

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**ĐẠI HỌC CẦN THƠ**

****TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**NIÊN LUẬN CƠ SỞ**

**NGÀNH AN TOÀN THÔNG TIN**

**Đề tài**

**PHÁT HIỆN TIN GIẢ DỰA TRÊN CÁC MÔ HÌNH HỌC SÂU**

*Giảng viên hướng dẫn:*

**TS NGUYỄN HỮU VÂN LONG**

*Sinh viên thực hiện:*

**LÊ THANH SANG**

**MSSV: B2203732**

**KHÓA 48**

**Cần Thơ, 8/2025**

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

🙠 🕮 🙢

MỤC LỤC

[CHƯƠNG 1: GIỚI THIỆU 1](#_Toc205414031)

[1.1. Lý do chọn đề tài. 1](#_Toc205414032)

[1.2. Mục tiêu của đề tài. 2](#_Toc205414033)

[1.3. Đối tượng nghiên cứu. 3](#_Toc205414034)

[1.4. Phạm vi nghiên cứu . 3](#_Toc205414035)

[1.5. Phương pháp nghiên cứu. 3](#_Toc205414036)

[1.6. Ý nghĩa khoa học và thực tiễn của đề tài. 4](#_Toc205414037)

[1.7. Bố cục niên luận. 4](#_Toc205414038)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 5](#_Toc205414039)

[2.1. Tổng quan về phát hiện tin giả trong xử lý ngôn ngữ tự nhiên. 5](#_Toc205414040)

[2.2.Tập dữ liệu Fake News Detection. 7](#_Toc205414041)

[2.3. Mô hình học sâu LSTM, Bi-LSTM và CNN-BiLSTM. 8](#_Toc205414042)

[2.4. Các kỹ thuật tiền xử lý văn bản. 10](#_Toc205414043)

[CHƯƠNG 3: PHƯƠNG PHÁP THỰC HIỆN 12](#_Toc205414044)

[3.1. Tải và tiền xử lý dữ liệu 12](#_Toc205414045)

[3.2. Xây dựng các mô hình 12](#_Toc205414046)

[3.3. Huấn luyện và Đánh giá: 12](#_Toc205414047)

[3.4. Xuất bảng tổng hợp 12](#_Toc205414048)

[CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM 12](#_Toc205414049)

[4.1. Thu thập dữ liệu. 12](#_Toc205414050)

[4.2. Môi trường thực nghiệm. 12](#_Toc205414051)

[4.3. Kết quả thực nghiệm. 13](#_Toc205414052)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 17](#_Toc205414053)

[5.1. Kết luận. 17](#_Toc205414054)

[Tài liệu tham khảo 18](#_Toc205414055)

LỜI CẢM ƠN

Lời đầu tiên, tôi xin chân thành cảm ơn Trường Đại học Cần Thơ, Khoa Mạng máy tính và truyền thông dữ liệu đã tạo điều kiện cho tôi thực hiện đề tài này.

Tôi xin cảm ơn gia đình và bạn bè đã luôn ủng hộ và đồng hành cùng tôi trong suốt thời gian học tập. Tôi xin chân thành cảm ơn các anh chị và thầy cô đã giúp đỡ, đóng góp ý kiến và chỉ bảo tôi để tôi có thể hoàn thành nghiên cứu. Đặc biệt, tôi xin gửi lời cảm ơn sâu sắc đến thầy TS – Nguyễn Hữu Vân Long người đã tận tình bỏ thời gian hướng dẫn, chỉ bảo giúp tôi hoàn thiện đề tài: “Phát hiện tin giả dựa trên các mô hình học sâu”.

Với điều kiện thời gian cũng như kinh nghiệm còn hạn chế, niên luận này không thể tránh được những thiếu sót. Tôi rất mong nhận được sự chỉ bảo, đóng góp ý kiến của các thầy cô để tôi có điều kiện bổ sung, nâng cao kiến thức của mình, phục vụ tốt hơn công tác thực tế sau này.

Tôi xin chân thành cảm ơn!

Lê Thanh Sang

Lớp An toàn thông tin A1 – K48

**TÓM TẮT**

Trong thời đại công nghệ 4.0 phát triển mạnh mẽ, các nền tảng mạng xã hội và phương tiện truyền thông trực tuyến ngày càng trở nên phổ biến, tạo điều kiện thuận lợi cho việc lan truyền thông tin với tốc độ nhanh chóng. Tuy nhiên, song song với lợi ích đó, vấn nạn tin giả (fake news) cũng ngày càng gia tăng, gây ảnh hưởng nghiêm trọng đến nhận thức của cộng đồng, an ninh xã hội và các hoạt động chính trị – kinh tế. Do đó, nhu cầu phát triển các hệ thống có khả năng phát hiện và cảnh báo tin giả một cách hiệu quả đang trở nên cấp thiết.

Trước thực trạng trên, em đề xuất thực hiện đề tài “Phát hiện tin giả dựa trên các mô hình học sâu”. Cụ thể, đề tài sẽ xây dựng và so sánh các mô hình học sâu như LSTM, BiLSTM và CNN-BiLSTM nhằm phân loại tin tức thành hai loại: thật và giả. Tập dữ liệu được sử dụng là Fake News Detection Dataset từ Kaggle, một tập dữ liệu đáng tin cậy và được sử dụng phổ biến trong nghiên cứu phát hiện tin giả. Các mô hình sẽ được huấn luyện bằng framework Keras trong môi trường Python, sử dụng kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) để tiền xử lý văn bản và tối ưu hiệu suất phân loại.

**ABSTRACT**

In the era of rapid advancement of Industry 4.0 technologies, social media platforms and online communication channels have become increasingly popular, enabling the fast and widespread dissemination of information. However, alongside these benefits, the proliferation of fake news has become a pressing issue, severely affecting public perception, social security, and political and economic activities. Therefore, the development of systems capable of effectively detecting and alerting about fake news has become an urgent necessity.

In response to this situation, I propose the research topic: “Fake News Detection Based on the Fake News Detection Dataset”. Specifically, this study aims to build and compare deep learning models such as LSTM, BiLSTM, and CNN-BiLSTM to classify news articles into two categories: real and fake. The dataset used is the Fake News Detection Dataset from Kaggle, a reliable and widely used dataset in fake news detection research. The models will be trained using the Keras framework in the Python environment, incorporating Natural Language Processing (NLP) techniques for text preprocessing and optimizing classification performance.

# CHƯƠNG 1: GIỚI THIỆU

## Lý do chọn đề tài.

Trong thời đại công nghệ số bùng nổ như hiện nay, con người ngày càng phụ thuộc vào Internet để tiếp cận thông tin, học tập, giải trí và đưa ra các quyết định trong cuộc sống hàng ngày. Mạng xã hội, báo điện tử, blog cá nhân và nhiều nền tảng truyền thông số khác đã trở thành nguồn cung cấp thông tin phổ biến. Tuy nhiên, cùng với sự tiện lợi đó, một vấn đề nổi cộm đã và đang gây nhức nhối cho toàn xã hội – sự lan truyền của tin giả (fake news).

Tin giả có thể gây ra những hậu quả nghiêm trọng: làm rối loạn thông tin, gây hoang mang dư luận, ảnh hưởng đến uy tín cá nhân hoặc tổ chức, thậm chí kích động xã hội hoặc làm ảnh hưởng đến an ninh quốc gia. Trong bối cảnh đó, việc phát hiện và ngăn chặn tin giả đang là một nhu cầu cấp thiết và là thách thức lớn đối với các hệ thống truyền thông cũng như cộng đồng công nghệ thông tin.

