PHÂN LỚP ĐA NHÃN CÁC NGÀNH NGHỀ CÓ LIÊN QUAN TỪ CÁC MÔ TẢ CÔNG VIỆC

1st Nguyễn Huỳnh Vương Quốc
Đại học Quốc gia TP.HCM
Đại học Công nghệ Thông tin
Khoa Khoa học và Kỹ thuật Thông tin
20521813
20521813@gm.uit.edu.vn

2nd Lê Văn Anh Tài
Đại học Quốc gia TP.HCM
Đại học Công nghệ Thông tin
Khoa Khoa học và Kỹ thuật Thông tin
20522216
20522216@gm.uit.edu.vn

3rd Lê Tuấn Cường
Đại học Quốc gia TP.HCM
Đại học Công nghệ Thông tin
Khoa Khoa học và Kỹ thuật Thông tin
20520146
20520146@gm.uit.edu.vn

***Tóm tắt nội dung*—Đối với các nhà tuyển dụng, việc đăng tin tuyển dụng để tìm ứng viên phù hợp và chính xác là một việc hết sức cần thiết. Tuy nhiên, các mô tả công việc thường được gán với nhiều ngành nghề và lĩnh vực khác nhau. Điều này làm cho các nhà tuyển dụng dễ mắc các thiếu sót trong việc đánh dấu các ngành nghề liên quan đến công việc đó khi đăng tin tuyển dụng. Do vậy, trong bài báo cáo lần này chúng tôi sẽ xây dựng mô hình phân lớp đa nhãn sử dụng các mô hình học máy như Linear Regression, Support Vector Machine, SGD, các mô hình học sâu như TextCNN, Bi-LSTM, Bi-GRU kết hợp với các mô hình pretrained gồm các biến thể của BERT bao gồm phoBERT, XLMBERT và distilBERT. Việc xây dựng các mô hình phân lớp này nhằm mục đích giúp các nhà tuyển dụng tự động hóa được quá trình lựa chọn ra các ngành nghề và lĩnh vực có liên quan nhất ứng với từng mô tả công việc khác nhau. Kết quả tốt nhất chúng tôi đạt được đến từ mô hình Bi-GRU kết hợp XLMBert cho Hamming score, EM score và F1-score lần lượt là 57%, 38% và 64% trên tập test. Ngoài ra, với kết quả thực nghiệm này chúng tôi còn phân tích lỗi ở các mô hình nhằm cải thiện hiệu suất trong tương lai.**

***Từ khóa*—mô tả công việc, phân lớp đa nhãn, mô hình transformer, Neural Network, TextCNN, Bi-LSTM, Bi-GRU**

I. Giới thiệu

Với sự phát triển vượt bậc của Công nghệ Thông tin và các lĩnh vực liên quan, việc tìm kiếm việc làm thông qua internet đã là điều quá bình thường đối với người tìm việc nói chung và sinh viên nói riêng. Một điều không thể trách khỏi là lượng thông tin trong các mô tả công việc là vô cùng đa dạng và có thể diễn tả theo nhiều cách thức khác nhau. Đối với các nhà tuyển dụng, việc đăng tin tuyển dụng để tìm ứng viên phù hợp và chính xác là một việc hết sức cần thiết. Tuy nhiên, các mô tả công việc thường được gán với nhiều ngành nghề và lĩnh vực khác nhau. Điều này làm cho các nhà tuyển dụng dễ mắc các thiếu sót trong việc đánh dấu các ngành nghề liên quan đến công việc đó khi đăng tin tuyển dụng. Do đó, chúng ta cần các mô hình giúp phân loại đa nhãn có khả năng đưa ra ngành nghề hoặc các lĩnh vực có liên quan đối với thông tin trên từng mô tả công việc. Điều này làm giảm đáng kể thời gian xem xét, chọn lọc và đăng tin tuyển dụng của các nhà tuyển dụng.

Bài toán phân loại mô tả công việc thành các lĩnh vực, ngành nghề là một bài toán phân loại văn bản (Text Classification). Đây là bài toán ứng dụng Học máy (Machine Learning) và Xử lý ngôn ngữ tự nhiên (Natural Language Processing) để dự đoán các ngành nghề, lĩnh vực dựa vào từng thông tin trong mô tả công việc như kiến thức, kỹ năng, sở thích,... Tuy nhiên, đây là một bài toán phân lớp đa nhãn (Multi-label Classification) vì các mô tả công việc thường sẽ có một hoặc nhiều ngành nghề tương ứng có liên quan nhất. Bài toán có đầu vào là một văn bản chứa nội dung của mô tả công việc, sau đó mô hình sẽ đưa ra dự đoán về các ngành nghề, lĩnh vực có liên quan nhất đối với mô tả công việc đó. Đầu ra có thể là một ngành nghề, một lĩnh vực hoặc có thể có nhiều ngành nghề, nhiều lĩnh vực khác nhau.

II. Bộ dữ liệu

A. Thu thập dữ liệu

Chúng tôi thu thập dữ liệu ở dạng văn bản từ hai trang tìm kiếm việc làm online: Vietnamworks và TopCV. Dữ liệu chúng tôi đã thu thập bao gồm mô tả công việc, yêu cầu của công việc, và các lĩnh vực trực tiếp liên quan đến công việc đó (có thể là một ngành nghề hoặc nhiều ngành nghề). Bảng VI chứa tất cả 74 các nhãn ngành nghề có trong tập dữ liệu. Tập dữ liệu sau khi được thu thập có 40090 dòng.

B. Phân tích thăm dò dữ liệu

Sau khi thu thập dữ liệu chúng tôi tiến hành phân tích thăm dò dữ liệu nhằm tìm ra các insights của dữ liệu. Hình 2 thể hiện số lần xuất từng ngành nghề, lĩnh vực trên toàn bộ tập dữ liệu. Chúng ta có thể thấy rằng đây là một bộ dữ liệu bị mất cân bằng nhãn. Các ngành phổ biến trên thị thường công việc hiện nay như Công nghệ Thông tin, Marketing, . . . sẽ xuất hiện nhiều hơn các ngành ít phổ biến trên thị trường công việc như Thể thao – thể hình, Khai thác khoáng sản,... Hình 1 là một phiên bản khác để thể hiện tần suất xuất hiện của các ngành nghề, lĩnh vực theo dạng wordmap.

