ở các chỉ số F1 macro, F1 micro và cả Test Accuracy thì mô hình TextCNN + TextCNN + CNNBiLSTM + Fuzzy là cao nhất trong tất cả dẫn đến mô hình này là tốt nhất.

+ Và thứ tự các mô hình tốt nhất đến kém nhất của nhóm em là:

**3.2. TC + TC + CB + FZ**

> 3.1. TC + TC + CB

> 2.2. BT + TC + CB + **FZ**

> 2.1. BT + TC + CB

> 1.2. TC + **FZ**

> 1.1. TC

Thứ tự hoàn toàn khớp với nhóm tác giả.

+ Nhìn chung, việc kết hợp thêm Fuzzy Layer vào mô hình đã giúp mô hình chạy tốt hơn và việc thay thế mô hình BERT với TextCNN đầu tiên của mô hình 3.1. TC + TC + CB nhằm xử lý news text không mang lại kết quả khả quan, thậm chí là kém hơn (hoàn toàn giống với kết luận của nhóm tác giả bài báo).

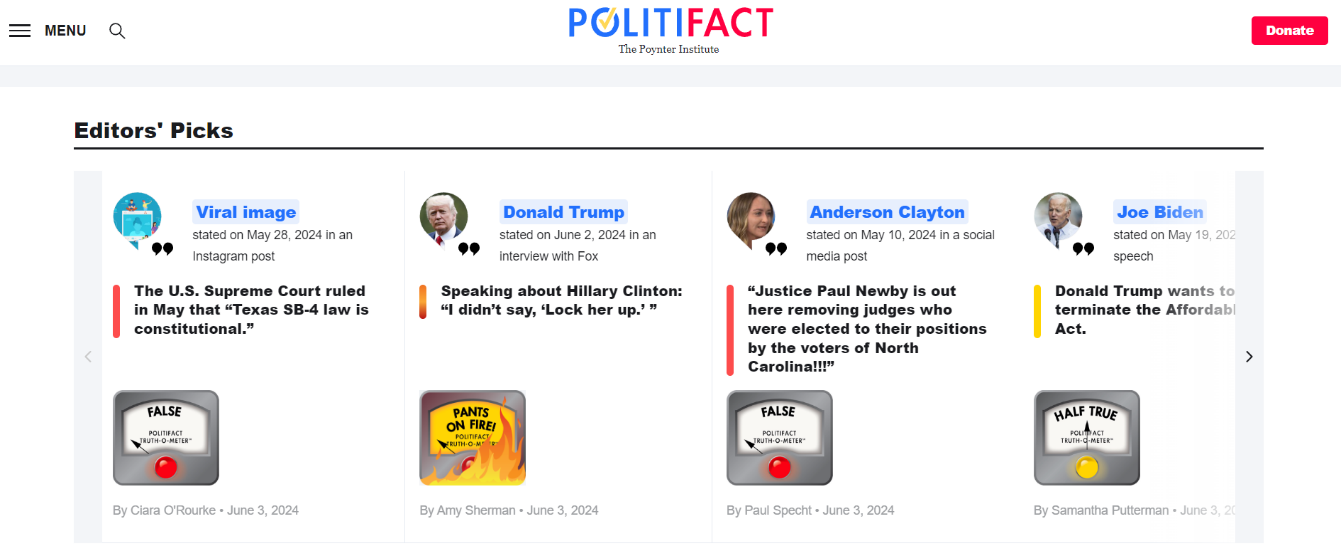
+ Về mặt tổng quát, kết quả sau khi chạy lại các mô hình thì có phần kém hơn so với nhóm tác giả khoảng 0.1 đến 0.2. Thời gian chạy lại cũng lâu hơn gấp 4 đến 5 lần so với nhóm tác giả (do vấn đề thiết bị).

+ Do các mô hình của bài báo còn ở dạng cơ bản (như mô hình TC - TextCNN) và phải xử lý văn bản ở mức khá phức tạp (nhiều lớp) nên các chỉ số đánh giá F1 micro và F1 macro đều ở mức 0.4 thì chấp nhận được. Khi xử lý với mô hình phù hợp hơn thì F1 micro và F1 macro đều tăng cùng với Accuracy. Và nhóm em nhận thấy F1 micro và F1 macro luôn gần bằng nhau trong cùng 1 mô hình, chứng tỏ các mô hình đều có hiệu suất khá đồng đều ở 6 lớp khác nhau.

**5.3. Thử nghiệm dự đoán trên bộ dữ liệu mới**

* Nguồn bộ dữ liệu

+ Thực hiện cào dữ liệu (crawl data) từ trang Politifact (nguồn của bộ dữ liệu LIAR trước đó)



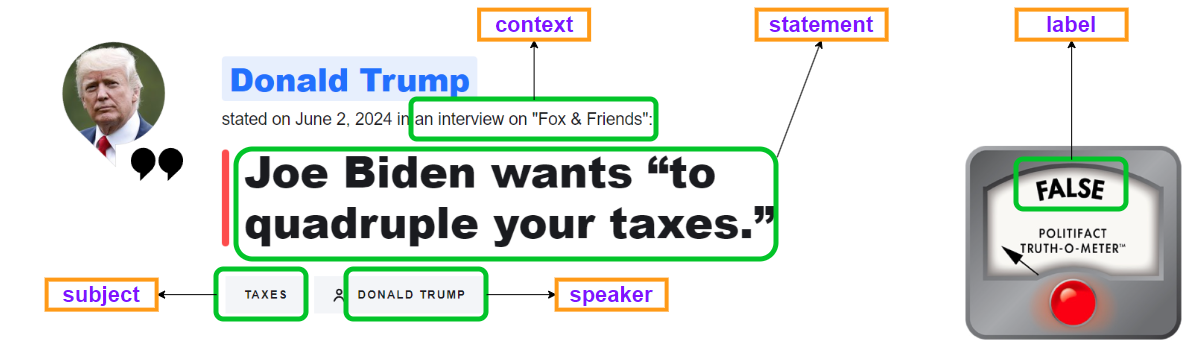
* Tiền xử lý dữ liệu:

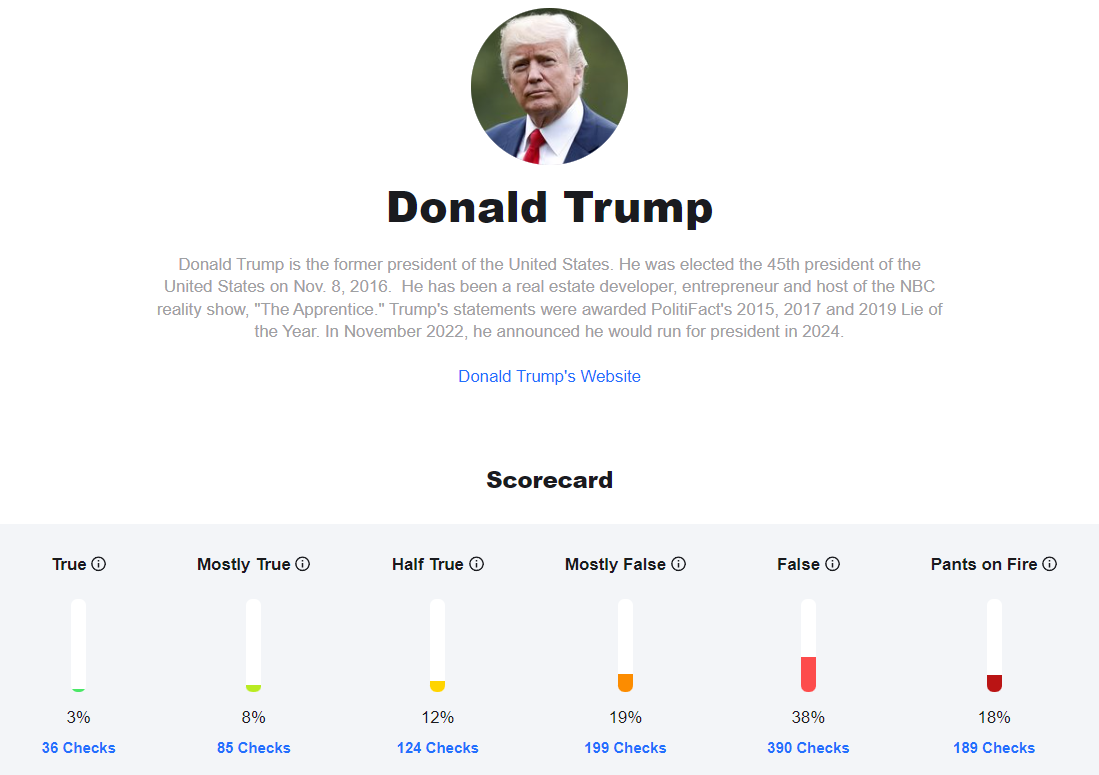
+ Dữ liệu đã được gán nhãn sẵn bởi Politifact.

+ Thực hiện tạo 14 cột thuộc tính cho dữ liệu:

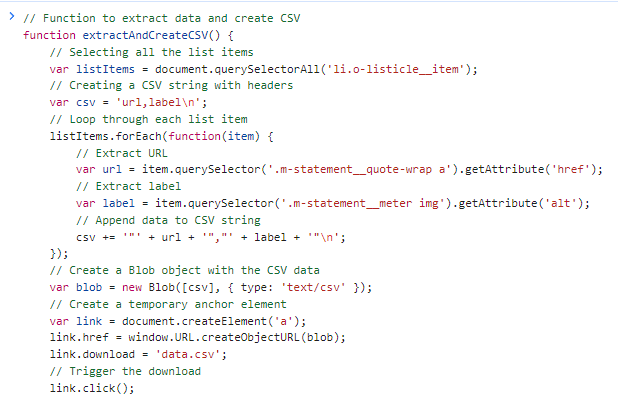
|  |  |  |
| --- | --- | --- |
| **STT** | **Tên thuộc tính** | **Cách xử lý** |
| **1** | id | Mã của mỗi dòng dữ liệu là độc nhất và là tên tệp json. |
| **2** | label | Nhãn đã được dán bởi Politifact |
| **3** | statement | Nội dung chính của tin tức, thường là tiêu đề. |
| **4** | subject | Lĩnh vực của nội dung tin tức, mỗi tin tức sẽ được gắn với các lĩnh vực phù hợp |
| **5** | speaker | Nhân vật chính hoặc đối tượng tạo nên tin tức (thường tên chính trị gia). |
| **6** | job\_title | Chức vụ, vị trí của người phát ngôn của tin. (thường không đổi, dữ liệu có thể lấy từ bộ dữ liệu cũ ). |
| **7** | state\_info | Bang mà người phát ngôn đang hoạt động. (thông tin có thể lấy từ bộ dữ liệu cũ) |
| **8** | party\_affiliation | Đảng phái của người phát ngôn. (thông tin có thể lấy từ bộ dữ liệu cũ) |
| **9** | barely\_true\_counts | Số lần người phát ngôn được dán nhãn barely\_true trước đó. (dựa vào dữ liệu của người này trong tập dữ liệu đã dùng để huấn luyện) |
| **10** | false\_counts | Dữ liệu lấy từ tập dữ liệu cũ. |
| **11** | half\_true\_counts | Dữ liệu lấy từ tập dữ liệu cũ. |
| **12** | mostly\_true\_counts | Dữ liệu lấy từ tập dữ liệu cũ. |
| **13** | pants\_on\_fire\_counts | Dữ liệu lấy từ tập dữ liệu cũ. |
| **14** | context | Nguồn tin lấy được tin tức này (dữ liệu này cũng có ở tin tức tại Politifact như “An interview on “Fox & Friends” ”) |