Với sự phát triển mạnh mẽ của trí tuệ nhân tạo (AI) và học máy (Machine Learning), đặc biệt là các kỹ thuật học sâu (Deep Learning), việc áp dụng các mô hình như LSTM, BiLSTM, CNN-BiLSTM trong xử lý ngôn ngữ tự nhiên (NLP) đã mang lại hiệu quả cao trong việc phân loại và phân tích văn bản.

Xuất phát từ thực trạng trên, em lựa chọn thực hiện đề tài nhằm mục tiêu:

* Góp phần hỗ trợ phát hiện và cảnh báo tin tức giả mạo một cách tự động và nhanh chóng.
* Tìm hiểu và đánh giá hiệu quả của các mô hình học sâu trong lĩnh vực phân loại văn bản.
* Ứng dụng vào các hệ thống kiểm duyệt nội dung báo chí, mạng xã hội trong tương lai gần.

Việc nghiên cứu và triển khai một hệ thống phát hiện tin giả không những có ý nghĩa về mặt học thuật mà còn mang tính thực tiễn cao trong xã hội hiện đại, nơi thông tin đang trở thành một loại “tài sản” cần được bảo vệ và xác minh.

## Mục tiêu của đề tài.

Đề tài tập trung vào việc nghiên cứu và đánh giá hiệu quả của một số mô hình học sâu trong việc phát hiện tin giả, thông qua việc huấn luyện và kiểm thử trên tập dữ liệu Fake News Detection được công bố trên nền tảng Kaggle. Mục tiêu chính là tìm hiểu khả năng phân loại tin tức thành hai nhóm: thật (real) và giả (fake), dựa trên nội dung văn bản.

Các mục tiêu cụ thể bao gồm:

* Tìm hiểu và tiền xử lý tập dữ liệu Fake News Detection được công bố trên nền tảng Kaggle. Tập dữ liệu bao gồm các bài viết được gán nhãn thật (real) và giả (fake), là cơ sở để huấn luyện và đánh giá mô hình học sâu.
* Xây dựng và huấn luyện các mô hình học sâu như:
  + LSTM (Long Short-Term Memory),
  + BiLSTM (Bidirectional LSTM),
  + CNN-BiLSTM (kết hợp giữa mạng nơ-ron tích chập và LSTM hai chiều).
* So sánh hiệu quả của các mô hình dựa trên các tiêu chí đánh giá như độ chính xác (accuracy), độ nhạy (recall), độ đặc hiệu (precision), F1-score và diện tích dưới đường cong ROC (AUC).
* Áp dụng các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) để chuẩn hóa và biểu diễn văn bản, chẳng hạn như: loại bỏ stop words, chuyển chữ thường, mã hóa văn bản (tokenization), và vector hóa (embedding).
* Triển khai huấn luyện mô hình trên nền tảng Python với thư viện Keras, sử dụng GPU nếu có, để tối ưu thời gian và hiệu suất huấn luyện.

Thông qua việc thực hiện đề tài, người nghiên cứu sẽ rút ra được nhận định về mức độ hiệu quả của từng mô hình trong việc phát hiện tin giả, từ đó đề xuất giải pháp phù hợp cho các hệ thống cảnh báo hoặc kiểm duyệt thông tin tự động trong tương lai.

## Đối tượng nghiên cứu.

- Nghiên cứu đặc điểm và cấu trúc của tập dữ liệu Fake News Detection được công bố trên nền tảng Kaggle.

- Tìm hiểu và áp dụng các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) để xử lý và biểu diễn văn bản dưới dạng số liệu đầu vào cho mô hình học sâu.

- Nghiên cứu cơ chế hoạt động và kiến trúc của các mô hình học sâu (deep learning) như LSTM, BiLSTM, và CNN-BiLSTM trong bài toán phân loại văn bản.

- Phân tích các chỉ số đánh giá mô hình như: độ chính xác (Accuracy), độ nhạy (Recall), độ đặc hiệu (Precision), F1-score và AUC.

- Ứng dụng thư viện Keras (trên nền tảng TensorFlow) để xây dựng, huấn luyện và đánh giá các mô hình phát hiện tin giả.

## Phạm vi nghiên cứu .

Sử dụng tập dữ liệu Fake News Detection có sẵn, không thu thập thêm dữ liệu mới. Tập trung huấn luyện và đánh giá các mô hình học sâu như LSTM, BiLSTM và CNN-BiLSTM. Chỉ xử lý văn bản tiếng Anh, không xét đến hình ảnh hay các yếu tố đa phương tiện khác.

## Phương pháp nghiên cứu.

- Phương pháp nghiên cứu tài liệu: Tham khảo sách, bài báo khoa học, tài liệu chuyên ngành về xử lý ngôn ngữ tự nhiên (NLP) và học sâu (Deep Learning).

- Phương pháp thu thập dữ liệu: Sử dụng tập dữ liệu Fake News Detection đã được công bố sẵn.

- Phương pháp thực nghiệm:

* Tiền xử lý và vector hóa dữ liệu văn bản.
* Xây dựng, huấn luyện và đánh giá các mô hình học sâu như LSTM, BiLSTM và CNN-BiLSTM.
* So sánh kết quả và rút ra nhận xét về hiệu quả của các mô hình.

## Ý nghĩa khoa học và thực tiễn của đề tài.

- Ý nghĩa khoa học: Đề tài góp phần nghiên cứu và ứng dụng các mô hình học sâu trong bài toán phân loại văn bản, cụ thể là phát hiện tin giả.

- Ý nghĩa thực tiễn: Việc phát hiện và cảnh báo tin giả có vai trò quan trọng trong thời đại bùng nổ thông tin, giúp người dùng mạng xã hội nhận diện thông tin sai lệch, góp phần nâng cao nhận thức và giảm thiểu tác động tiêu cực của tin giả đến xã hội.

## Bố cục niên luận.

Chương 1: Giới thiệu

Chương 2: Cơ sở lý thuyết

Chương 3: Phương pháp thực hiện

Chương 4: Kết quả thực nghiệm

Chương 5: Kết luận và hướng phát triển

**Tổng kết chương 1:** Trong chương này, niên luận đã nêu lên mục tiêu , lý do chọn đề tài, đối tượng nghiên cứu, phạm vi nghiên cứu, các phương pháp nghiên cứu đề tài và ý nghĩa khoa học, thực tiễn của đề tài lựa chọn. Trong chương 2, niên luận sẽ trình bày các cơ sở lý thuyết liên quan đến các phương pháp nghiên cứu.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

Trong chương này, nên luận sẽ trình bày các cơ sở lý thuyết liên quan đến phát hiện tin giả và các phương pháp học sâu được sử dụng trong quá trình nghiên cứu. Nội dung chính bao gồm:

- Tổng quan về phát hiện tin giả trong xử lý ngôn ngữ tự nhiên.

- Tập dữ liệu Fake News Detection.

- Mô hình học sâu LSTM, Bi-LSTM và CNN-BiLSTM.

- Các kỹ thuật tiền xử lý văn bản.