Hình 1. Wordmap các ngành nghề, lĩnh vực trong tập dữ liệu

C. Tiền xử lý dữ liệu

Sau khi thu thập dữ liệu hoàn chỉnh 40090 dòng dữ liệu, chúng tôi tiến hành tiền xử lý để xây dựng và chuẩn bị dữ liệu cho quá trình huấn luyện mô hình. Các bước tiền xử lý bao gồm:

- Xóa tất cả các dòng tiếng Anh đi (1).
- Lowercase (2).
- Tách từ bằng công cụ **VNCoreNLP** (3).
- Xóa tất cả ký tự đặc biệt (4)..
- Xóa tất cả các stopword của tiếng Việt (5).

Ở bước (1), chúng tôi xóa đi tất cả các dòng dữ liệu chỉ có mô tả bằng tiếng Anh. Sau bước này, dữ liệu chỉ còn lại khoảng 34000 mẫu dữ liệu.

Ở bước (2), chúng tôi còn lower case tất cả các dòng dữ liệu vì ngữ nghĩa giữa các từ viết hoa và viết thường không có nhiều khác biệt nhưng nếu mô hình phải xử lý cả hai thì làm tăng độ phức tạp cho mô hình.

Tiếp theo ở bước (3), chúng tôi xóa tất cả các ký tự đặc biệt. Những thành phần này là các phần không quan trọng và có khả năng gây nhiễu đối với mô hình. Tuy nhiên, dấu chấm câu "." với các văn bản gồm nhiều câu thì đây là cách thức nhận biết đâu là kết câu hoặc các trường hợp đặc biệt khác.

Ở bước (4), chúng tôi xóa tất cả các stopwords của tiếng Việt. Việc xóa các stopwords giúp cho mô hình được đơn giản hóa, không cần xử lý và loại các yếu tố ít liên quan hoặc ít ảnh hưởng đến kết quả đầu ra.

Chúng tôi sẽ áp dụng quy trình tiền xử lý này cho cả hai thông tin dạng văn bản là mô tả công việc (job description) và yêu cầu công việc (job requirement) sau đó gộp cả hai thành một cột - gọi đơn giản là thông tin công việc (information) để tránh việc bỏ sót thông tin.

D. Chuẩn bị dữ liệu

Chúng tôi phân chia ngẫu nhiên tập dữ liệu thành các tập training, validation và test với tỉ lệ lần lượt là 70%, 10% và 20%. Bảng I thể hiện chi tiết về số lượng mẫu trong cả ba tập.

Bảng I
SỐ LƯỢNG MẪU THEO TỪNG TẬP DỮ LIỆU

	Số lượng mẫu
Tập training	23978
Tập validation	6851
Tập test	3426

Ngoài ra, để chọn ra các siêu tham số phù hợp cho mô hình, chúng tôi đã xem xét tới phân phối về độ dài của mô tả công việc đối với từng tập dữ liệu. Độ dài của từng mô tả công việc được tính theo từng từ trong tiếng Việt. Bảng II thể hiện chi tiết thông tin này. Chúng ta có thể thấy rằng độ dài mô tả công việc trong cả ba tập dữ liệu này khá tương đồng, mặc dù độ dài tối đa trên tập training có phần thấp hơn so với hai tập còn lại. Hình 3 thể hiện phân phân độ dài mô tả công việc trên cả ba tập dưới dạng boxplot.

Hơn nữa, bởi vì bài toán chúng tôi đang giải quyết là bài toán phân lớp đa nhãn, nên chúng ta cũng cần phải quan tâm đến số lượng nhãn cho từng mô tả công việc được phân bố ra sao. Bảng III tóm tắt phân phối số lượng nhãn đối với từng mô tả công việc trên cả ba tập. Chúng ta có thể dễ dàng thấy rằng, các mô tả công việc chỉ có 1 nhãn là nhiều nhất trên cả ba tập. Điều này chứng minh rằng các mô tả công việc có số lượng nhãn càng nhiều thì chiếm thiểu số trong tập dữ liệu. Bên cạnh đó, số nhãn tối đa của một mô tả công việc có thể có là 8 nhãn. Các mô tả công việc gồm có 8 nhãn chỉ xuất hiện trên tập dữ liệu

Bảng II
THỐNG KÊ ĐỘ DÀI TỐI ĐA, ĐỘ DÀI TỐI THIỂU, ĐỘ DÀI TRUNG BÌNH CỦA CÁC MÔ TẢ CÔNG VIỆC THEO TỪNG TẬP DỮ LIỆU

	Max	**Min**	**Avg**
Tập traning	1460	20	187.7
Tập validation	1520	22	185.6
Tập test	1513	20	190

training. Để thêm phần trực quan, hình 4 mô tả số lương nhãn đối với từng mô tả công việc trên các tập training, validation và test.

III. PHƯƠNG PHÁP TIẾP CẬN

A. Mô hình

Trong đồ án này, chúng tôi thực nghiệm trên các mô hình học máy đơn giản như: Linear Regression, Support Vector Machine, Stochastic Gradient Descent và các mô hình học sâu được ứng dụng tốt cho dữ liệu dạng chuỗi như: Neural Network, TextCNN, Bidirectional LSTM, Bidirectional GRU. Các mô hình học máy trên cho kết quả về thời gian huấn luyện khá tốt, mặc khác các mô hình học sâu thì lại vượt trội về mặt xử lý ngữ nghĩa và các mối quan hệ trong câu.

1) ***Logistic Regression****:* Logistic Regression là một thuật toán phân loại trong Machine Learning, được sử dụng để giải quyết bài toán phân lớp nhị phân, đa lớp và đa nhãn. Trên cơ sở dữ liệu đầu vào, mô hình Logistic Regression xác định xác suất của một mẫu thuộc vào từng nhãn trong tập nhãn đã cho. Thuật toán Logistic Regression dựa trên hàm sigmoid để biểu diễn quyết định phân lớp. Đầu tiên, mô hình tính tổng trọng số của các đặc trưng của mẫu đầu vào, sau đó áp dụng hàm sigmoid để chuyển đổi tổng trọng số thành xác suất. Hàm sigmoid chuyển đổi giá trị đầu vào thành giá trị nằm trong khoảng từ 0 đến 1, thể hiện xác suất thuộc về một nhãn cụ thể. Mô hình Logistic Regression được sử dụng trong bài toán phân lớp đa nhãn khi có nhiều hơn hai nhãn được dự đoán. Trong trường hợp này, ta áp dụng phương pháp One-vs-All hoặc One-vs-Rest, trong đó ta huấn luyện một mô hình Logistic Regression cho mỗi nhãn riêng biệt. Mỗi mô hình sẽ dự đoán xác suất thuộc về nhãn đó hoặc không thuộc về nhãn đó. Cuối cùng, ta chọn nhãn có xác suất dự đoán cao nhất là nhãn được dự đoán cho mẫu đầu vào.

