BÁO CÁO CUỐI KỲ HỌC MÁY

DỰ ĐOÁN THỂ LOẠI PHIM

Họ và tên: Nguyễn Thiên Hảo
Mã sinh viên: *21020906*
Lớp: K66R
Email: 21020906@vnu.edu.vn

Họ và tên: Vũ Trung Hiếu
Mã sinh viên: 22022215
Lớp: AI1
Email: 22022515@vnu.edu.vn

Họ và tên: Bùi Đình Đăng
Mã sinh viên: 21020899
Lớp: K66R
Email: 21020899@vnu.edu.vn

Họ và tên: Trần Đàm Mạnh Cường
Mã sinh viên: *21020891*
Lớp: *K66R*
Email: 21020891@vnu.edu.vn

Họ và tên: Nguyễn Huy Hoàng
Mã sinh viên: *21020912*
Lớp: *K66R*
Email: 21020912@vnu.edu.vn

Tóm tắt

Tổng quan:

Bài toán dự đoán thể loại phim được xử lý dựa trên các bài toán nhỏ: Xử lý ngôn ngữ tự nhiên (NLP), Xử lý hình ảnh (Thị giác máy), Các thuật toán Học máy,….

Các mô hình, thuật toán con được huấn luyện riêng với từng thể loại dữ liệu. Sau cùng các mô hình được tổng hợp để tạo ra mô hình cuối cùng có thể dự đoán thể loại phim dựa trên tổ hợp đầu vào: tiêu đề, ảnh, đánh giá người dùng.

Mục tiêu:

Phát triển mô hình dự đoán thể loại phim bằng cách sử dụng văn bản, poster phim và đánh giá người dùng đầu vào, đạt được độ chính xác cao trong việc dự đoán thể loại phim dựa trên tiêu đề của nó.

Phương pháp:

Phương pháp này bao gồm việc xem xét tài liệu toàn diện để xác định các thuật toán phù hợp để huấn luyện mô hình. Các phương pháp thực nghiệm được sử dụng để đánh giá tính chính xác của mô hình trong việc dự đoán thể loại phim.

Quá trình này bao gồm việc thu thập tập dữ liệu, xử lý trước dữ liệu, huấn luyện mô hình bằng cách sử dụng các thuật toán phân loại khác nhau.

Kết quả:

Trong các thuật toán, mô hình phân loại đã xác định, các mô hình đạt hiệu suất nổi bật, được sử dụng để xây dựng mô hình cuối cùng:

+ Xử lý tiêu đề: mô hình BERT và Mô hình tổng hợp cho các thuật toán Học máy, mạng Neural

+ Xử lý ảnh: Mô hình Dense Net 121

+ Xử lý đánh giá người dùng: Mô hình dựa trên ý tưởng của thuật toán KNN và Recommender System.

Mô hình cuối cùng đạt độ chính xác (chỉ số Map@k) 57.5%

A. MÔ HÌNH XỬ LÝ NGÔN NGỮ

I. Tiền xử lý

1. Tiêu đề

Nhận thấy các tiêu đề đều có cấu trúc chung:

- Cấu trúc: Tên theo ngôn ngữ chính, Mạo từ phổ biến (Tên địa phương) (Năm công chiếu)
- Ví dụ: Eighth Day, The (Le HuitiÃ¨me jour) (1996)
- Chú ý: tiêu đề các bộ phim có thể bị thiếu sót 1 số phần, ví dụ như: Face in the Crowd, A (1957)

Phương pháp xử lý: Thực hiện chuẩn hóa

- Loại bỏ các từ, chữ cái xuất hiện sau dấu ‘(’
- Thực hiện đảo vị trí mạo từ lên đầu tiêu đề

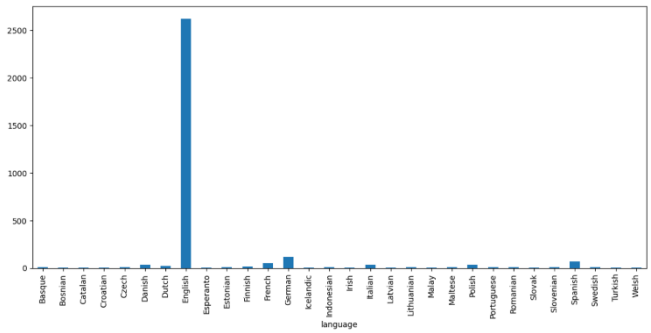
  Ví dụ Eighth Day, The ⇒ The Eighth Day