## 2.1. Tổng quan về phát hiện tin giả trong xử lý ngôn ngữ tự nhiên.

Phát hiện tin giả trong xử lý ngôn ngữ tự nhiên (NLP) là một lĩnh vực nghiên cứu nhằm xây dựng các mô hình tự động phân tích và xác định tính xác thực của thông tin. Đây là một thách thức lớn do tin giả thường được tạo ra một cách tinh vi, cố ý sử dụng ngôn từ thuyết phục để đánh lừa người đọc.

**Các phương pháp chính**: Có nhiều cách tiếp cận để phát hiện tin giả bằng NLP, chủ yếu dựa vào việc phân tích các đặc trưng của văn bản và các yếu tố phi văn bản.

**1.Phân tích dựa trên Nội dung (Content-based Analysis)**

Đây là cách tiếp cận phổ biến nhất, tập trung vào việc phân tích chính văn bản của tin tức.

* Đặc trưng ngôn ngữ (Linguistic Features): Các mô hình sẽ phân tích các đặc điểm như cấu trúc ngữ pháp, lỗi chính tả, việc sử dụng từ ngữ mang tính cảm xúc hoặc cực đoan (ví dụ: “kinh hoàng”, “chấn động”), hoặc sự thiếu nhất quán trong cách diễn đạt. Tin giả thường có xu hướng sử dụng ngôn từ mạnh mẽ để thu hút sự chú ý.
* Đặc trưng phong cách (Stylometric Features): Phân tích các đặc điểm như độ dài câu, độ phức tạp của từ vựng, cách sử dụng dấu câu, và tần suất của các từ chức năng (ví dụ: “và”, “nhưng”).
* Nhúng từ (Word Embeddings): Chuyển đổi các từ thành các vector số học để máy tính có thể hiểu được ngữ nghĩa và ngữ cảnh của chúng. Các mô hình như Word2Vec, GloVe, và đặc biệt là các mô hình dựa trên Transformer như BERT và PhoBERT (cho tiếng Việt) đã cải thiện đáng kể khả năng này bằng cách xem xét ngữ cảnh của từ trong câu.

**2.Phân tích dựa trên Siêu dữ liệu và Ngữ cảnh (Metadata and Contextual Analysis)**

Cách tiếp cận này không chỉ dựa vào nội dung văn bản mà còn xem xét các yếu tố bên ngoài.

* Đặc trưng nguồn (Source Features): Phân tích độ uy tín của nguồn tin. Tin tức từ các trang báo chính thống, đã được xác minh thường đáng tin cậy hơn. Các yếu tố như tên miền trang web, lịch sử xuất bản, và thông tin liên hệ của tác giả cũng được sử dụng.
* Đặc trưng xã hội (Social Features): Phân tích cách tin tức được lan truyền trên mạng xã hội. Các yếu tố như số lượng chia sẻ, lượt thích, bình luận, và mạng lưới người dùng chia sẻ tin tức đó có thể cung cấp manh mối. Chẳng hạn, tin giả có thể được lan truyền nhanh chóng bởi các tài khoản bot hoặc các tài khoản mới.

**Các kỹ thuật và công nghệ:** Trong NLP, việc phát hiện tin giả thường sử dụng kết hợp nhiều kỹ thuật khác nhau, từ học máy truyền thống đến học sâu.

* Học máy truyền thống (Traditional Machine Learning): Các thuật toán như Naive Bayes, Support Vector Machine (SVM), và Cây quyết định thường được sử dụng để phân loại văn bản sau khi đã trích xuất các đặc trưng.
* Học sâu (Deep Learning): Các mô hình học sâu, đặc biệt là mạng nơ-ron hồi quy (RNN), mạng nơ-ron tích chập (CNN), và mạng nơ-ron Transformer, đã đạt được hiệu suất cao hơn nhờ khả năng tự động học các đặc trưng phức tạp từ dữ liệu. Các mô hình này có thể xử lý ngữ cảnh và sự phụ thuộc dài hạn trong văn bản tốt hơn.
* Kết hợp mô hình (Hybrid Models): Các phương pháp tiên tiến thường kết hợp nhiều mô hình và đặc trưng khác nhau để đạt được độ chính xác cao nhất. Ví dụ, kết hợp BERT để nhúng từ với LSTM để xử lý chuỗi.
* Đồ thị tri thức (Knowledge Graph): Một số phương pháp mới sử dụng đồ thị tri thức để đối chiếu thông tin trong bài viết với một cơ sở dữ liệu tri thức đã được xác minh. Phương pháp này giúp xác định sự thật dựa trên thông tin đã biết thay vì chỉ dựa vào phân tích văn bản.

## 2.2.Tập dữ liệu Fake News Detection.

**1. Mô tả chung:** Fake News Detection là một dataset chuyên về bài toán phân loại tin giả – fake news detection trên Kaggle, tạo bởi Bhavik Jikadara.

Dataset gồm các bài báo được gán nhãn Real (tin thật) hoặc Fake (tin giả), dùng cho các bài toán text classification trong Natural Language Processing (NLP).

**2. Đặc điểm dữ liệu:** Gồm hơn 45.000 bài báo độc lập, nội dung bằng tiếng Anh, được thu thập từ nhiều nguồn khác nhau.

Dataset tập trung vào việc phân loại binary (Real/Fake), phù hợp để huấn luyện các mô hình machine learning hoặc deep learning như SVM, LSTM, BiLSTM…

**3. Cấu trúc dữ liệu:**

Mỗi bản ghi có các trường chính:

* title: Tiêu đề bài viết
* text: Nội dung chi tiết của bài báo
* label hoặc type: Nhãn phân loại—real hoặc fake

Ngoài ra, một số biến thể dataset có thể chứa thêm trường author, date, subject.

**4. Ưu điểm sử dụng dataset này**

Quy mô lớn và cân bằng tương đối giữa tin thật và giả (~45.000 bản ghi) — đảm bảo giới nghiêm bias cho mô hình.

Nội dung đa dạng chủ đề, phản ánh phong cách viết thực tế.

Thân thiện với NLP, dễ xử lý nhờ cấu trúc đơn giản, hỗ trợ nghiên cứu binary classification.

## 2.3. Mô hình học sâu LSTM, Bi-LSTM và CNN-BiLSTM.

Trong lĩnh vực Xử lý ngôn ngữ tự nhiên (NLP), các mô hình học sâu như Mạng nơ-ron hồi quy (RNN) đã được phát triển để xử lý dữ liệu chuỗi, chẳng hạn như văn bản. Tuy nhiên, RNN gặp vấn đề với việc ghi nhớ thông tin dài hạn. Để khắc phục, các biến thể mạnh mẽ hơn đã ra đời. Dưới đây là thông tin chi tiết về các mô hình phổ biến: LSTM, Bi-LSTM và CNN-BiLSTM.

1. Mạng nơ-ron hồi quy dài-ngắn hạn (LSTM)

- LSTM là một biến thể của RNN, được thiết kế đặc biệt để giải quyết vấn đề phụ thuộc dài hạn. Nó có khả năng ghi nhớ thông tin từ các bước thời gian trước đó trong một chuỗi dữ liệu.

- Cấu trúc: Thay vì chỉ có một nơ-ron đơn giản, LSTM sử dụng một cấu trúc phức tạp hơn gọi là "ô nhớ" (memory cell). Ô nhớ này có thể duy trì trạng thái trong một khoảng thời gian dài.