2) ***SGD (Stochastic Gradient Descent)****:* SGD (Stochastic Gradient Descent) là một thuật toán tối ưu hóa được sử dụng trong Machine Learning, đặc biệt là trong bài toán phân lớp đa nhãn. Thuật toán này giúp tìm ra bộ trọng số tối ưu của mô hình phân loại bằng cách tối thiểu hóa hàm mất mát dựa trên các mẫu dữ liệu huấn luyện. Trong bài toán phân lớp đa nhãn, SGD được áp dụng để điều chỉnh trọng số của mô hình phân loại. Thuật toán này sử dụng một mẫu dữ liệu ngẫu nhiên từ tập huấn luyện để tính toán gradient của hàm mất mát. Gradient này biểu thị hướng và độ lớn của sự thay đổi trong hàm mất mát khi thay đổi các trọng số. Sau đó, SGD cập nhật các trọng số của mô hình dựa trên gradient tính toán được. SGD là một thuật toán linh hoạt và phổ biến trong Machine Learning, đặc biệt là khi có số lượng lớn mẫu dữ liệu. Với khả năng tối ưu hóa hàm mất mát một cách hiệu quả, SGD được sử dụng rộng rãi trong bài toán phân lớp đa nhãn để tìm ra bộ trọng số tối ưu cho mô hình phân loại.

3) ***SVM (Support Vector Machine)****:* (Support Vector Machine) là một thuật toán phân loại mạnh mẽ được sử dụng trong bài toán phân lớp đa nhãn. SVM xây dựng một siêu phẳng (hyperplane) trong không gian đặc trưng để phân tách các điểm dữ liệu thuộc các nhãn khác nhau. Mục tiêu của SVM là tìm ra siêu phẳng tối ưu sao cho khoảng cách giữa siêu phẳng và các điểm dữ liệu gần nhất của các lớp là lớn nhất. Trong bài toán phân lớp đa nhãn, SVM được mở rộng để xử lý việc phân loại cho nhiều nhãn cùng một lúc. Điều này được thực hiện bằng cách xây dựng các bộ phân loại nhị phân cho mỗi cặp nhãn. Ví dụ, nếu có N nhãn, thì SVM sẽ tạo *N*(N-1)/2* bộ phân loại nhị phân để phân loại giữa các cặp nhãn khác nhau. Các bộ phân loại nhị phân sử dụng trong SVM được xây dựng dựa trên một số mẫu dữ liệu hỗ trợ (support vectors) nằm gần với siêu phẳng. Những mẫu dữ liệu hỗ trợ này đóng vai trò quan trọng trong việc xác định vị trí và hình dạng của siêu phẳng. SVM cố gắng tìm ra siêu phẳng tối ưu mà tối đa hóa khoảng cách từ các mẫu dữ liệu hỗ trợ đến siêu phẳng, đồng thời giới hạn việc phân loại sai lệch (margin) của các điểm dữ liệu.

4) ***TextCNN****:* TextCNN là một kiến trúc mạng nơ-ron tích chập được sử dụng cho bài toán phân lớp văn bản có thể ứng dụng cho bài toán phân lớp đa nhãn. Kiến trúc này có khả năng mô hình hóa cấu trúc và ý nghĩa của văn bản thông qua việc áp dụng các bộ lọc tích chập trên đầu vào văn bản. TextCNN chủ yếu bao gồm ba phần chính: lớp tích chập (convolutional layer), lớp pooling và lớp kết nối đầy đủ (fully connected layer). Trong lớp tích chập, các bộ lọc có kích thước nhỏ được áp dụng trên từng phần của văn bản để tìm ra các đặc trưng cục bộ. Sau đó, lớp pooling được sử dụng để lấy thông tin quan trọng từ các đặc trưng đã tìm được. Cuối cùng, thông tin được đưa vào lớp kết

Bảng III
SỐ LƯỢNG NHÃN ĐỐI VỚI TỪNG MÔ TẢ CÔNG VIỆC TRÊN TỪNG TẬP DỮ LIỆU

	1 nhãn	**2 nhãn**	**3 nhãn**	**4 nhãn**	**5 nhãn**	**6 nhãn**	**7 nhãn**	**8 nhãn**
Training	11147	7391	5030	339	60	8	1	2
Validation	3115	2154	1462	100	19	1	0	0
Test	1583	1039	744	51	8	1	0	0

nối đầy đủ để phân loại văn bản vào các nhãn tương ứng. TextCNN có thể áp dụng cho bài toán phân lớp đa nhãn bằng cách sử dụng kỹ thuật mã hóa one-hot cho nhãn đầu ra. Mỗi nhãn được biểu diễn dưới dạng vector one-hot, sau đó được đưa vào lớp kết nối đầy đủ để dự đoán xác suất của mỗi nhãn. Khi huấn luyện, mô hình sẽ tối ưu hóa hàm mất mát (loss function) để đạt được kết quả phân loại tốt nhất cho các nhãn đầu ra.

*5) **Bi-LSTM**:* Bi-LSTM là một kiến trúc mạng nơ-ron sử dụng trong việc xử lý dữ liệu dạng sequence, có thể ứng dụng vào bài toán phân lớp đa nhãn văn bản trong xử lý ngôn ngữ tự nhiên. Nó kết hợp cả khả năng ghi nhớ thông tin lâu dài (long-term memory) và khả năng học thông tin ngắn hạn (short-term memory) của mạng LSTM với khả năng mô hình hóa thông tin văn bản theo cả hai hướng (trước và sau) thông qua việc sử dụng hai bộ LSTM song song. Bi-LSTM hoạt động bằng cách xử lý dữ liệu văn bản theo cả hai hướng: từ trái sang phải và từ phải sang trái. Điều này cho phép nó hiểu được ngữ cảnh từ cả hai phía của từ hoặc cụm từ trong văn bản. Mỗi bộ LSTM trong Bi-LSTM sẽ ghi nhớ thông tin từ cả hai phía và truyền thông tin này qua lớp kết nối đầy đủ (fully connected layer) để phân loại văn bản vào các nhãn tương ứng.

*6) **Bi-GRU**:* Bi-GRU có một kiến trúc tương tự với Bi-LSTM nhưng đơn giản việc tính toán nhưng vẫn giữ được hiệu suất. LSTM có khả năng ghi nhớ thông tin từ quá khứ trong quá trình xử lý dữ liệu chuỗi. Nó sử dụng các cổng (gates) để kiểm soát luồng thông tin và xử lý vấn đề biến mất gradient trong mạng nơ-ron hồi quy. Trong khi đó, Bi-GRU sử dụng GRU (Gated Recurrent Unit) làm đơn vị cơ bản. GRU cũng có khả năng ghi nhớ thông tin từ quá khứ như LSTM, nhưng nó có một số thay đổi trong cấu trúc. GRU chỉ sử dụng hai cổng - cổng cập nhật (update gate) và cổng khôi phục (reset gate), giúp giảm độ phức tạp tính toán và số lượng tham số so với LSTM.