Kết quả:

- Original Title: Eighth Day, The (Le HuitiÃ¨me jour) (1996)
- Modified Title: The Eighth Day

2. Xử lý ngôn ngữ

Sau khi thực hiện chuẩn hóa, thực hiện xác định ngôn ngữ của các tiêu đề, sử dụng thư viện langid [4]. Kết quả ngôn ngữ thu được:



Nhận thấy rằng, ngoài Tiếng anh chiếm khoảng hơn 2500/3106 mẫu dữ liệu, xuất hiện các ngôn ngữ khác như French, German, Malay, …

Thực hiện kiểm tra chi tiết hơn với các ngôn ngữ chiếm tỷ lệ nhỏ, ví dụ như: Latvian

	movieid	title	genre	language_code	language
657	3574	Carnosaur 3: Primal Species	[Horror, Sci-Fi]	lv	Latvian
824	2050	Herbie Goes Bananas	[Adventure, Children's, Comedy]	lv	Latvian
1182	1078	Bananas	[Comedy, War]	lv	Latvian
2324	756	Carmen Miranda: Bananas Is My Business	[Documentary]	lv	Latvian

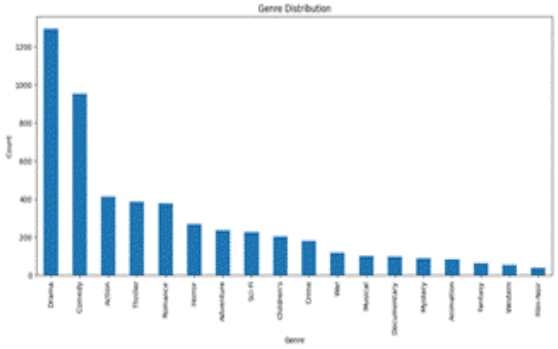
Nhận xét: các ngôn ngữ tỷ lệ nhỏ đều bị chuẩn đoán sai.

Nghĩa là, trên thực tế, toàn bộ tiêu đề phim sau khi được chuẩn hóa ở bước 1 đều là Tiếng anh.

3. Xử lý nhãn (Thể loại)

Chuyển mảng thể loại phim mảng số tự nhiên 18 phần tử chỉ chứa 0 và 1

Ví dụ [Sci-Fi, Thriller]⇒[0.0, 1.0, 0.0, 0.0, 1.0, 0.0, 0.0, 0.0, 0.0]



Thực hiện khảo sát tỷ lệ các loại nhãn phim, thu được biểu đồ.

Nhận xét:

Trong phân bố nhãn phim, các thể loại Drama, Comedy chiếm tỷ lệ nhiều nhất với khoảng 1300 và 1000 lần xuất hiện. Ngược lại với các thể loại phim tỷ lệ nhỏ như Western và Film-Noir, xuất hiện không quá 100 lần.

⇒ Đây là 1 tập dữ liệu mất cân bằng (imbalanced data).

Việc dữ liệu mất cân bằng có thể dẫn tới hậu quả như:

- Trong quá trình đào tạo, các tập dữ liệu nhỏ có sự mất cân bằng có thể khiến mô hình khó học từ các lớp thiểu số, dẫn đến hiệu suất tổng thể kém.
- Khi đánh giá các mô hình dựa trên dữ liệu không cân bằng, việc chỉ dựa vào độ chính xác có thể gây ấn tượng sai lệch về hiệu suất tốt, vì mô hình có thể vượt trội so với nhóm đa số nhưng hoạt động kém về tổng thể. [5]

4. Cân bằng dữ liệu

Ở bước này, chỉ dữ liệu huấn luyện được cân bằng, dữ liệu kiểm thử được giữ nguyên để đảm bảo phân bố trong môi trường thực tế.

Phương pháp cân bằng: Oversampling với các lớp thiểu số

Tương tự như tăng thêm ảnh (Augmentation) trong xử lý ảnh, ta có thể sinh thêm tiêu đề mới dựa trên tiêu đề đã biết.