+ Một số hình ảnh minh họa về nguồn dữ liệu:





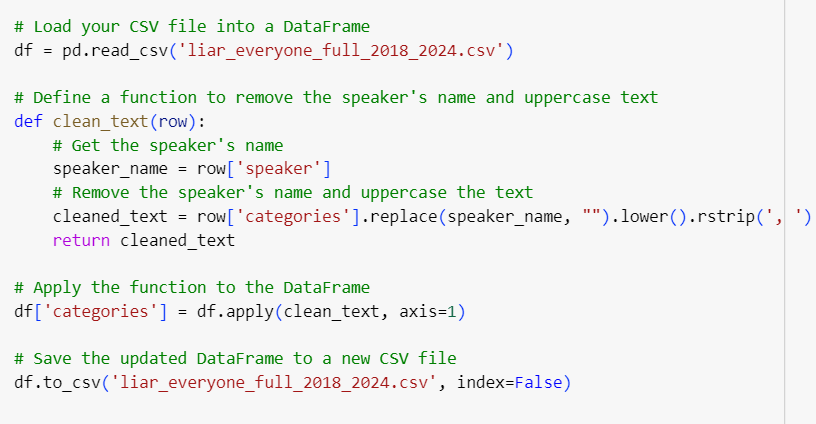
+ Cào tất cả url bài phát ngôn bằng lệnh Javascript

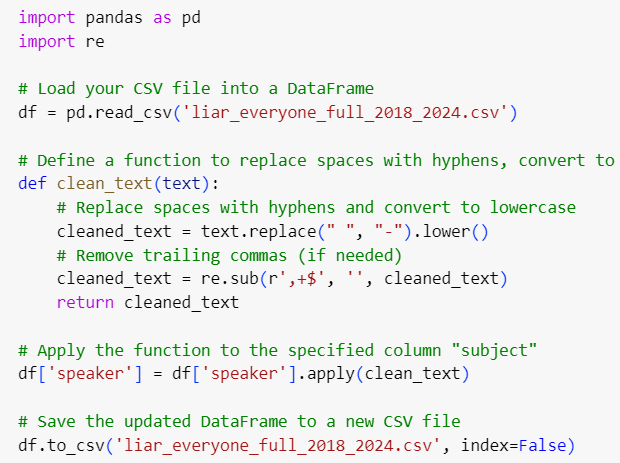


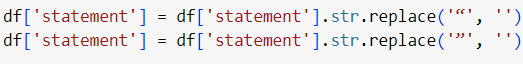
+ Cào dữ liệu từ từng url bài phát biểu trong trang Politifact với hàm BeautifulShop của thư viện bs4