B. Trích xuất đặc trưng

Vì dữ liệu của chúng ta đang ở dạng văn bản cho nên chúng ta không thể trực tiếp đưa vào mô hình huấn luyện được mà cần phải qua một bước trích xuất đặc trưng hay còn gọi là quá trình ánh xạ chữ sang dạng số. Biểu diễn vector của các dữ liệu dạng văn bản đóng vai trò quan trọng trong các bài toán NLP có thể nâng cao hiệu suất của các mô hình phân loại bằng cách tăng khả năng nắm bắt và hiểu các từ trong ngữ cảnh.

Đối với các mô hình học máy, để đơn giản hóa cho bước này và quá trình huấn luyện, chúng tôi sử dụng phương pháp **TF-IDF**, đây là một phương pháp dựa trên thống kê.

Đối với các mô hình học sâu, chúng tôi sử dụng các mô hình pretrained để có thể biểu diễn các câu ở dạng số có ngữ cảnh và mang các thông tin quan trọng hơn. Ở đây, chúng tôi sử dụng 3 biến thể của BERT:

- **phoBERT**
- **XLMBERT**
- **distilBERT**

*1) **TF-IDF (Term Frequency-Inverse Document Frequency)**:* Là một phương pháp tính toán trọng số cho các từ trong một văn bản dựa trên tần suất xuất hiện của từ đó trong văn bản và trong toàn bộ tập dữ liệu. TF-IDF được sử dụng rộng rãi trong xử lý ngôn ngữ tự nhiên và trích xuất thông tin văn bản. Trọng số TF-IDF cho một từ trong một văn bản được tính bằng cách kết hợp hai thành phần:

- **Term Frequency (TF)**: Đây là đo lường tần suất xuất hiện của từ trong văn bản. Giá trị TF cao cho biết từ đó xuất hiện nhiều trong văn bản.
- **Inverse Document Frequency (IDF)**: Đây là đo lường độ quan trọng của từ đó trong toàn bộ tập dữ liệu. Giá trị IDF cao cho biết từ đó xuất hiện ít trong các văn bản khác.

$$W_{x,y} = tf_{x,y} \times log\frac{N}{df_x}$$

*2) **phoBERT**:* Là một mô hình ngôn ngữ tiếng Việt được huấn luyện trước với hai phiên bản "base" và "large" là các mô hình ngôn ngữ quy mô lớn đầu tiên được huấn luyện trước cho tiếng Việt. Đây là một pretrained model được huấn luyện monolingual language, tức là chỉ huấn luyện dành riêng cho tiếng Việt. Việc huấn luyện dựa trên

kiến trúc và cách tiếp cận giống RoBERTa của Facebook được Facebook giới thiệu giữa năm 2019. PhoBERT được train trên khoảng 20GB dữ liệu bao gồm khoảng 1GB Vietnamese Wikipedia corpus và 19GB còn lại lấy từ Vietnamese news corpus. PhoBERT có thể được sử dụng để giải quyết các vấn đề trong lĩnh vực xử lý ngôn ngữ tự nhiên như phân loại văn bản như phân loại văn bản, nhận diện cảm xúc văn bản, tóm tắt văn bản và nhiều hơn nữa.

3) ***XLMBERT****:* Là một mô hình ngôn ngữ đa ngôn ngữ được đào tạo trước bằng cách sử dụng hai mục tiêu học có giám sát: Causal Language Modeling (CLM) và Masked Language Modeling (MLM). Mục tiêu CLM là dự đoán từ tiếp theo trong một câu dựa trên các từ trước đó. Mục tiêu MLM là dự đoán các từ bị che trong một câu dựa trên ngữ cảnh xung quanh. Bằng cách kết hợp hai mục tiêu này, XLMBERT có thể học được các đặc trưng ngôn ngữ chung và cụ thể cho từng ngôn ngữ. XLMBert có thể được sử dụng để giải quyết nhiều nhiệm vụ xử lý ngôn ngữ tự nhiên khác nhau, như phân loại văn bản, dịch máy, trả lời câu hỏi và suy luận ngôn ngữ.

4) ***distilBERT****:* Là một mô hình Transformer nhỏ, nhanh, rẻ và nhẹ dựa trên kiến trúc BERT. DistilBERT được sử dụng trong các ứng dụng xử lý ngôn ngữ tự nhiên như phân loại văn bản, dịch máy, tóm tắt văn bản và phát hiện ngôn ngữ tục tĩu. Nó sử dụng kỹ thuật gọi là distillation để giảm kích thước của một mô hình BERT lên đến 40%. So với BERT-base, DistilBERT nhỏ hơn 40% và nhanh hơn 60%, tất cả trong khi vẫn giữ được hơn 95% hiệu suất của BERT. DistilBERT học một phiên bản xấp xỉ của BERT, giữ lại 97% hiệu quả dự đoán nhưng chỉ sử dụng một nửa tham số.

C. Cài đặt thực nghiệm

Đối với việc tiền xử lý dữ liệu, chúng tôi sử dụng công cụ **VNCoreNLP**. Đây là công cụ tách từ cho tiếng Việt đem lại hiệu suất cao.

Đối với các mô hình học máy, chúng tôi sẽ áp dụng TF-IDF để trích xuất đặc trưng. Số lượng từ tối đa trong từ điển của TF-IDF là 5000. Từ điển này bao gồm 5000 từ phổ biến nhất trong toàn bộ tập dữ liệu. Chúng tôi sử dụng phương pháp phân tích câu thành các từ riêng lẻ để tính tần suất và trọng số của từng từ.

Đối với các mô hình học sâu, chúng tôi sẽ thực hiện một chuỗi các thực nghiệm bằng cách fine-tuning trên các biến thể của BERT bao gồm: phoBERT, XLMBERT và distilBERT (đều được hỗ trợ bởi Huggingface) kết hợp với các mô hình như: MLP, TextCNN, Bi-LSTM và Bi-GRU để so sánh kết quả giữa chúng. Độ dài tối đa của câu đầu vào là 200 (200 tokens), 768 đơn vị tính toán cho cả Bi-LSTM và Bi-GRU. Tất cả các bộ tham số cần thiết của mô hình đều được áp dụng phương thức khởi tạo Uniform Xavier. Hàm mất mát, thuật toán tối ưu, learning rate được chúng tôi sử dụng lần lượt là Binary Crossentropy, Adam và 2e-5. Ngoài ra, đối với learning rate chúng tôi sử dụng thêm phương pháp điều chỉnh tốc độ học *Linear Learning Rate Scheduler with Warmup* nhằm cân bằng tốc độ học và số bước huấn luyện, đồng thời giúp thuật toán nhanh hội tụ. Tất cả các mô hình trên đều được huấn luyện với batch-size là 32 với 10 epochs.