Một số biện pháp cơ bản:

- Tráo đổi vị trí từ
- Xóa từ trong tiêu đề
- Thay bằng từ đồng nghĩa
- ….[6]

Các cách xử lý trên có thể được thực hiện với thư viện: Nlpaug

Các bước xử lý:

- Xác định số lần sinh thêm tiêu đề với mỗi loại nhãn riêng lẻ.

$$augument_{times} = \frac{số\ lần\ xuất\ hiện\ của\ nhãn\ phổ\ biến\ nhất}{số\ lần\ xuất\ hiện\ của\ nhãn\ hiện\ tại} - 1$$

Ví dụ: Nhãn Drama xuất hiện nhiều nhất với 1200 lần

Nhãn Children's xuất hiện 100 lần

⇒ Số lần sinh thêm của Children's là 1200/ 100 - 1 = 11

- Với mỗi phim, số lần sinh thêm tiêu đề = trung bình số lần sinh thêm tiêu đề của các nhãn con. Nếu thể loại phim có Drama hoặc Comedy thì chỉ định số lần sinh tiêu đề là 0.

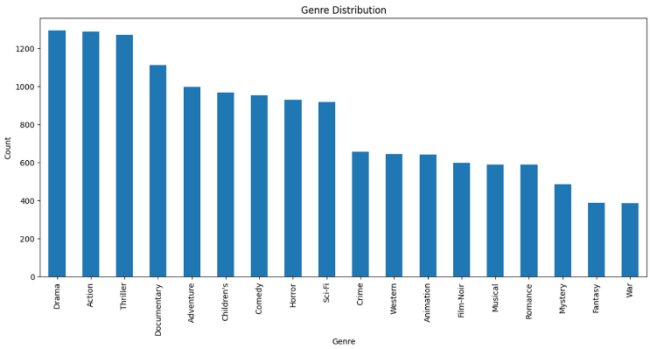
Ví dụ phim A có nhãn [Children's (sinh thêm 11 lần), Animation (sinh thêm 5 lần)]

⇒ Số lần sinh thêm tiêu đề của phim A: (11 + 5) / 2= 8

- Thực hiện chọn số ngẫu nhiên để xác định cách sinh tiêu đề cho mỗi lần
- Kết quả:

Số mẫu data của tập train tăng từ 3106 lên 8781

Tỷ lệ các nhãn



II. Chọn mô hình

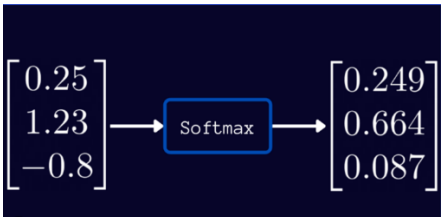
- Xây dựng 1 mô hình RNN và thực hiện huấn luyện từ đầu: LSTM
- Sử dụng mô hình có sẵn đã được huấn luyện: BERT
- Tổ hợp các thuật toán Học máy và mạng Neural cơ bản

III Hàm mất mát

Đối với các bài toán phân loại, có thể sử dụng các các phương pháp, các lớp như Logistic Regression, Softmax Regression

1. Catogorical Cross Entropy (Softmax Regression)

Hàm mất mát này hướng tới bài toán phân loại đơn nhãn:



Vecto đầu ra có đặc trưng:

- Các giá trị đầu ra thể hiện xác xuất dữ liệu rơi vào nhóm xác định
- Tổng các phần tử bằng 1

Phân tích:

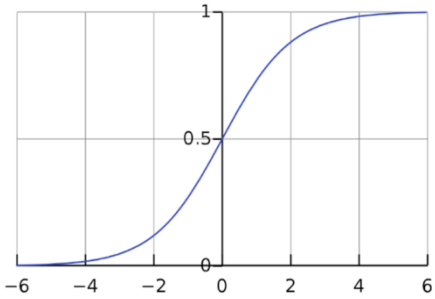
- Có thể sử dụng theo phương pháp sau: đặt 1 ngưỡng (threshold), nếu đầu ra của nhãn lớn hơn threshold thì khẳng định phim thuộc thể loại đó
- Nhược điểm:

  Phải thực hiện thử với nhiều threshold để tìm ra giá trị lý tưởng mà tại đó mô hình có độ chính xác cao nhất