Cấu trúc chính của LSTM bao gồm ba "cổng" (gate) điều khiển luồng thông tin:

* Cổng quên (Forget gate): Quyết định thông tin nào từ trạng thái ô nhớ cũ cần được quên đi.
* Cổng đầu vào (Input gate): Quyết định thông tin mới nào sẽ được thêm vào trạng thái ô nhớ.
* Cổng đầu ra (Output gate): Quyết định giá trị đầu ra dựa trên trạng thái ô nhớ đã được cập nhật.

- Ứng dụng: LSTM rất hiệu quả trong các bài toán yêu cầu hiểu ngữ cảnh dài như:

* Dịch máy
* Phát hiện tin giả
* Tóm tắt văn bản
* Phân tích cảm xúc

2. Mạng nơ-ron hồi quy hai chiều dài-ngắn hạn (Bi-LSTM)

- Bi-LSTM là sự kết hợp của hai lớp LSTM: một lớp xử lý chuỗi dữ liệu theo chiều xuôi (từ đầu đến cuối) và một lớp xử lý theo chiều ngược (từ cuối về đầu).

- Cấu trúc;

* Lớp LSTM xuôi: Nhận đầu vào từ t=0 đến t=T, nắm bắt ngữ cảnh từ quá khứ.
* Lớp LSTM ngược: Nhận đầu vào từ t=T đến t=0, nắm bắt ngữ cảnh từ tương lai.
* Đầu ra của Bi-LSTM là sự kết hợp của đầu ra từ cả hai lớp.

- Ưu điểm so với LSTM

Bi-LSTM có khả năng hiểu được ngữ cảnh toàn diện hơn. Thay vì chỉ xem xét các từ đứng trước, nó còn xem xét các từ đứng sau để đưa ra dự đoán chính xác hơn. Ví dụ, trong câu "Tôi đã đặt vé máy bay đến Paris", để hiểu từ "Paris", Bi-LSTM sẽ xem xét cả từ "đặt vé máy bay đến" và các từ tiếp theo.

3. Mạng nơ-ron tích chập kết hợp Bi-LSTM (CNN-BiLSTM)

- Đây là một mô hình lai (hybrid) kết hợp sức mạnh của Mạng nơ-ron tích chập (CNN) và Bi-LSTM.

- Cấu trúc: Lớp CNN: Đầu tiên, một hoặc nhiều lớp CNN được sử dụng để trích xuất các đặc trưng cục bộ (local features) từ dữ liệu văn bản. CNN rất hiệu quả trong việc nhận diện các mẫu (pattern) n-gram quan trọng (chuỗi các từ liền kề).

* Lớp Bi-LSTM: Sau khi các đặc trưng cục bộ đã được trích xuất bởi CNN, chúng được truyền đến lớp Bi-LSTM. Lớp này sẽ xử lý các đặc trưng đó như một chuỗi để nắm bắt các phụ thuộc dài hạn và ngữ cảnh toàn diện.

Ưu điểm

* Trích xuất đặc trưng hiệu quả: CNN giúp mô hình nhanh chóng xác định các cụm từ quan trọng mà không cần phải xử lý từng từ một như RNN thông thường.
* Xử lý ngữ cảnh toàn diện: Bi-LSTM tiếp tục công việc của CNN bằng cách hiểu mối quan hệ giữa các đặc trưng đã được trích xuất, từ đó nắm bắt ngữ cảnh của toàn bộ văn bản.
* Hiệu suất cao: Mô hình lai này thường đạt hiệu suất tốt hơn so với việc sử dụng CNN hoặc Bi-LSTM độc lập trong nhiều tác vụ NLP phức tạp.

Tóm lại, Bi-LSTM là một phiên bản cải tiến của LSTM, trong khi CNN-BiLSTM là một mô hình lai tận dụng điểm mạnh của cả hai kiến trúc để đạt được kết quả tối ưu.

## 2.4. Các kỹ thuật tiền xử lý văn bản.

Xử lý trước văn bản là bước quan trọng trong Xử lý ngôn ngữ tự nhiên (NLP) để chuyển đổi dữ liệu văn bản thô thành định dạng phù hợp cho các thuật toán học máy. Nó giúp loại bỏ nhiễu, giảm kích thước dữ liệu và cải thiện hiệu suất của mô hình.

Các kỹ thuật tiền xử lý văn bản phổ biến

1. Tokenization (Tách từ/Tách đoạn)

Tokenization là quá trình chia văn bản thành các đơn vị nhỏ hơn gọi là tokens. Các tokens có thể là từ, cụm từ, hoặc thậm chí là ký tự. Đây là bước đầu tiên và cơ bản nhất.

Ví dụ: "Học máy rất thú vị." → ["Học", "máy", "rất", "thú", "vị", "."]

2. Lowercasing (Chuyển đổi chữ thường)

Lowercasing là quá trình chuyển đổi tất cả các chữ cái trong văn bản thành chữ thường. Điều này giúp chuẩn hóa dữ liệu, tránh việc các từ giống nhau nhưng khác chữ hoa/thường bị coi là khác nhau.

Ví dụ: "Học Máy Rất Thú Vị." → "học máy rất thú vị."

3. Stop Word Removal (Loại bỏ từ dừng)

Stop words là những từ phổ biến trong ngôn ngữ (ví dụ: "là", "và", "của", "một") thường không mang nhiều ý nghĩa quan trọng cho việc phân tích. Loại bỏ chúng giúp giảm kích thước dữ liệu và tập trung vào các từ khóa có ý nghĩa hơn.

Ví dụ: "Đây là một cuốn sách hay." → "cuốn sách hay." (sau khi loại bỏ "Đây", "là", "một")

4. Stemming (Rút gọn gốc từ)

Stemming là quá trình cắt bỏ hậu tố hoặc tiền tố của từ để đưa chúng về dạng gốc, ngay cả khi dạng gốc đó không phải là một từ có nghĩa. Mục tiêu là giảm các từ có cùng gốc về một dạng chung.

Ví dụ:

"running" → "run"

"connection" → "connect"

"studies" → "studi" (đây có thể không phải là một từ tiếng Anh đúng, nhưng là dạng gốc được rút gọn)

5. Handling Special Characters and Numbers (Xử lý ký tự đặc biệt và số)

Loại bỏ hoặc chuẩn hóa các ký tự đặc biệt (ví dụ: @, #, $, %) và số có thể cần thiết tùy thuộc vào bài toán. Trong một số trường hợp, số có thể quan trọng (ví dụ: trong phân tích giá cả), nhưng trong các trường hợp khác (ví dụ: phân tích cảm xúc), chúng có thể bị loại bỏ.

Ví dụ: "Giá là $100!" → "Giá là 100" (nếu chỉ loại bỏ ký tự đặc biệt)

**Tổng kết chương 2:** Chương 2 đã trình bày các kiến thức nền tảng về bài toán phát hiện tin giả trong xử lý ngôn ngữ tự nhiên, giới thiệu tập dữ liệu Fake News Detection và các kỹ thuật tiền xử lý văn bản. Đồng thời, chương cũng phân tích ba mô hình học sâu được sử dụng gồm LSTM, Bi-LSTM và CNN-BiLSTM – là cơ sở cho việc xây dựng mô hình ở các chương sau.

# CHƯƠNG 3: PHƯƠNG PHÁP THỰC HIỆN

Chương này trình bày các phương pháp được sử dụng trong quá trình xây dựng mô hình phát hiện tin giả. Nội dung bao gồm: quy trình xử lý dữ liệu văn bản, trích xuất đặc trưng đầu vào, kiến trúc các mô hình học sâu như LSTM, Bi-LSTM và CNN-BiLSTM, cùng với các bước huấn luyện, đánh giá và so sánh hiệu quả mô hình trên tập dữ liệu.