D. Phương pháp đánh giá

1) ***Precision, Recall, F1-score****:* Trong tác vụ phân loại đa lớp hoặc đa nhãn, các khái niệm về độ đo precision, recall và f1-score có thể được áp dụng độc lập cho từng nhãn. Như vậy đối với mỗi lớp và mỗi độ đo sẽ ứng với một giá trị khác nhau. Có một số cách để kết hợp các kết quả đánh giá khác nhau trên các nhãn bằng cách lấy hàm trung bình trên tất cả các lớp khác nhau:

- **Micro**: Tính các độ đo trên toàn cục bằng cách đếm tổng số các giá trị true positive, true negative, ...
- **Macro**: Tính các độ đo cho từng nhãn và tìm giá trị trung bình không trọng số của chúng. Phương pháp này bỏ qua việc xem xét đến sự mất cân bằng nhãn.
- **Weighted**: Tính các độ đo cho từng nhãn và tìm trọng số trung bình của chúng theo trọng số support (số lượng mẫu đối với từng nhãn). Vì vậy, ngược lại với phương pháp **macro** thì phương pháp có xem xét đến sự mất cân bằng nhãn.
- **Samples**: Tính các độ đo cho từng mẫu dữ liệu và tìm giá trị trung bình (cách này chỉ có ý nghĩa trong việc phân lớp đa nhãn).

2) ***Exact Match Ratio****:* Exact Match Ratio là một độ đo được sử dụng trong các bài toán phân loại đa nhãn. EMR đo lường tỷ lệ các dự đoán chính xác trên toàn bộ các mẫu. EMR được tính bằng số lượng các mẫu mà dự đoán chính xác trên tổng số lượng mẫu. EMR là một phép đo đơn giản và trực quan, cho biết tỷ lệ dự đoán chính xác trên toàn bộ tập dữ liệu.

$$\text{EMR} = \frac{\text{Số lượng mẫu dự đoán hoàn toàn chính xác}}{\text{Tổng số lượng mẫu}}$$

3) ***Hamming score****:* Hamming Score là một độ đo đánh giá độ chính xác trong bài toán phân loại đa nhãn. Hamming score đo lường tỷ lệ các nhãn được dự đoán chính xác trên toàn bộ các nhãn có thể có. Để tính Hamming score chúng ta cần tính các giá trị TP, FP và FN trên toàn bộ các nhãn có thể có.

$$\text{Hamming-score} = \frac{\text{TP}}{\text{TP + FP + FN}}$$

IV. Kết quả thực nghiệm và đánh giá

Bảng VII thể hiện toàn bộ kết quả thực nghiệm. Cụ thể, chúng tôi đã tiến hành huấn luyện 15 mô hình bao gồm 3 mô hình học máy và 12 mô hình học sâu khác nhau. Mô hình fine-tuning kết hợp việc sử dụng XLMBERT và Bi-GRU cho kết quả cao nhất với Haming score, EM score, F1-score lần lượt là **0.57**, **0.38** và **0.64**. Hơn nữa đối với các mô hình học máy, SVM cho kết quả EM score là 0.38 khá cao so với các mô hình còn lại. Đối với các mô hình sử dụng BERT, XLMBERT cho kết quả cao nhất, tốt hơn cả phoBERT – là mô hình ngôn ngữ cho tiếng Việt và distilBERT.

V. Phân tích lỗi và hướng phát triển

Bảng V mô tả và thể hiện số lần dự đoán chính xác hoàn toàn của mô hình đối với các mô tả công việc từ 1 đến 6 nhãn bởi vì trên tập dữ liệu test chỉ có tối đa 6 nhãn. Chúng ta có thể thấy rằng với các mô tả công việc chỉ có đơn nhãn thì công việc dự đoán này sẽ có số lần chính xác cao hơn nhiều so với các mô tả công việc có đa nhãn. Điều này chứng tỏ rằng số lần chính xác sẽ tỉ lệ nghịch với số lượng nhãn của mô tả công việc nhất định. Chính thử thách này làm cản trở bài toán phân lớp đa nhãn (multi-label classification) bởi vì mô hình không chỉ cần một dự đoán chính xác mà phải dự đoán chính xác tất cả. Một điều mà chúng ta dễ nhận thấy rằng, đối với các mô hình học máy chúng tôi chỉ sử dụng TF-IDF, đây là phương pháp trích xuất đặc trưng không có ngữ nghĩa nên các mô hình này chỉ tập trung vào việc dự đoán đúng các mô tả công việc chỉ có 1 nhãn. Vì vậy, với các mô tả công việc chỉ có 1 nhãn, số lượng dự đoán chính xác hoàn toàn của mô hình học máy khá cao. Ngược lại, đối với các mô hình học sâu, do sử dụng trích xuất đặc trưng là các biến thể của BERT, nên việc hiểu ngữ nghĩa khá tốt. Vì vậy, các dự đoán chính xác đối với các mô tả công việc có nhiều hơn 1 nhãn là vượt trội hơn hẳn.

Hơn nữa, chúng ta còn có thể nhận biết từ các đầu ra của mô hình rằng các mô hình vẫn chưa hiểu hết mô tả công việc hoặc chưa nhận ra đủ thông tin để đưa ra dự đoán chính xác, một phần vì có quá nhiều ngành có mô tả công việc khá tương tự nhau.

VI. Kết luận

Trong bài báo cáo này, chúng tôi đã trình bày về bài toán phân loại mô tả công việc thành các ngành nghề, lĩnh vực tương ứng. Chúng tôi đã xây dựng một mô hình phân lớp đa nhãn sử dụng các mô hình học máy và học sâu khác nhau, bao gồm Linear Regression, SVM, SGD, TextCNN, Bi-LSTM, Bi-GRU và các biến thể của BERT như phoBERT, XLMBERT và distilBERT.