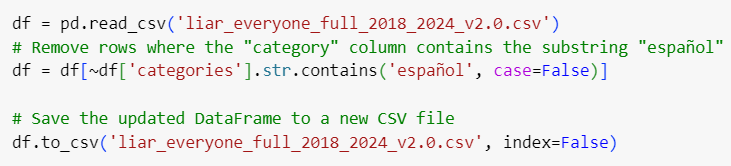


+ Chuẩn hóa văn bản

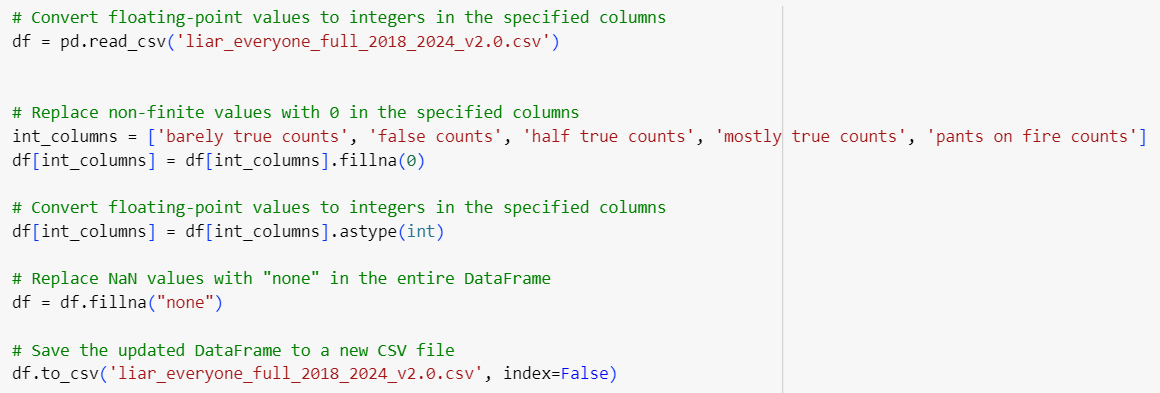




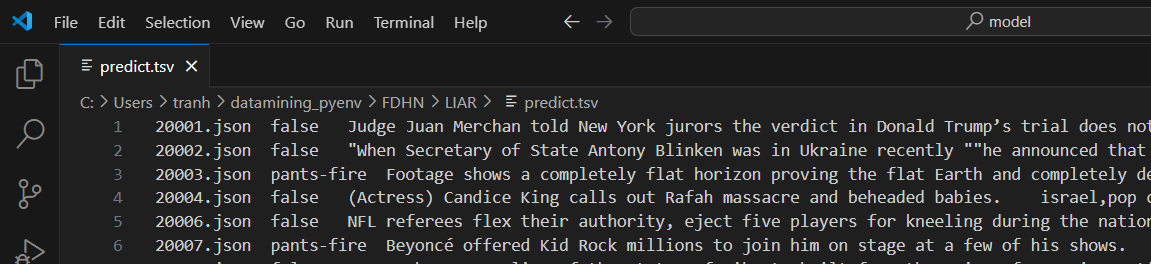




+ Ánh xạ dữ liệu cũ và điền khuyết dữ liệu

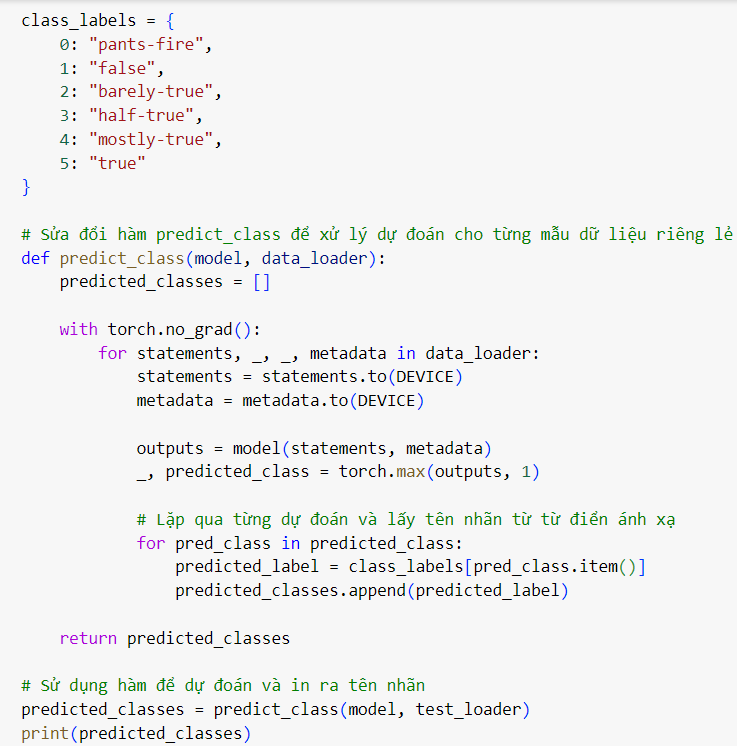


+ Chuyển đổi sang dạng tệp tsv và lưu tệp thành công

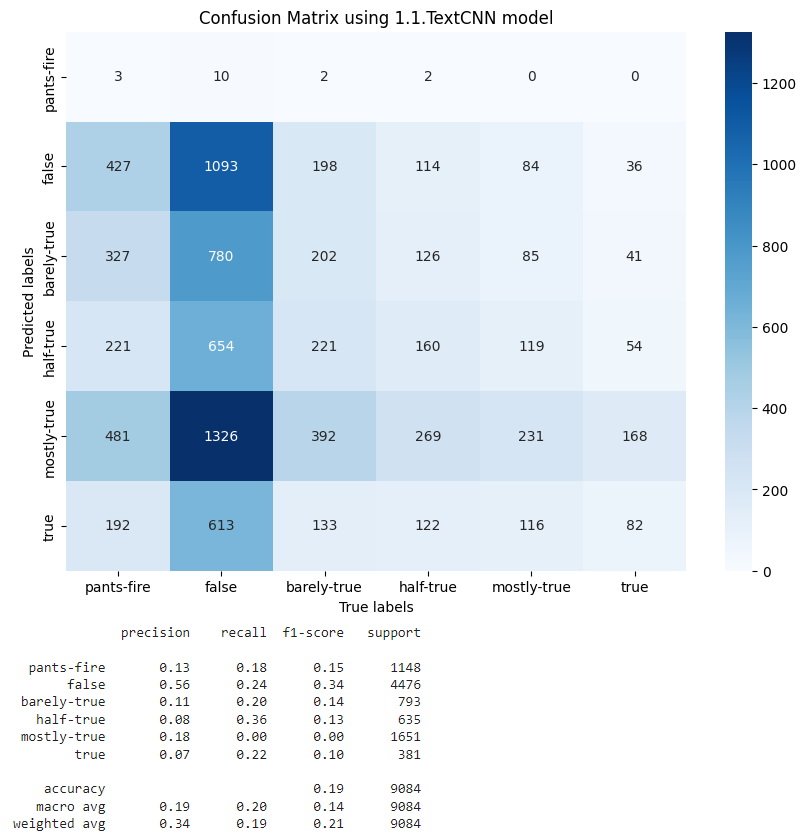


* Kết quả dự đoán với các mô hình đã huấn luyện:

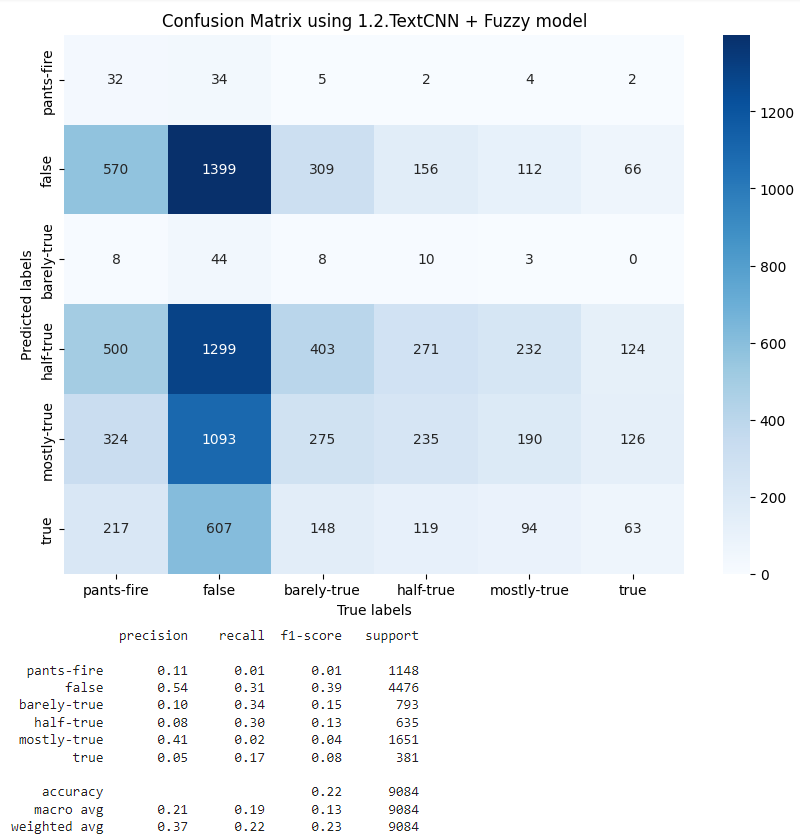
Hình ảnh code đọc và dự đoán mô hình đã huấn luyện



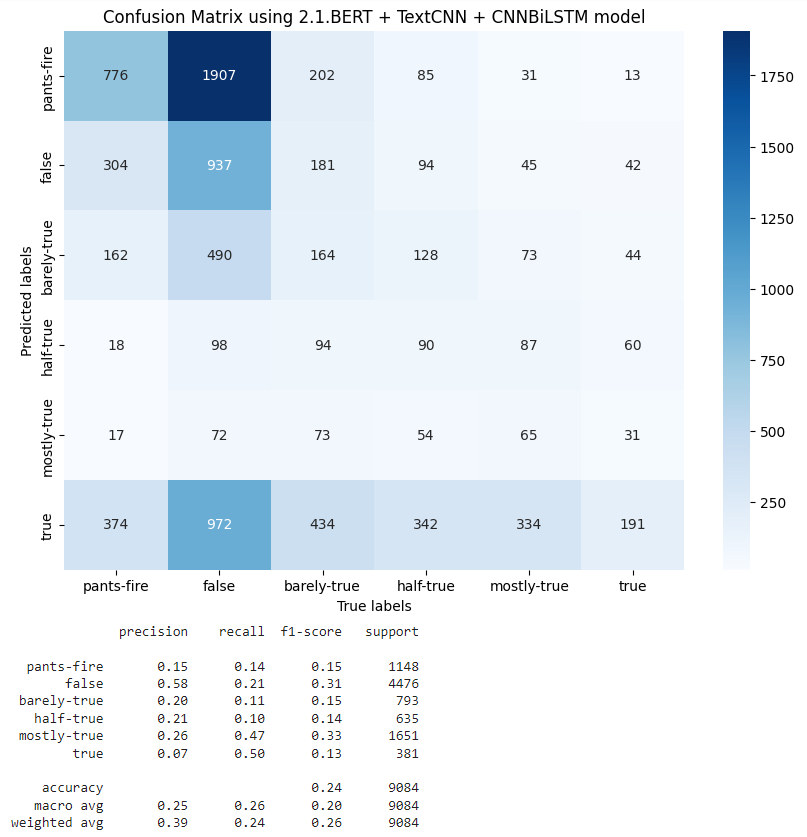
+ Với mô hình 1.1.TextCNN:



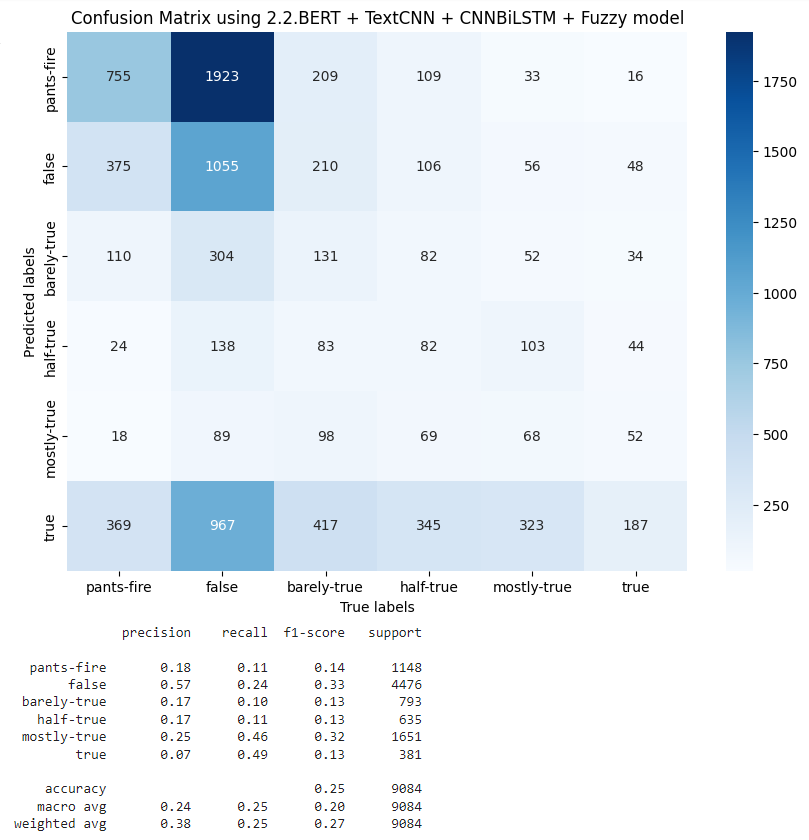
+ Với mô hình 1.2.TextCNN + Fuzzy:



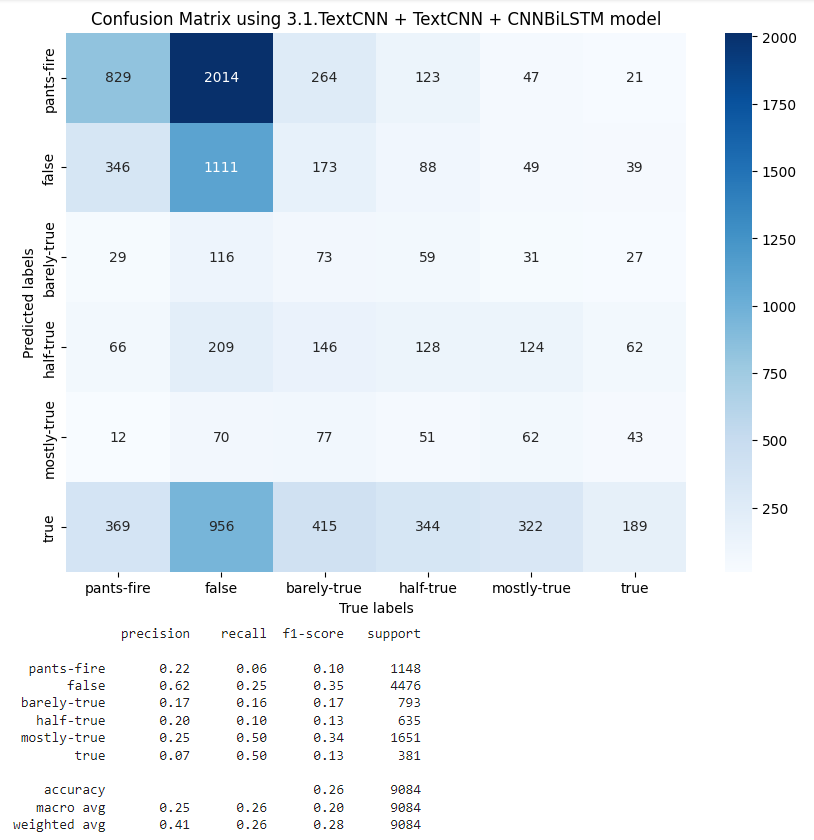
+ Với mô hình 2.1.BERT + TextCNN + CNNBiLSTM:



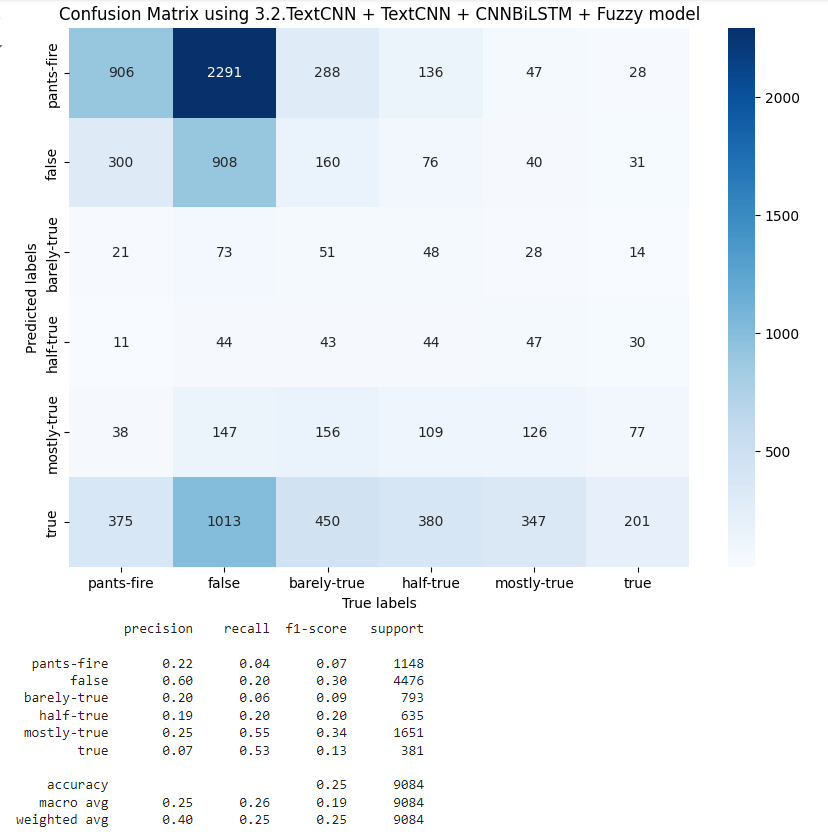
+ Với mô hình 2.2.BERT + TextCNN + CNNBiLSTM + Fuzzy:



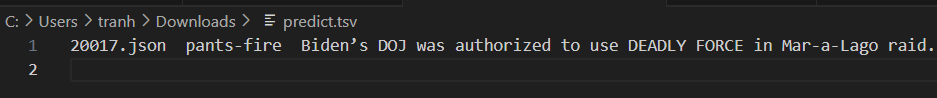
+ Với mô hình 3.1.TextCNN + TextCNN + CNNBiLSTM:



+ Với mô hình 3.2.TextCNN + TextCNN + CNNBiLSTM:

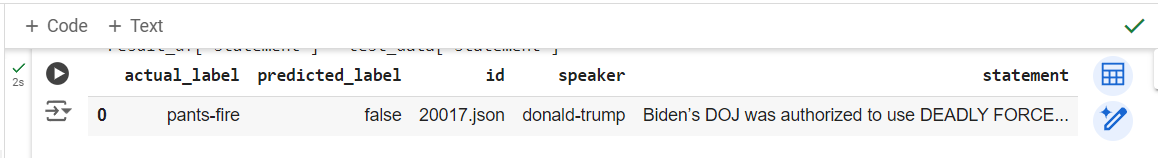


+ Dự đoán 1 nhãn khi đầu vào là một tin tức bất kỳ với mô hình 3.3.TC+TC+CB+FZ

Hình ảnh tin tức được dự đoán

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thuộc tính** | **Nội dung** | **Tên thuộc tính** | **Nội dung** |
| **id** | 20017.json | **party\_affiliation** | republican |
| **label** | pants-fire | **barely\_true\_counts** | 63 |
| **statement** | Biden’s DOJ was authorized to use DEADLY FORCE in Mar-a-Lago raid. Biden was locked and loaded and ready to take me out. | **false\_counts** | 114 |
| **subject** | national,elections,legal issues | **half\_true\_counts** | 51 |
| **speaker** | donald-trump | **mostly\_true\_counts** | 37 |
| **job\_title** | President-Elect | **pants\_on\_fire\_counts** | 61 |
| **state\_info** | New York | **context** | a campain email |

Bảng nội dung một tin tức làm đầu vào để dự đoán

Hình ảnh kết quả nhãn được dự đoán là false

* Đánh giá kết quả dự đoán với bộ dữ liệu mới:

+ Nhãn được dự đoán chính xác nhiều nhất là nhãn “false” khi luôn ở mức 0.6, ít nhất (kém nhất) là những dự đoán về nhãn “true” ở mức 0.07.

+ Các mô hình có kết hợp fuzzy có Accuracy cao hơn khi không kết hợp. Và với mô hình càng phù hợp (theo đánh giá kết quả huấn luyện) thì độ chính xác dự đoán càng cao.

+ Khi xem xét đến đánh giá hiệu suất, chỉ số có trọng số (weighted avg) luôn cao hơn không trọng số (macro avg), chứng tỏ hiệu suất mô hình có chịu ảnh hưởng của kích thước mẫu của các lớp được dự đoán, nhãn false có kích thước lớn nhất và chiếm khoảng 50% tổng kích thước mẫu dự đoán, đồng thời cũng là nhãn có mức dự đoán chính xác cao nhất nên đã làm cải hiện weighted avg.

+ Tuy được dự đoán từ mô hình đã được huấn luyện nhưng chỉ số macro lại thấp hơn đáng kể so với kết quả huấn luyện.

# ƯU ĐIỂM, HẠN CHẾ VÀ HƯỚNG PHÁT TRIỂN

## Ưu điểm

Báo cáo này giới thiệu rõ ràng các nội dung chính của bài báo, bao gồm cơ chế hoạt động của mô hình và các công thức liên quan. Bài báo tập trung vào việc phân tích và thực nghiệm mô hình mới, giải thích chi tiết các bước triển khai và các khía cạnh kỹ thuật. Điều này giúp người đọc dễ dàng hiểu được phương pháp luận và các yếu tố quan trọng trong quá trình phát triển mô hình.

Trong quá trình thực nghiệm, nhóm đã thành công trong việc chạy mô hình và dự đoán kết quả. Việc thực nghiệm này đã được tiến hành cẩn thận, từ việc cài đặt mô hình cho đến quá trình dự đoán, đảm bảo rằng mọi khía cạnh của mô hình đều được kiểm tra và xác minh. Kết quả dự đoán của mô hình được nhóm đánh giá là chính xác và đáp ứng được các tiêu chí đã đề ra.

Nhóm đã tiến hành so sánh kết quả của bài báo với kết quả thu được từ các thử nghiệm của chính mình. Qua đó, nhóm đã đưa ra nhận xét chi tiết về sự khác biệt và những điểm tương đồng giữa hai bộ kết quả. Sự so sánh này giúp làm rõ hơn về hiệu quả và độ chính xác của mô hình, đồng thời cung cấp cái nhìn sâu sắc về khả năng ứng dụng của nó trong các tình huống khác nhau.

Một điểm nổi bật trong báo cáo là nhóm đã tự cài đặt mã nguồn và tiến hành thu thập (crawl) bộ dữ liệu mới. Quá trình này đòi hỏi sự tỉ mỉ và kỹ năng kỹ thuật cao, nhằm đảm bảo rằng dữ liệu thu thập được phù hợp và đủ chất lượng để áp dụng vào mô hình. Bộ dữ liệu mới thu thập được sau đó đã được ứng dụng vào mô hình để tiến hành dự đoán. Kết quả từ việc sử dụng bộ dữ liệu mới cho thấy mô hình không chỉ hoạt động tốt với dữ liệu gốc mà còn có khả năng thích ứng và dự đoán chính xác với các dữ liệu mới. Điều này khẳng định tính linh hoạt và hiệu quả của mô hình trong việc xử lý các loại dữ liệu khác nhau, mở ra nhiều cơ hội ứng dụng thực tiễn trong tương lai.

## Hạn chế

Một hạn chế đáng chú ý trong quá trình thực hiện báo cáo là nhóm chưa thể cài đặt và chạy mô hình ở mức cấu hình tốt nhất do hạn chế về phần cứng của laptop các thành viên. Việc thiếu cấu hình phần cứng mạnh mẽ đã ảnh hưởng đáng kể đến hiệu suất và tốc độ xử lý của mô hình, từ đó có thể làm giảm độ chính xác và hiệu quả của các dự đoán.