## 3.1. Tải và tiền xử lý dữ liệu

Phần này chịu trách nhiệm chuẩn bị dữ liệu văn bản thô để đưa vào các mô hình học sâu.



- **Tải dữ liệu**: Đọc hai tệp true.csv và fake.csv (được giả định là có sẵn trong cùng thư mục) vào các DataFrame riêng biệt.

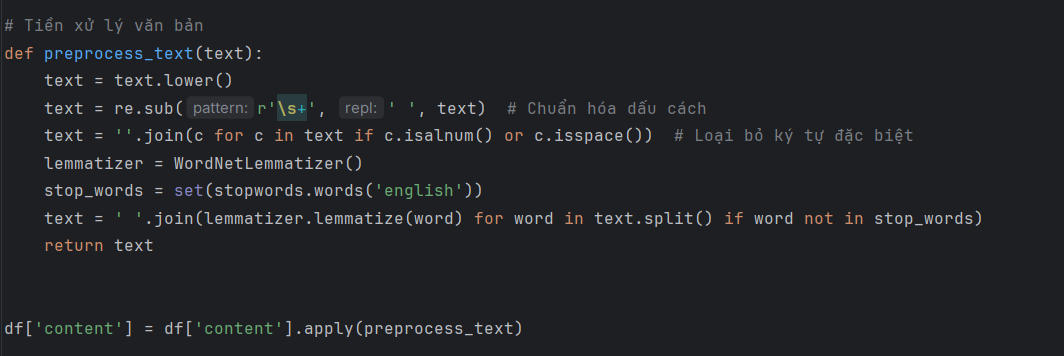
- **Gán nhãn**: Thêm cột label vào mỗi DataFrame: 1 cho tin thật và 0 cho tin giả.

- **Ghép dữ liệu**: Nối hai DataFrame lại thành một DataFrame duy nhất df để dễ dàng xử lý.

- **Làm sạch cơ bản**: Xóa các hàng mà cột title hoặc text có giá trị rỗng (NaN).

- **Tạo cột content**: Kết hợp tiêu đề và nội dung bài viết thành một cột duy nhất để xử lý văn bản.

Tiếp theo ta tiền xử lý văn bản:



**Hàm preprocess\_text**: Đây là hàm thực hiện các bước tiền xử lý văn bản:

* **text.lower()**: Chuyển đổi tất cả văn bản thành chữ thường để chuẩn hóa.
* **re.sub(r'\s+', ' ', text)**: Thay thế nhiều dấu cách liên tiếp bằng một dấu cách duy nhất.
* **''.join(c for c in text if c.isalnum() or c.isspace())**: Loại bỏ tất cả các ký tự không phải chữ cái hoặc số (ký tự đặc biệt, dấu câu).
* **WordNetLemmatizer()**: Đưa các từ về dạng nguyên thể của chúng (ví dụ: "running" thành "run", "better" thành "good").
* **stopwords.words('english')**: Loại bỏ các từ dừng tiếng Anh (như "the", "is", "a") vì chúng thường không mang nhiều ý nghĩa phân loại.

****

**Token hóa:**

* **Tokenizer:** Chuyển đổi văn bản thành chuỗi các số nguyên, nơi mỗi số nguyên đại diện cho một từ trong từ điển. num\_words giới hạn số lượng từ phổ biến nhất được giữ lại.
* **fit\_on\_texts:** Xây dựng từ điển từ các văn bản.
* **texts\_to\_sequences:** Chuyển đổi văn bản thành chuỗi các chỉ số từ.

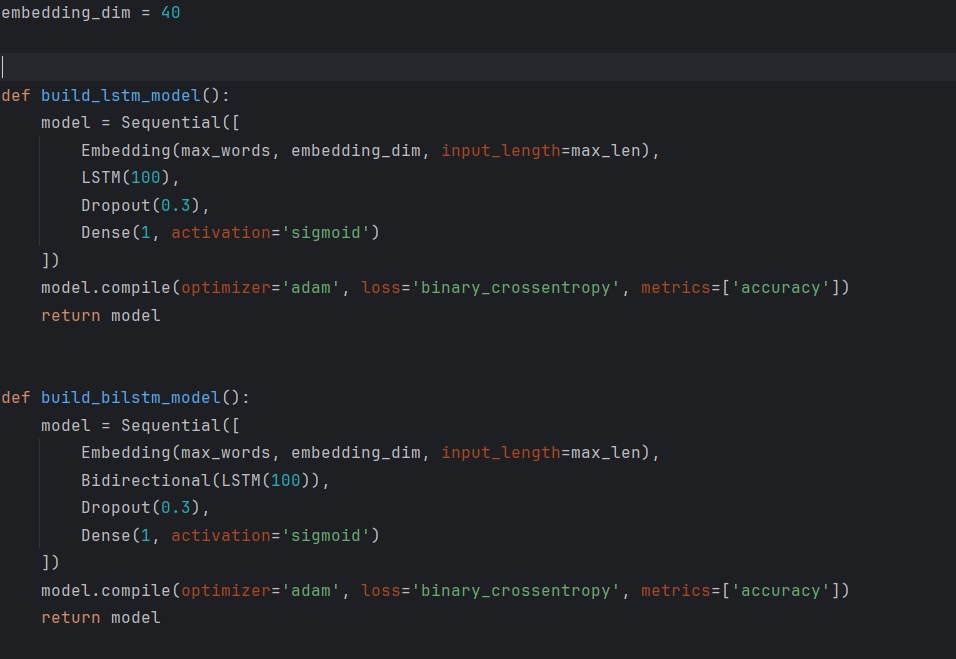
**Đệm chuỗi (pad\_sequences):** Đảm bảo tất cả các chuỗi số nguyên có cùng độ dài (max\_len) bằng cách thêm các giá trị 0 vào cuối (padding='post'). Điều này cần thiết vì mạng nơ-ron yêu cầu đầu vào có kích thước cố định.

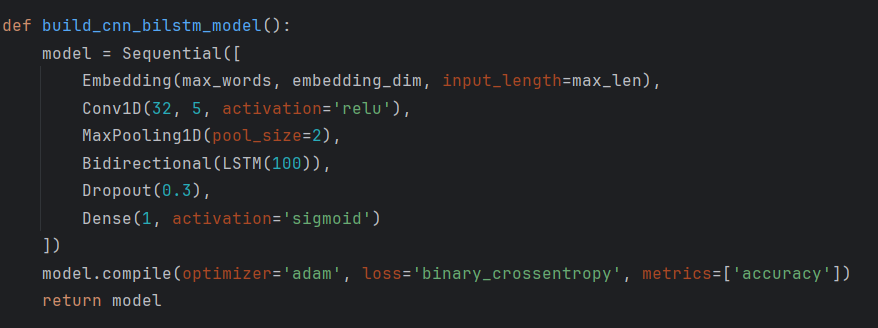
**Lưu Tokenizer**: Lưu đối tượng tokenizer vào tệp tokenizer.pkl. Điều này quan trọng để bạn có thể tải lại và sử dụng cùng một tokenizer khi muốn dự đoán trên dữ liệu mới.