Qua quá trình thực nghiệm, chúng tôi đã thu thập dữ liệu từ hai trang web tìm việc làm online và tiến hành tiền xử lý dữ liệu để loại bỏ các thành phần không quan trọng và làm tăng hiệu suất của mô hình. Chúng tôi đã thực hiện huấn luyện và đánh giá 15 mô hình khác nhau trên tập dữ liệu, và kết quả tốt nhất đã đạt được từ mô hình kết hợp giữa XLMBERT và Bi-GRU, với Haming score là 0.57, EM score là 0.38 và F1-score là 0.64 trên tập test.

Chúng tôi cũng đã phân tích lỗi của các mô hình và nhận thấy rằng độ chính xác của mô hình tỉ lệ nghịch với số lượng nhãn của mô tả công việc. Các mô hình học máy tập trung vào dự đoán đúng các mẫu chỉ có 1 nhãn, trong khi các mô hình học sâu sử dụng các biến thể của BERT có khả năng hiểu ngữ nghĩa tốt hơn và đạt được độ chính xác cao hơn đối với các mô tả công việc có nhiều hơn 1 nhãn.

Tổng quan, bài báo cáo này xây dựng một mô hình phân lớp đa nhãn cho bài toán phân loại mô tả công việc thành các ngành nghề, lĩnh vực. Các kết quả thực nghiệm cho thấy hiệu suất của các mô hình học sâu vượt trội hơn so với các mô hình học máy truyền thống. Các kết quả này có thể hỗ trợ nhà tuyển dụng trong việc tìm kiếm ứng viên phù hợp với yêu cầu công việc và cải thiện quá trình tuyển dụng trong tương lai.

Tuy nhiên, mô hình còn nhiều khía cạnh để cải thiện. Một trong những khía cạnh quan trọng là tăng kích thước dữ liệu. Mặc dù đã thu thập được một lượng lớn dữ liệu từ hai trang web tìm việc trực tuyến, nhưng việc mở rộng tập dữ liệu huấn luyện có thể đem lại lợi ích đáng kể. Một khía cạnh khác là tối ưu hóa siêu tham số. Trong quá trình huấn luyện, các giá trị siêu tham số ban đầu đã được chọn. Tuy nhiên, việc tìm kiếm các giá trị tối ưu hơn cho các siêu tham số có thể cải thiện hiệu suất của mô hình. Sử dụng các kỹ thuật như lưới tìm kiếm hoặc tối ưu hóa ngẫu nhiên, chúng ta có thể tìm ra các giá trị siêu tham số tốt nhất để đạt được kết quả tối ưu.

Tài liệu

[1] ETNLP: A Toolkit for Extraction, Evaluation and Visualization of Pre-trained Word Embeddings

[2] VnCoreNLP: A Vietnamese natural language processing toolkit

[3] The AI community building the future.

[4] Job Prediction: From Deep Neural Network Models to Applications - Tin Van Huynh1,2,*, Kiet Van Nguyen1,2,† , Ngan Luu-Thuy Nguyen1,2,† , and Anh Gia-Tuan Nguyen1,2,†

[5] Predicting Job Titles from Job Descriptions with Multi-label Text Classification – Hieu Trung Tran, Hanh Hong Phuc Vo, Son Thanh Luu

[6] Multi Label Classification | Solving Multi Label Classification problems (analyticsvidhya.com)

Bảng IV
KẾT QUẢ THỰC NGHIỆM

Mô hình	**Trích xuất đặc trưng**	**Hamming**	**EM**	**F1**	**Precision**	**Recall**
LR	TFIDF	0.47	0.33	0.52	0.61	0.49
SGD	TFIDF	0.46	0.34	0.51	0.60	0.48
SVM	TFIDF	0.54	**0.38**	0.59	0.68	0.57
MLP	phoBERT	0.55	0.37	0.61	**0.69**	0.59
TextCNN	phoBERT	0.54	0.34	0.61	0.68	0.61
Bi-LSTM	phoBERT	0.47	0.32	0.52	0.62	0.49
Bi-GRU	phoBERT	0.55	0.36	0.62	**0.69**	0.61
MLP	XLMBERT	0.55	0.36	0.62	0.68	0.61
TextCNN	XLMBERT	0.56	0.35	0.62	0.67	**0.64**
Bi-LSTM	XLMBERT	0.52	0.36	0.59	0.67	0.56
Bi-GRU	XLMBERT	**0.57**	**0.38**	**0.64**	**0.69**	**0.64**
MLP	distilBERT	0.46	0.30	0.52	0.60	0.50
TextCNN	distilBERT	0.47	0.30	0.53	0.60	0.52
Bi-LSTM	distilBERT	0.42	0.29	0.46	0.55	0.43
Bi-GRU	distilBERT	0.48	0.32	0.54	0.62	0.52

[7] Multi Label Model Evaluation | Kaggle
[8] Metrics for Multilabel Classification | Mustafa Murat ARAT (mmuratarat.github.io)
[9] Multi Label Classification | Solving Multi Label Classification problems (analyticsvidhya.com)

Bảng VI: Tất cả các ngành nghề và lĩnh vực

STT	Tên ngành nghề (tiếng Việt)
1	An ninh - Bảo vệ
2	An toàn lao động
3	Biên phiên dịch
4	Bác sĩ
5	Bán buôn - Bán lẻ - Quản lý cửa hàng
6	Bảo hiểm
7	Bất động sản
8	Chăm sóc khách hàng

Continued on next page

Bảng VI: Tất cả các ngành nghề và lĩnh vực (Continued)

STT	Tên ngành nghề (tiếng Việt)
9	Cơ khí
10	Quản lý điều hành
11	AI - Data Science - Business Intelligence
12	Dược Phẩm - Công nghệ sinh học
13	Dược sĩ
14	Dệt may - Da giày
15	Giáo dục - Đào tạo
16	Hàng hải
17	Hàng không - Du lịch
18	Hành chính - Thư ký
19	Hóa học - Hóa sinh

Continued on next page

Bảng V
SỐ LẦN DỰ ĐOÁN CHÍNH XÁC HOÀN TOÀN XẾP THEO SỐ LƯỢNG NHÃN CỦA TỪNG MÔ TẢ CÔNG VIỆC

Số lượng nhãn	**1**	**2**	**3**	**4**	**5**	**6**
LR	908	141	0	0	0	0
SGD	938	134	0	0	0	0
SVM	**962**	171	0	0	0	0
MLP + phoBERT	941	262	53	0	0	0
TextCNN + phoBERT	878	252	48	1	0	0
Bi-LSTM + phoBERT	946	156	4	0	0	0
Bi-GRU + phoBERT	933	253	50	2	0	0
MLP + XLMBERT	906	287	54	2	0	0
TextCNN + XLMBERT	840	288	**78**	2	0	0
Bi-LSTM + XLMBERT	935	242	42	1	0	0
Bi-GRU + XLMBERT	931	**308**	70	**4**	0	0
MLP + distilBERT	862	179	20	1	0	0
TextCNN + distilBERT	840	184	22	0	0	0
Bi-LSTM + distilBERT	859	137	10	0	0	0
Bi-GRU + distilBERT	879	179	26	0	0	0