  Trong quá trình huấn luyện, với threshold không chuẩn, biểu đồ về độ chính xác của mô hình không thể hiện đúng như kì vọng. Do đó, khó đánh giá hơn về tính chất của mô hình trong quá trình huấn luyện (overfit, underfit, …)

2. Binary Cross Entropy (Logistic Regression)

Hàm mất mát này hướng tới bài toán phân loại nhị phân (Binary Classification):



Vecto đầu ra có đặc trưng: Chứa một số thể hiện xác suất dữ liệu rơi vào lớp "Có"

Phân tích:

- Có thể sử dụng hàm mất mát Sigmoid với vecto nhiều phần tử. Đầu ra tại mỗi vị trí thể hiện xác xuất dữ liệu rơi vào lớp tương ứng

3. BCE with logtits

Hàm mất mát này là sự kết hợp của hàm kích hoạt Sigmoid và BCELoss (Binary Cross Entropy) trong một lớp duy nhất. Phiên bản này ổn định số học (đầu ra chính xác, không bị tràn số tính toán, …) hơn so với việc sử dụng Sigmoid đơn giản, theo sau là BCELoss. Vì bằng cách kết hợp thành một lớp, thủ thuật log-sum-exp được tận dụng để ổn định số học. [9]

IV. Huấn luyện

1. Học chuyển giao

Transfer learning (TL) là một kỹ thuật trong máy học (ML) mà kiến thức học từ một tác vụ được tái sử dụng để cải thiện hiệu suất trên một nhiệm vụ có liên quan. [10]

Dưới đây là mô tả cho các bước thực hiện học chuyển giao với một mô hình đã được đào tạo trước:

a. Chọn mô hình cơ sở

Lựa chọn một mô hình đã được đào tạo trước có sẵn, thường được huấn luyện trên một tập dữ liệu lớn và đa dạng để học được các đặc trưng tổng quát.

b. Tải mô hình, đóng băng vào loại bỏ

Tải trọng số của mô hình đã được đào tạo.

Đóng băng các lớp của mô hình cơ sở

Các lớp ban đầu của mô hình đã được đào tạo thường học được các đặc trưng tổng quát và chung cho nhiều tác vụ. Việc đóng băng các lớp này giúp bảo toàn tri thức mà mô hình đã học được.

Loại bỏ lớp Fully Connected (FC) cuối cùng của mô hình, vì nó thường liên quan chặt chẽ với tập dữ liệu ban đầu.

Đối với mô hình BERT đang được sử dụng, không cần loại bỏ lớp cuối cùng do đây là mô hình cơ sở với mục đích tái sử dụng, không hướng tới nhiệm vụ cụ thể.

c. Thiết lập kiến trúc mô hình mới

Thêm một số lớp mới ở cuối để phù hợp với nhiệm vụ.

Việc huấn luyện các lớp mới trên nền tri thức được bảo toàn của các mô hình cơ sở, giúp các lớp mới nhanh hội tụ, trở nên phù hợp với nhiệm vụ.

d. Kết quả

Mô hình mới có thể đạt được kết quả mong muốn sau một vài epoch đối với những bài toán, nhiệm vụ nhỏ. Tiết kiệm tài nguyên hơn rất nhiều so với việc huấn luyện từ đầu.

2. Tinh chỉnh mô hình

Fine-tuning đề cập đến việc sử dụng trọng số của một mạng đã được huấn luyện trước đó như là giá trị khởi đầu để huấn luyện một mạng mới.

Dưới đây là mô tả cho các bước thực hiện chuyển giao học với một mô hình đã được đào tạo trước:

a. Chọn Mô Hình Cơ Sở (Base Model):

Lựa chọn một mô hình đã được huấn luyện trước đó trên một tập dữ liệu lớn và có khả năng tổng quát cao. Đây sẽ là mô hình cơ sở để bắt đầu quá trình Fine-tuning.

b. Thực Hiện Fine-tuning (Perform Fine-tuning):

Huấn luyện lại toàn bộ mô hình trên tập dữ liệu mới.

c. Kết quả:

Từ tri thức tổng quát ban đầu, tri thức mô hình được thu gọn lại và học được sâu hơn các khía cạnh về trong lĩnh vực nhiệm vụ.