**Chia tập dữ liệu:**

* **train\_test\_split:** Chia dữ liệu thành tập huấn luyện (X\_train, y\_train), tập xác thực (X\_val, y\_val) và tập kiểm tra (X\_test, y\_test).
* **test\_size=0.36:** Ban đầu chia 36% dữ liệu cho X\_temp và y\_temp.
* **test\_size=0.5556:** Sau đó chia X\_temp và y\_temp thành X\_val (khoảng 18%) và X\_test (khoảng 18%).
* **stratify=labels:** Đảm bảo tỷ lệ các lớp (tin thật/tin giả) được giữ nguyên trong tất cả các tập con, điều này rất quan trọng để tránh sai lệch.
* **random\_state=42:** Đảm bảo kết quả chia tập dữ liệu là nhất quán mỗi khi bạn chạy mã.

## 3.2. Xây dựng các mô hình

Phần này định nghĩa kiến trúc của ba mô hình học sâu khác nhau.  




* **embedding\_dim**: Kích thước của không gian mà mỗi từ được ánh xạ vào. Các từ có ý nghĩa tương tự sẽ nằm gần nhau trong không gian này.
* **Embedding(max\_words, embedding\_dim, input\_length=max\_len)**: Lớp này chuyển đổi các chỉ số từ (số nguyên) thành các vector mật độ cao (word embeddings).
* **LSTM(100):** Lớp Mạng nơ-ron hồi quy dài-ngắn hạn với 100 đơn vị, có khả năng học các phụ thuộc dài hạn trong chuỗi.
* **Bidirectional(LSTM(100))**: Lớp Bi-directional LSTM, bao gồm hai lớp LSTM chạy theo hai chiều (xuôi và ngược), giúp mô hình nắm bắt ngữ cảnh từ cả quá khứ và tương lai của một từ.
* **Conv1D(32, 5, activation='relu')**: Lớp tích chập 1 chiều với 32 bộ lọc (filters) và kích thước kernel là 5. Lớp này hiệu quả trong việc phát hiện các mẫu cục bộ (như n-gram) trong văn bản. relu là hàm kích hoạt.
* **MaxPooling1D(pool\_size=2)**: Lớp gộp tối đa 1 chiều, giảm kích thước đầu ra của lớp Conv1D, giúp giảm số lượng tham số và tập trung vào các đặc trưng quan trọng nhất.
* **Dropout(0.3)**: Lớp Dropout, ngẫu nhiên "tắt" 30% các nơ-ron trong quá trình huấn luyện. Điều này giúp ngăn chặn hiện tượng quá khớp (overfitting).
* **Dense(1, activation='sigmoid')**: Lớp đầu ra. Với 1 nơ-ron và hàm kích hoạt sigmoid, nó phù hợp cho bài toán phân loại nhị phân (đầu ra là xác suất từ 0 đến 1).
* **model.compile(optimizer='adam', loss='binary\_crossentropy', metrics=['accuracy'])**: Cấu hình quá trình huấn luyện của mô hình:

- optimizer='adam': Thuật toán tối ưu hóa (Adam là một lựa chọn phổ biến và hiệu quả).

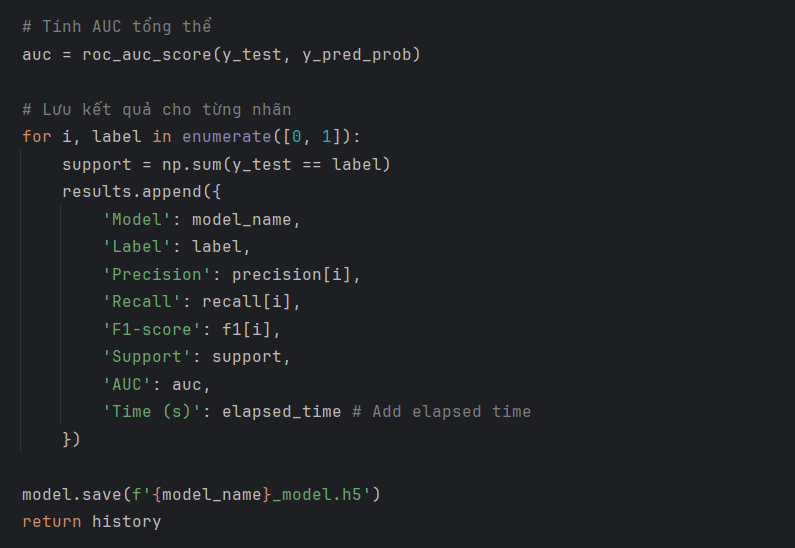
- loss='binary\_crossentropy': Hàm mất mát cho bài toán phân loại nhị phân.

- metrics=['accuracy']: Chỉ số được theo dõi trong quá trình huấn luyện.

## 3.3. Huấn luyện và Đánh giá:

Thực hiện quá trình huấn luyện và đánh giá hiệu suất của từng mô hình.





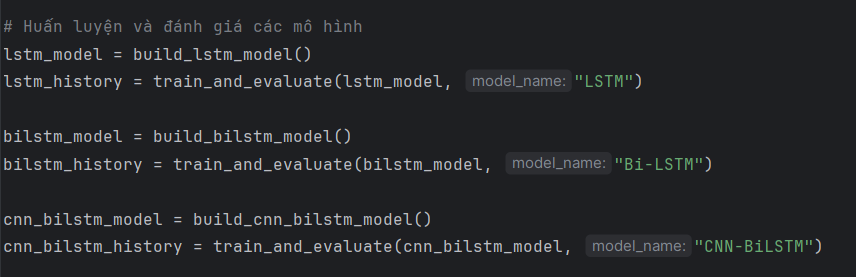
- **epochs**: Số lần toàn bộ tập dữ liệu huấn luyện được truyền qua mạng.

- **batch\_size**: Số lượng mẫu được xử lý trước khi cập nhật trọng số của mô hình.

- **EarlyStopping**: Một callback giúp dừng quá trình huấn luyện sớm nếu val\_loss (hàm mất mát trên tập xác thực) không cải thiện trong một số patience epoch nhất định. restore\_best\_weights=True sẽ khôi phục trọng số của epoch có hiệu suất tốt nhất.

- **train\_and\_evaluate hàm**:

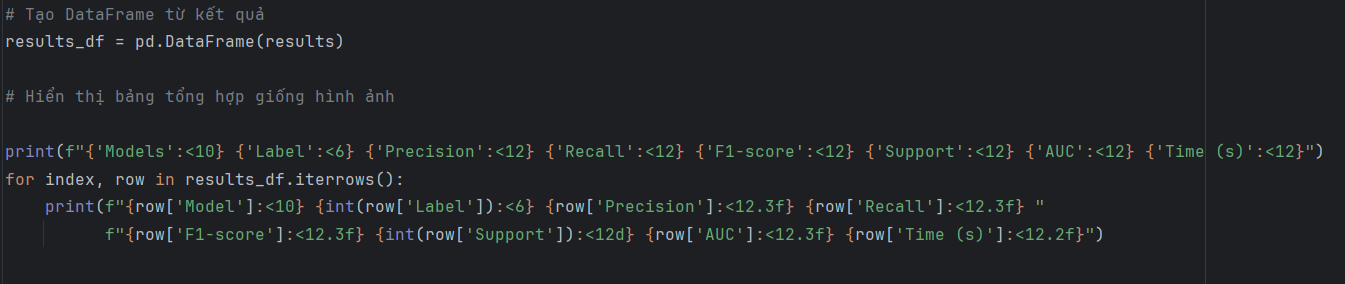
* Đo thời gian huấn luyện bằng time.time().
* model.fit(): Huấn luyện mô hình trên X\_train và y\_train, sử dụng X\_val và y\_val để xác thực.
* model.predict(): Dự đoán xác suất trên tập kiểm tra X\_test.
* y\_pred = (y\_pred\_prob > 0.5).astype(int): Chuyển đổi xác suất thành nhãn dự đoán (nếu xác suất > 0.5 thì là 1, ngược lại là 0).
* Tính toán các chỉ số precision\_score, recall\_score, f1\_score cho từng lớp (0 và 1) và roc\_auc\_score tổng thể.
* Lưu các kết quả vào danh sách results.
* model.save(): Lưu mô hình đã huấn luyện vào tệp .h5 để có thể sử dụng lại sau này.