Bảng VI: Tất cả các ngành nghề và lĩnh vực (Continued)

STT	Tên ngành nghề (tiếng Việt)
20	IT Phần cứng - Mạng - Viễn Thông
21	IT Phần mềm
22	Internet - Online Media
23	Khai thác năng lượng - Khoáng sản
24	Khoa học - Kỹ thuật
25	Khách sạn - Nhà hàng - Du lịch
26	Kinh doanh
27	Kiến trúc - Thiết kế nội thất
28	Kiểm toán
29	Kế toán

Continued on next page

Bảng VI: Tất cả các ngành nghề và lĩnh vực (Continued)

STT	Tên ngành nghề (tiếng Việt)
30	Marketing
31	Môi trường - Xử lý chất thải
32	Nghề nghiệp khác
33	Ngân hàng
34	Nhân sự
35	Nông - Lâm - Ngư nghiệp
36	Phi chính phủ - Phi lợi nhuận
37	Pháp Lý - Tuân thủ
38	Quản lý dự án
39	Quản lý tiêu chuẩn và chất lượng
40	Quảng cáo - Khuyến mãi
41	Sản phẩm công nghiệp
42	Sản xuất - Lắp ráp - Chế biến
43	Thiết kế - Sáng tạo nghệ thuật
44	Thu mua - Kho Vận - Chuỗi cung ứng
45	Thông tin - Truyền thông - Xuất bản - In ấn
46	Thương mại điện tử
47	Thời trang
48	Thực phẩm - Đồ uống
49	Trình dược viên
50	Tài chính - Đầu tư
51	Tài chính công nghệ
52	Tư vấn
53	Tự động hóa - Ô tô
54	Vận Tải - Lái xe - Giao nhận
55	Vận hành máy - Bảo trì - Bảo dưỡng thiết bị
56	Xuất Nhập Khẩu
57	Xây dựng
58	Y tế - Chăm sóc sức khỏe
59	Điện lạnh - Nhiệt lạnh
60	Điện - Điện tử
61	Bán hàng kỹ thuật

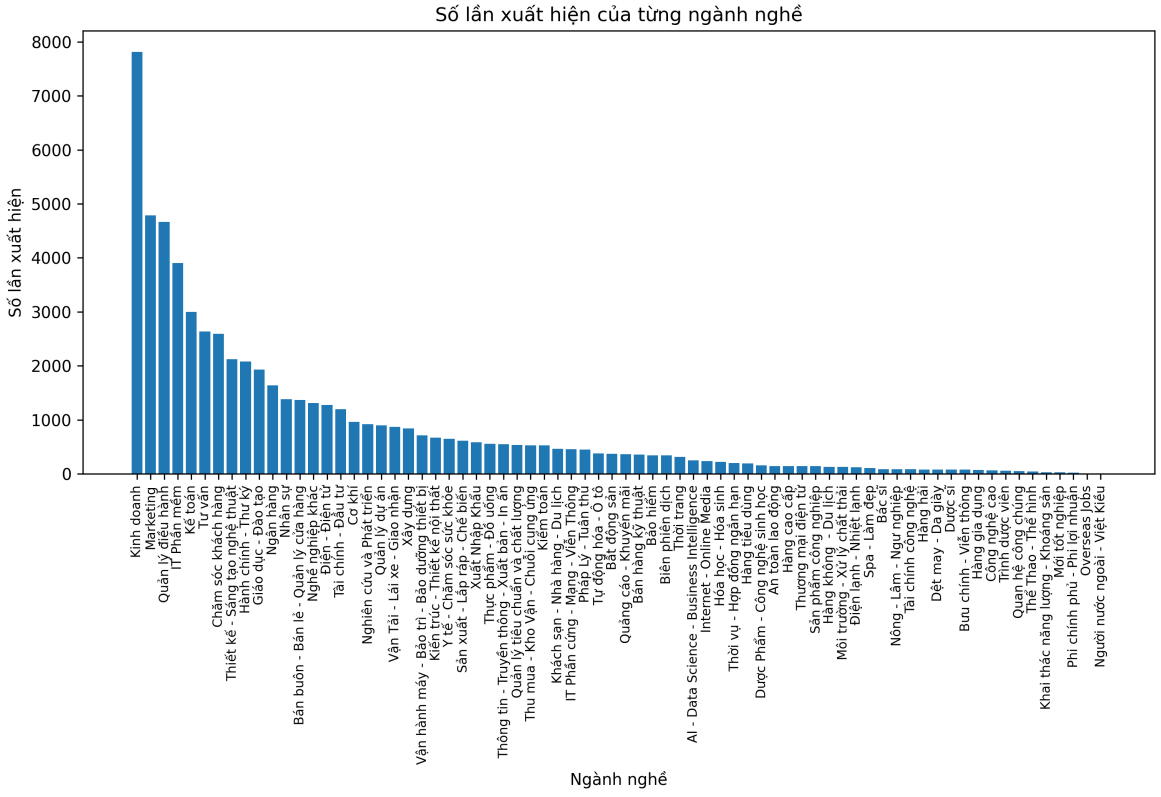
Continued on next page

Bảng VI: Tất cả các ngành nghề và lĩnh vực (Continued)