3. Huấn luyện mô hình BERT

Tải mô về với trọng số đã được huấn luyện

Thực hiện Transfer learning với Learning rate (hệ số học) = 10^{-3}

Đây là Learning rate phổ biến thường được dùng với việc huấn luyện 1 mô hình từ ban đầu. Tương đương với việc xây dựng các lớp mới được thêm vào mô hình từ khi chưa có chi thức gì, đến khi phù hợp với nhiệm vụ bài toán.

Thực hiện Fine Tuning với Learning rate = 10^{-5}

Các tri thức học được ở mô hình cơ sở đã rất ổn định và bao quát, việc đặt Learning rate cao sẽ khiến các tri thức, trọng số cũ bị phá vỡ, mô hình không hội tụ.

Mô hình sau khi đã tương đối ổn định bằng Transfer Learning, Learning rate nên được thu nhỏ lại để dần dần thích hợp hơn với dữ liệu riêng của bài toán khi thực hiện Fine Tuning

Do Learning rate nhỏ, nên có thể huấn luyện với nhiều Epoch (số lần duyệt qua toàn bộ dữ liệu) để mô hình đạt tới kì vọng.

4. Huấn luyện mô hình tổ hợp

STT	Model	Predata
1	Logistic Regression	Oversampling(BorderlineSMOTE)
2	Logistic Regression	Over+Under(SMOTEENN)
3	Gradient Boosting	SMOTEENN
4	Logistic (Classifier Chains)	None
5	SVM	None
6	Neural Network	None

- Đối với các hướng xây dựng mô hình bằng thuật toán Logistic Regression và Gradient Boosting:

Dữ liệu được chia nhỏ thành các nhãn riêng lẻ và thực hiện huấn luyện mô hình trên từng nhãn đơn.

Nghĩa là, đối với từng hướng tiếp cận (1, 2, 3, 4 liệt kê ở bảng), thay vì huấn luyện một mô hình để dự đoán đầu ra của phim (một vecto 18 phần tử), nhóm đã huấn luyện 18 mô hình nhỏ cho riêng từng loại nhãn, đầu ra là 1 số (xác suất phim rơi vào nhãn đó). Sau cùng, thực hiện nối đầu ra của 18 mô hình để đạt đầu ra hoàn chỉnh.

Do mỗi loại phim có đặc trưng riêng, phương pháp này được áp dụng với kì vọng mỗi mô hình có thể hiểu sâu được về thể loại, loại nhãn yêu cầu. Từ đó có thể cải thiện độ chính xác cho mô hình.

- SVM, Neural Network

Mô hình được huấn luyện cho chung 18 nhãn

Sau khi huấn luyện các mô hình con, đầu ra cuối cùng của Mô hình Tổng hợp thu được bằng cách lấy trung bình cộng của đầu ra của các mô hình con.

V. Đánh giá

Mô hình sau khi huấn luyện được đánh giá thông qua 3 thông số: Map@K, F1 Micro, F1 Macro

1. Map@K

MAP@K là một chỉ số đo lường được sử dụng để đánh giá hiệu suất của hệ thống đề xuất (Reccomender System). Nó đo lường trung bình chính xác đề xuất của K đề xuất hàng đầu cho các người dùng khác nhau và tính trung bình qua tất cả các truy vấn trong tập dữ liệu.

Ví dụ, nếu chúng ta có một tập dữ liệu gồm 100 người dùng và muốn đánh giá hiệu suất của một hệ thống đề xuất gợi ý 10 mục cho mỗi người dùng, chúng ta có thể tính toán MAP@10 bằng cách tính trung bình chính xác cho mỗi người dùng và sau đó lấy trung bình của những giá trị này. [11]

2. F1 Micro

Chỉ số F1 Micro là công thức cho chỉ số F1 thông thường nhưng được tính bằng cách sử dụng tổng số True Positives (TP), False Positives (FP) và False Negatives (FN), thay vì tính riêng lẻ cho từng lớp. [13]

$$\text{Micro F1 score} = \frac{TP}{TP + \frac{1}{2} \bullet (FP + FN)}$$

3. F1 Macro

Chỉ số F1 Macro là trung bình không trọng số của các chỉ số F1 được tính toán riêng cho mỗi loại nhãn. [13]

$$\text{Macro F1 score} = \frac{\text{sum (F1 scores)}}{\text{number of classes}}$$

Đối với dữ liệu mất cân bằng, chỉ số F1 Macro thường thể hiện tốt hơn so với F1 Micro. Nguyên nhân là vì F1 Micro coi mỗi mẫu dữ liệu có sự quan trọng bằng nhau, trong khi F1 macro coi mỗi lớp có sự quan trọng bằng nhau.