Cụ thể, mô hình yêu cầu khả năng xử lý cao và dung lượng bộ nhớ lớn để hoạt động tối ưu, trong khi các máy tính cá nhân của nhóm không đủ khả năng đáp ứng các yêu cầu này. Điều này dẫn đến việc nhóm phải thực cấu hình một số phần trong mô hình, gây ảnh hưởng đến kết quả cuối cùng. Hi vọng để đạt được kết quả tốt hơn trong tương lai, nhóm cần có sự hỗ trợ từ các thiết bị phần cứng mạnh mẽ hơn hoặc sử dụng các dịch vụ điện toán đám mây có khả năng xử lý cao.

## Hướng phát triển

Dựa trên những ưu điểm và nhược điểm đã được phân tích, hướng phát triển trong tương lai của nghiên cứu này sẽ tập trung vào việc tối ưu hóa mô hình và khắc phục những hạn chế về phần cứng để nâng cao hiệu suất và độ chính xác của kết quả dự đoán.

Trước hết, nhóm sẽ tìm cách cải thiện cấu hình phần cứng bằng cách sử dụng các máy tính có hiệu năng cao hơn hoặc tận dụng các dịch vụ điện toán đám mây. Điều này sẽ giúp mô hình hoạt động ở mức cấu hình tốt nhất, giảm thời gian xử lý và tăng cường độ chính xác của các dự đoán. Sự đầu tư vào phần cứng mạnh mẽ hơn sẽ cho phép nhóm thực hiện nhiều thử nghiệm và tối ưu hóa mô hình một cách hiệu quả hơn.

Ngoài ra, nhóm cũng sẽ tiếp tục phát triển và tối ưu hóa mã nguồn của mô hình. Điều này bao gồm việc cải thiện các thuật toán hiện tại, thử nghiệm với các cấu hình và tham số khác nhau, cũng như áp dụng các kỹ thuật mới trong lĩnh vực học máy và trí tuệ nhân tạo. Bằng cách này, mô hình có thể được cải thiện không chỉ về mặt hiệu năng mà còn về khả năng dự đoán chính xác trên nhiều loại dữ liệu khác nhau.

Một hướng phát triển quan trọng khác là tiếp tục mở rộng và đa dạng hóa bộ dữ liệu. Việc thu thập thêm các bộ dữ liệu mới từ nhiều nguồn khác nhau sẽ giúp mô hình trở nên linh hoạt hơn và có khả năng dự đoán chính xác trong nhiều ngữ cảnh khác nhau. Nhóm sẽ tập trung vào việc phát triển các phương pháp thu thập và tiền xử lý dữ liệu hiệu quả, đảm bảo rằng dữ liệu mới luôn đạt chất lượng cao và phù hợp với yêu cầu của mô hình.

Cuối cùng, nhóm sẽ tiến hành nghiên cứu và phát triển các ứng dụng thực tiễn của mô hình trong các lĩnh vực khác nhau. Điều này không chỉ giúp kiểm chứng tính ứng dụng của mô hình mà còn mở ra nhiều cơ hội để mô hình được sử dụng rộng rãi hơn trong thực tế.

# PHÂN CÔNG CÔNG VIỆC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Họ và tên | Vai trò | Nhiệm vụ | Đánh giá |
| 1 | Hồ Văn Vinh – 21520530 | Nhóm trưởng | * Báo cáo:   + Lập sườn báo cáo, viết nội dung giới thiệu bài báo, format file báo cáo  + Hỗ trợ tìm hiểu mô hình thuật toán   * Chương trình demo:   + Crawl lấy bộ dữ liệu mới từ Politifact từ năm 2018-2024  + Tiền xử lý dữ liệu bộ dữ liệu mới để phù hợp với input của mô hình thuật toán   * Hoạt động khác:   + Quản lý nhóm, phân công công việc  + Báo cáo đồ án bộ dữ liệu  + Hỗ trợ tìm bài báo. | *9.5/10* |
| 2 | Trần Minh Hoàng - 21522101 | Thành viên | * Báo cáo:   + Viết nội dung thực nghiệm lại chương trình, dự đoán và đánh giá kết quả.  + Hỗ trợ tìm hiểu và viết các phương pháp và kỹ thuật.   * Chương trình demo:   + Tìm hiểu và cấu hình lại tham số mô hình cho phù hợp.  + Thực hiện lại code các mô hình.  + Tạo và dùng lệnh dự đoán mô hình đã học.   * Hoạt động khác:   + Báo cáo đồ án phần demo.  + Hỗ trợ tìm bài báo. | *10/10* |
| 3 | Nguyễn Lý Đăng Khoa - 21522229 | Thành viên | * Báo cáo:   + Viết nội dung giới thiệu bài báo, giới thiệu chung bài toán, hướng tiếp cận, phân tích ưu điểm, hạn chế từ kết quả mô hình.  + Hỗ trợ tìm hiểu mô hình thuật toán.   * Chương trình demo:   + Chạy test code bài báo trong quá trình tìm kiếm bài báo phù hợp   * Hoạt động khác:   + Format file báo cáo  + Báo cáo đồ án mở đầu  + Hỗ trợ tìm bài báo | *9.5/10* |
| 4 | Trương Công Quốc Triệu - 21522714 | Thành viên | * Báo cáo:   + Viết nội dung giới thiệu bộ dữ liệu, mô hình thuật toán  + Tìm hiểu mô hình thuật toán chính.   * Chương trình demo:   + Chạy test code bài báo trong quá trình tìm kiếm bài báo phù hợp   * Hoạt động khác:   + Format file báo cáo  + Báo cáo đồ án phần mô hình thuật toán  + Hỗ trợ tìm bài báo | *9.5/10* |
|  |  |  |  |  |