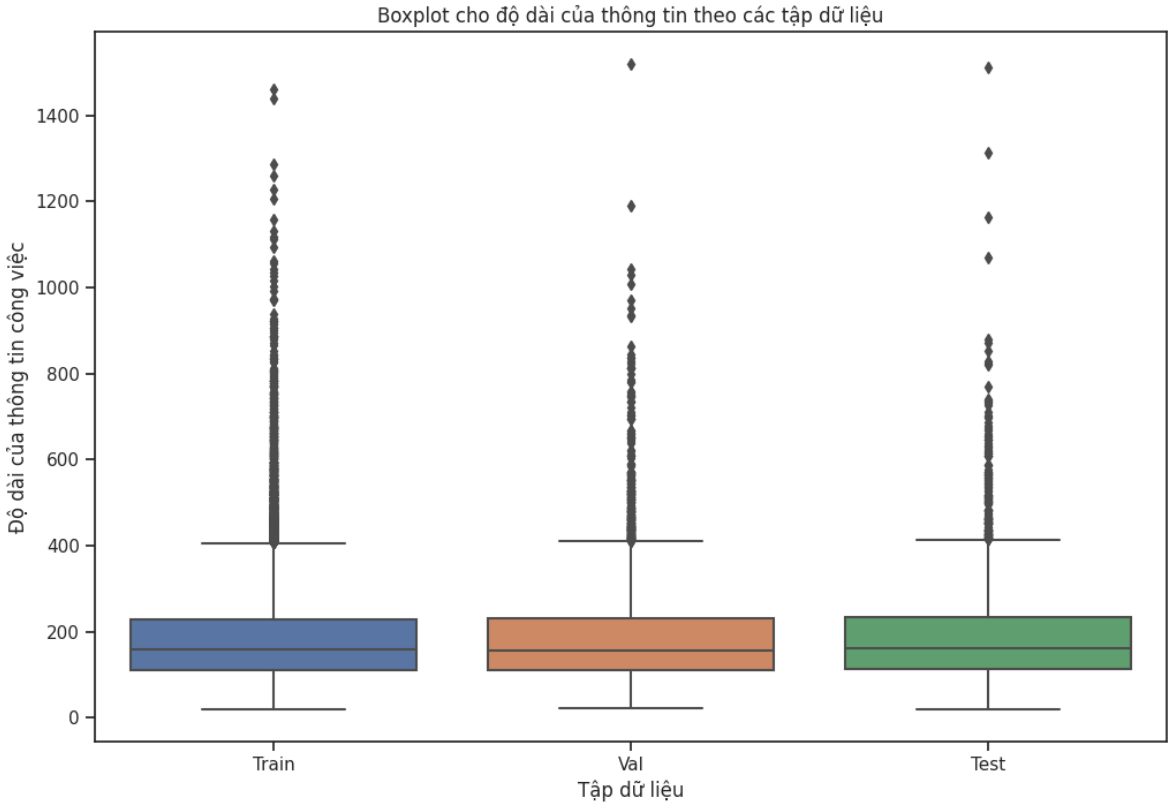
STT	Tên ngành nghề (tiếng Việt)
62	Công nghệ cao
63	Hàng cao cấp
64	Hàng gia dụng
65	Hàng tiêu dùng
66	Mới tốt nghiệp
67	Người nước ngoài - Việt Kiều
68	Overseas Jobs
69	Thời vụ - Hợp đồng ngắn hạn
70	Spa - Làm đẹp
71	Bưu chính - Viễn thông
72	Nghiên cứu và Phát triển
73	Thể Thao - Thể hình
74	Quan hệ công chúng

Bảng VII
PHÂN CÔNG CÔNG VIỆC

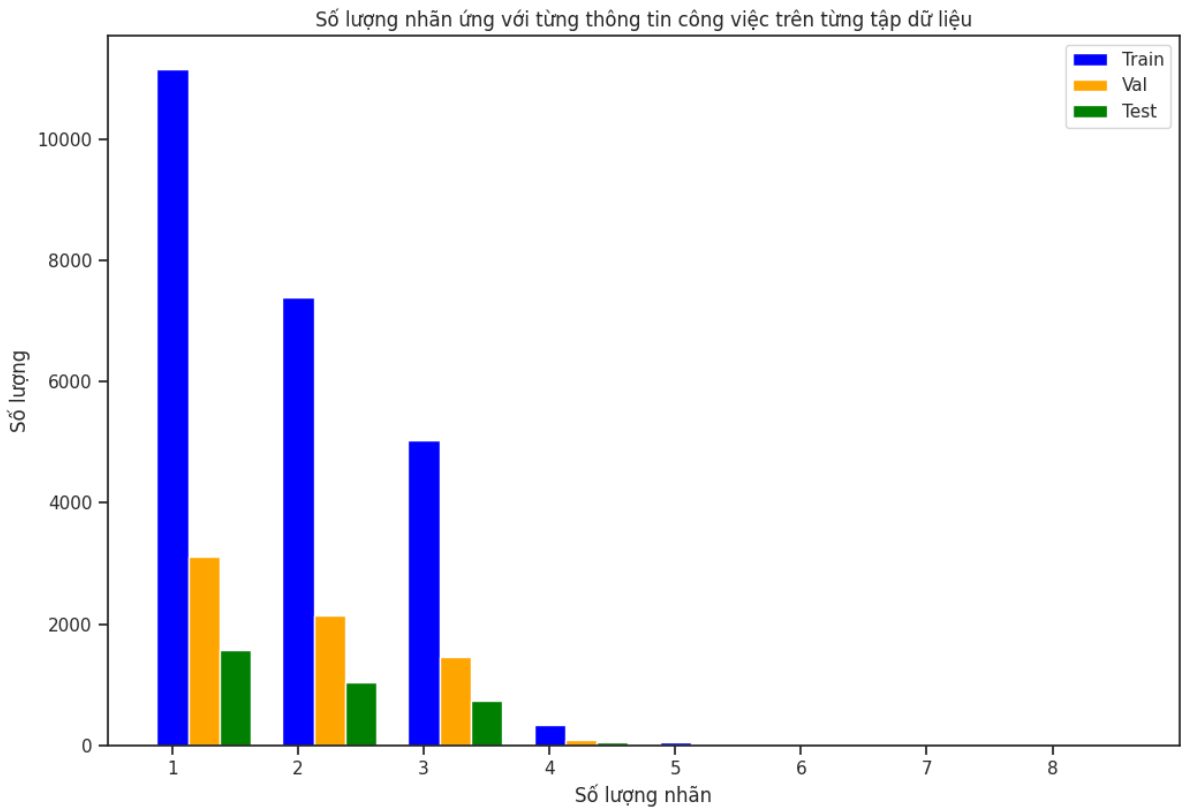
Công việc	**Ngày bắt đầu**	**Ngày hoàn thành**	**Người thực hiện**
Thu thập dữ liệu	14/03/2023	02/04/2023	Tài, Cường
Tiền xử lý dữ liệu	03/04/2023	13/04/2023	Quốc, Tài
Phân tích dữ liệu	03/04/2023	19/04/2023	Quốc, Tài, Cường
Xây dựng mô hình	20/04/2023	15/06/2023	Quốc
Triển khai demo	15/06/2023	17/06/2023	Cường
Soạn nội dung, viết báo cáo và làm slide	15/06/2023	17/06/2023	Quốc, Tài, Cường



Hình 2. Tần suất xuất hiện của từng ngành nghề, lĩnh vực trong tập dữ liệu



Hình 3. Phân phối độ dài mô tả công việc trên từng tập dữ liệu



Hình 4. Barplot số lượng nhãn đối với từng mô tả công việc trên các tập training, validation và test