VI. Thí nghiệm

Code mẫu của mô hình BERT [14][15]

Kết quả huấn luyện các mô hình:

STT	Mô hình	Epoch (Transfer + Fine Tune)	Data	Map@K (k = 5)	F1 Micro	F1 Macro
1	LSTM	26	Augment	0.33	0.12	0.08
2	BERT Model 1	5 + 10	Non-Augment	0.25	0.09	0.07
3	BERT Model 1	5 + 10	Augment	0.30	0.12	0.09
4	BERT Model 2	5 + 10	Augment	0.21	0.08	0.04
5	BERT Model 2	15 + 30	Augment	0.41	0.31	0.18
6	BERT Model 1	15 + 30	Augment	0.39	0.30	0.19

Kết quả của các mô hình con trong Mô hình tổng hợp:

STT	Model	Data	Map@K (k = 5)
1	Logistic Regression	BorderlineSMOTE	0.54
2	Logistic Regression	SMOTEENN	0.50
3	Gradient Boosting	SMOTEENN	0.45
4	Logistic (Classifier Chains)	None	0.52
5	SVM	None	0.55
6	Neural Network	None	0.54

Chú thích

Data:

- Augment: Sử dụng nguồn dữ liệu huấn luyện đã được sinh thêm (gần 9000 mẫu)
- Non-Augment: Dữ liệu huấn luyện chỉ được xử lý bỏ đi các phần không cần thiết (hơn 3000 mẫu)
- BorderlineSMOTE, SMOTEENN: Các phương pháp cân bằng dữ liệu dựa vào kỹ thuật Resampling

VII. Phân tích

1. Nên sử dụng dữ liệu nào

Dựa vào bảng kết quả huấn luyện, theo thí nghiệm số 2 và 3.

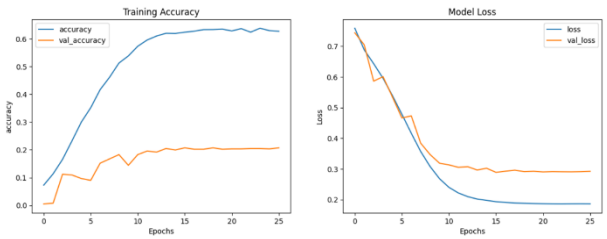
Nhận thấy, việc sử dụng dữ liệu đã được sinh thêm có tác dụng khiến độ chính xác của mô hình tăng đáng kể. Dữ liệu cân bằng có tác dụng tốt đối với mô hình.

Dựa vào bảng kết quả các mô hình con của Mô hình Tổng hợp, đóng góp của Resampling dữ liệu đối với biểu hiện của mô hình không rõ ràng.

Tuy nhiên, đối với các phương pháp Resampling: dữ liệu được xử lý theo chiều hướng có UnderSampling (SMOTEENN), nhìn chung gây ảnh hưởng không tốt tới mô hình.

2. Nên sử dụng mô hình nào

a. LSTM model



Nhận thấy mô hình đã hội tụ sau khoảng 25 epochs, trong khi các thông số thể hiện độ chính xác rất thấp.

Map@K (k = 5)	F1 Micro	F1 Macro
0.33	0.12	0.08

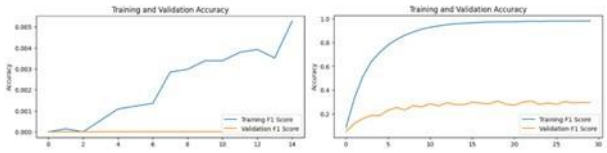
Điều này thể hiện mô hình đang gặp phải Underfit. Nguyên nhân có thể là do:

- Dữ liệu ít, xuất hiện nhiều tên thực thể, đối tượng mới trong tập kiểm thử mà chưa được gặp ở trong từ điểm của tập huấn luyện
- Công cụ Tokenize phổ thông khi gặp từ mới trong từ điển đều đưa về giá trị <UNK>. Các từ như vậy thông qua quá trình học, không thu được trọng số riêng. Do vậy, mô hình khó để nhận diện được thể loại phim khi gặp các từ vựng mới.