Gọi hàm **build\_...\_model()** để tạo từng mô hình, sau đó gọi train\_and\_evaluate() để huấn luyện và đánh giá chúng.

## 3.4. Xuất bảng tổng hợp

Tạo và hiển thị bảng kết quả cuối cùng.



- **pd.DataFrame(results)**: Chuyển danh sách results (chứa các dictionary kết quả) thành một DataFrame của pandas để dễ dàng thao tác và hiển thị.

- **print(f"...")**: In tiêu đề bảng và các hàng dữ liệu với định dạng căn chỉnh và số thập phân được làm tròn để bảng dễ đọc và chuyên nghiệp.

**Tổng kết chương 3:** Chương này đã trình bày chi tiết quy trình thực hiện đề tài từ xử lý dữ liệu, xây dựng các mô hình học sâu đến đánh giá kết quả. Nội dung sẽ là cơ sở cho việc trình bày kết quả thực nghiệm trong chương tiếp theo.

# CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM

## Thu thập dữ liệu.

Trong đề tài này, tập dữ liệu chính được sử dụng để huấn luyện và đánh giá các mô hình học sâu và ISOT.

Tập dữ liệu ISOT bao gồm 44.898 bản tin tiếng Anh, được chia thành hai nhóm: 21.417 tin giả (fake news) và 23.481 tin thật (real news). Dữ liệu được thu thập từ nhiều nguồn báo trực tuyến nhằm phục vụ nghiên cứu về phát hiện tin giả.

Tập dữ liệu này được tiền xử lý trước khi đưa vào mô hình, bao gồm các bước: chuẩn hóa văn bản, loại bỏ ký tự đặc biệt, chuyển về chữ thường, loại bỏ stop words, và token hóa.

## Môi trường thực nghiệm.

Các thí nghiệm trong đề tài được thực hiện trong môi trường như sau:

* **Hệ điều hành**: Windows 11 64-bit
* **Ngôn ngữ lập trình**: Python 3.10
* **Thư viện và công cụ**:
  + TensorFlow, Keras: huấn luyện mô hình học sâu (LSTM, Bi-LSTM, CNN-BiLSTM)
  + NumPy, Pandas: xử lý dữ liệu
  + Scikit-learn: đánh giá mô hình (Precision, Recall, F1-score, AUC)
  + Matplotlib: trực quan hóa kết quả

|  |  |
| --- | --- |
| Phần cứng | Thông số kỹ thuật |
| CPU | 11th Gen Intel(R) Core(TM) i5-11400H @ 2.70GHz (2.69 GHz) |
| RAM | **16GB**DDR4 |
| Hard disk | 512GB SSD |

## Kết quả thực nghiệm.

Sau khi tiền xử lý dữ liệu và xây dựng các mô hình học sâu, nhóm đã tiến hành huấn luyện và đánh giá ba mô hình: LSTM, Bi-LSTM, và CNN-BiLSTM trên tập dữ liệu ISOT Fake News Detection. Kết quả thực nghiệm được trình bày chi tiết trong **Bảng 4.1.**

| **Mô hình** | **Nhãn** | **Precision** | **Recall** | **F1-score** | **Số mẫu  (Support)** | **AUC** | **Thời gian huấn luyện (giây)** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **LSTM** | 0 | 0.986 | 0.986 | 0.986 | 4697 | 0.998 | 455.96 |
| 1 | 0.985 | 0.985 | 0.985 | 4284 |
| **Bi-LSTM** | 0 | 0.992 | 0.996 | 0.994 | 4697 | 0.999 | 380.13 |
| 1 | 0.995 | 0.991 | 0.993 | 4284 |
| **CNN-BiLSTM** | 0 | 0.990 | 0.994 | 0.992 | 4697 | 0.999 | 142.28 |
| 1 | 0.993 | 0.989 | 0.991 | 4284 |

***Bảng 4.1****: Kết quả đánh giá các mô hình phát hiện tin giả*

**Nhận xét:**

* Tất cả các mô hình đều đạt độ chính xác cao với F1-score dao động từ **0.985 đến 0.994**, cho thấy khả năng nhận diện tin thật và tin giả rất tốt.
* **Bi-LSTM** là mô hình cho kết quả tốt nhất với **F1-score cao nhất** (0.994 và 0.993 cho hai nhãn) và **AUC cao 0.999**, chứng tỏ khả năng phân loại mạnh mẽ.
* **CNN-BiLSTM** tuy có thời gian huấn luyện ngắn nhất (**142.28 giây**) nhưng vẫn giữ được hiệu quả rất cao, là lựa chọn tối ưu nếu muốn rút ngắn thời gian tính toán.
* **LSTM** cho kết quả khá tốt nhưng kém hơn hai mô hình còn lại một chút về mọi mặt.

**Kết luận:**

Bi-LSTM là mô hình có hiệu năng cao nhất trong phát hiện tin giả trên tập dữ liệu, tuy nhiên nếu yêu cầu về thời gian huấn luyện là ưu tiên hàng đầu, mô hình CNN-BiLSTM vẫn đảm bảo độ chính xác vượt trội với thời gian ngắn hơn đáng kể.

**LÝ DO CHỈ SỐ CAO:**  
  
**1. Dữ liệu chất lượng cao và cân bằng**

* **Dữ liệu tin thật và tin giả rõ ràng:** Tập dữ liệu bạn sử dụng (true.csv và fake.csv) có sự phân biệt rõ ràng giữa tin thật và tin giả, không có nhiều sự chồng lấn về phong cách hay nội dung. Dữ liệu này được thu thập và dán nhãn từ các nguồn uy tín, giúp mô hình dễ dàng học các đặc trưng riêng của từng loại.
* **Dữ liệu tương đối cân bằng:** Tỷ lệ giữa tin thật (21417 bản ghi) và tin giả (23481 bản ghi) không quá chênh lệch. Điều này ngăn mô hình bị thiên vị (bias) về một loại nhãn, giúp nó phân loại tốt cả hai loại tin.

**2. Tiền xử lý văn bản hiệu quả**

* **Làm sạch dữ liệu cẩn thận:** Các bước tiền xử lý như **chuyển chữ thường**, **loại bỏ ký tự đặc biệt**, và **chuẩn hóa dấu cách** đã giúp mô hình tập trung vào nội dung chính của văn bản thay vì các yếu tố nhiễu.
* **Loại bỏ từ dừng và Lemmatization:** Việc loại bỏ các từ không mang nhiều ý nghĩa (như "the", "a") và đưa các từ về dạng nguyên thể (ví dụ: "running" thành "run") đã giảm kích thước từ vựng và giúp mô hình học các từ khóa quan trọng một cách hiệu quả hơn.
* **Ghép tiêu đề và nội dung:** Kết hợp title và text vào một trường duy nhất giúp mô hình có đầy đủ ngữ cảnh của bài viết để đưa ra quyết định chính xác hơn.

**3. Sức mạnh của các mô hình học sâu**

* **Lớp Embedding:** Đây là lớp đầu tiên và rất quan trọng, nó chuyển đổi từ vựng thành các vector số mật độ cao. Quá trình này giúp mô hình hiểu được ngữ nghĩa và mối quan hệ giữa các từ.
* **Kiến trúc Bi-LSTM:** Đây là yếu tố cốt lõi mang lại hiệu suất cao. Bi-LSTM xử lý dữ liệu tuần tự theo cả hai chiều (tiến và lùi), giúp mô hình nắm bắt được **toàn bộ ngữ cảnh của một câu** (cả quá khứ và tương lai của một từ). Điều này rất quan trọng để nhận diện các sắc thái ngôn ngữ tinh tế của tin giả.
* **Kiến trúc kết hợp CNN-BiLSTM:** Việc kết hợp lớp **CNN** (tích chập) và **Bi-LSTM** đã mang lại hiệu quả vượt trội.
  + **CNN (Conv1D):** Hiệu quả trong việc trích xuất các **đặc trưng cục bộ**, giống như việc nhận diện các n-gram (các chuỗi từ ngắn) mang ý nghĩa nhất. Điều này giúp mô hình phát hiện nhanh các cụm từ quan trọng hoặc các mẫu từ lặp lại.
  + **Bi-LSTM:** Tiếp nhận các đặc trưng đã được CNN trích xuất và xử lý các phụ thuộc dài hạn trong toàn bộ văn bản.

Sự kết hợp này cho phép mô hình tận dụng cả khả năng nhận diện mẫu cục bộ của CNN và khả năng hiểu ngữ cảnh toàn diện của Bi-LSTM, tạo nên một kiến trúc vừa mạnh mẽ vừa hiệu quả về mặt tính toán.

**Tổng kết chương 4:** Trong chương này, đề tài đã trình bày quá trình thu thập và xử lý dữ liệu từ tập Fake News Detection, cấu hình môi trường thực nghiệm và các mô hình học sâu được triển khai gồm LSTM, Bi-LSTM và CNN-BiLSTM. Kết quả đánh giá trên tập kiểm tra cho thấy cả ba mô hình đều đạt hiệu suất cao, với F1-score và AUC gần tiệm cận mức tối đa. Đặc biệt, mô hình Bi-LSTM và CNN-BiLSTM cho kết quả vượt trội hơn so với LSTM truyền thống. Tuy nhiên, độ chính xác quá cao cũng là dấu hiệu tiềm ẩn của hiện tượng overfitting, do đó cần được kiểm chứng thêm bằng các kỹ thuật đánh giá chéo và kiểm soát quá trình huấn luyện. Những kết quả đạt được trong chương này là cơ sở quan trọng để đưa ra đánh giá tổng thể và định hướng phát triển tiếp theo cho hệ thống trong chương kết luận.

# CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong luận văn này, chúng tôi đã thực hiện nghiên cứu xây dựng ứng dụng mua bán trực tuyến ngoại tuyến đầu tiên. Ứng dụng hoạt động khi không có Internet và đồng bộ khi có Internet trở lại. Sau đó dữ liệu được thể hiện qua giao diện người dùng và người dùng thực hiện mua bán trực tuyến trên ứng dụng này.

## Kết luận.

Đề tài đã nghiên cứu và ứng dụng các mô hình học sâu gồm LSTM, Bi-LSTM và CNN-BiLSTM để giải quyết bài toán phát hiện tin giả trên tập dữ liệu Fake News Detection. Qua quá trình huấn luyện và đánh giá mô hình, kết quả cho thấy các mô hình đều đạt độ chính xác và hiệu suất cao, đặc biệt là Bi-LSTM và CNN-BiLSTM với AUC và F1-score gần như tuyệt đối.

Việc áp dụng các kỹ thuật xử lý ngôn ngữ tự nhiên kết hợp với mô hình học sâu đã chứng minh tính hiệu quả trong việc phát hiện và phân loại tin giả, góp phần giảm thiểu sự lan truyền thông tin sai lệch trên môi trường số.

* + 1. Hạn chế.

- Chỉ mới sử dụng một tập dữ liệu duy nhất (Fake News Detection), dẫn đến tính tổng quát chưa cao khi áp dụng cho các nguồn dữ liệu khác.

- Chưa áp dụng các kỹ thuật xử lý nâng cao như embedding ngữ cảnh (BERT, RoBERTa).

- Chưa tích hợp mô hình vào một hệ thống hoặc ứng dụng hoàn chỉnh để kiểm tra khả năng vận hành thực tế.

* + 1. Hướng phát triển.

- Mở rộng nghiên cứu trên nhiều tập dữ liệu khác nhau để tăng tính khái quát và đánh giá hiệu năng mô hình trong các bối cảnh khác nhau.

- Kết hợp các mô hình ngôn ngữ hiện đại như BERT, RoBERTa, hoặc transformer-based models để cải thiện độ chính xác.

- Phát triển một hệ thống phát hiện tin giả hoàn chỉnh dưới dạng ứng dụng web hoặc API, hỗ trợ kiểm tra nhanh nội dung tin tức.

- Áp dụng kỹ thuật kiểm tra chéo (cross-validation) và tối ưu hóa tham số để tránh hiện tượng overfitting.

# Tài liệu tham khảo

Tìm hiểu về học sâu và mạng nơ-ron hồi tiếp (RNN, LSTM, Bi-LSTM).  
 <https://colah.github.io/posts/2015-08-Understanding-LSTMs/>   
 <https://machinelearningmastery.com/understanding-simple-recurrent-neural-networks-in-python/>   
 <https://towardsdatascience.com/understanding-bidirectional-lstm-and-its-implementation-in-keras-4ca496dfbfae>

Tìm hiểu về CNN kết hợp LSTM trong xử lý ngôn ngữ tự nhiên.  
 <https://towardsdatascience.com/combining-cnn-and-lstm-for-text-classification-836f3c2e1c2e>   
 <https://www.analyticsvidhya.com/blog/2021/06/text-classification-using-cnn-bilstm/>

Tìm hiểu về các thư viện TensorFlow và Keras để xây dựng mô hình học sâu.  
 <https://www.tensorflow.org/>   
 <https://keras.io/>   
 <https://github.com/fchollet/keras>

Tài liệu về tập dữ liệu LIAR và ISOT dùng trong phát hiện tin giả.  
 <https://www.cs.ucsb.edu/~william/data/liar_dataset.zip>   
 <https://www.uvic.ca/> (ISOT Fake News Dataset)

Các bài nghiên cứu liên quan đến phát hiện tin giả.  
 <https://aclanthology.org/D17-1317.pdf>   
 <https://arxiv.org/abs/1708.07104>

Tìm hiểu về các phép đánh giá mô hình phân loại (Accuracy, Precision, Recall, F1, AUC).  
 <https://scikit-learn.org/stable/modules/model_evaluation.html>   
 <https://en.wikipedia.org/wiki/Precision_and_recall>

Tìm hiểu về xử lý dữ liệu văn bản và biểu diễn từ (Word Embedding, Tokenization).  
 <https://www.tensorflow.org/text/guide/word_embeddings>   
 <https://www.nltk.org/>   
 <https://huggingface.co/docs/tokenizers/index>