b. BERT Model 1 và BERT Model 2

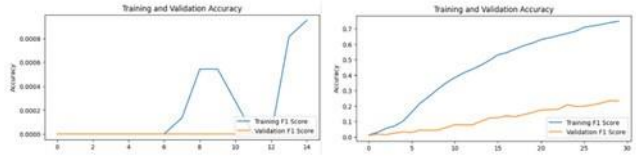
Hai mô hình trên sử dụng công cụ Tokenize và lớp Embedding có sẵn của mô hình BERT. So sánh với mô hình LSTM, hai mô hình đã đề cập có thể nhận diện được cả các từ vựng mới ngay cả khi chưa gặp trong tập từ điển huấn luyện. Tuy nhiên, tên đối tượng mới vẫn là một thách thức.

Độ chính xác của BERT Model 1(đánh giá theo thông số F1) (kết quả của 2 quá trình liên tiếp Transfer Learning và Fine Tuning):



Mô hình BERT 1 đã hội tụ sau khoảng 30 epochs. Và bị vướng phải vấn đề Overfiting

Độ chính xác của BERT Model 2(đánh giá theo thông số F1) (kết quả của 2 quá trình liên tiếp Transfer Learning và Fine Tuning):



Nhận xét:

- Do mô hình được xây dựng với tính chất phức tạp hơn mô hình BERT 1. Do đó, khi so sánh, nhận thấy rằng tốc độ hội tụ của mô hình 2 chậm hơn.
- Đồng thời, mô hình sau khoảng 40 epochs cũng bị vướng phải vấn đề Overfitting.

c. Lựa chọn

Dựa theo bảng thông số huấn luyện ở mục IV.

Mô hình BERT 2 và Mô hình tổng hợp được lựa chọn là mô hình có độ chính xác cao nhất trong các mô hình đã đề ra, và được sử dụng để kết hợp với các mô hình xử lý ảnh, mô hình xử lý đánh giá để dự đoán cho đầu ra thể loại phim cuối cùng.

B. MÔ HÌNH XỬ LÝ ẢNH

I. Tiền xử lý

1. Lọc dữ liệu

a. Lọc dữ liệu ảnh

Để tránh duyệt phải những ảnh có chất lượng thấp, tập ảnh được đưa vào một chương trình để duyệt, và kết quả là tìm và xóa được 23 ảnh có chất lượng không đảm bảo

```
Issue checks completed. 23 issues found in the dataset
         issue_type  num_images
0  exact_duplicates          18
1   low_information           3
2         grayscale           2
3            blurry           0
4              dark           0
5             light           0
6  odd_aspect_ratio           0
7          odd_size           0
8   near_duplicates           0
```

Hình 1. Số lượng ảnh lỗi quét được

b. Lọc dữ liệu trong các các tệp text

Bởi vì mô hình được xây dựng để huấn luyện dữ liệu ảnh, nên nếu không tồn tại đường dẫn ảnh trong các tệp movies_train và movies_test, quá trình duyệt dữ liệu của mô hình sẽ gặp phải trục trặc và lỗi, vậy nên bước tiền xử lý sẽ được thực hiện trên các tệp này để xóa đi các movieid không tồn tại trong folder ảnh. Và các file sau khi được lọc sẽ được lưu tên mới là movies_train_update, movies_test_update. Dữ liệu sau khi được lọc có sự khác biệt như hình dưới.

	movieid	title	genre
0	1650	Washington Square (1997)	Drama
1	185	Net, The (1995)	Sci-Fi|Thriller
2	1377	Batman Returns (1992)	Action|Adventure|Comedy|Crime
3	3204	Boys from Brazil, The (1978)	Thriller
4	1901	Dear Jesse (1997)	Documentary
...	...	...	...
3101	2539	Analyze This (1999)	Comedy
3102	3038	Face in the Crowd, A (1957)	Drama
3103	1832	Heaven's Burning (1997)	Action|Drama
3104	657	Yankee Zulu (1994)	Comedy|Drama
3105	1750	Star Kid (1997)	Adventure|Children's|Fantasy|Sci-Fi

3106 rows × 3 columns

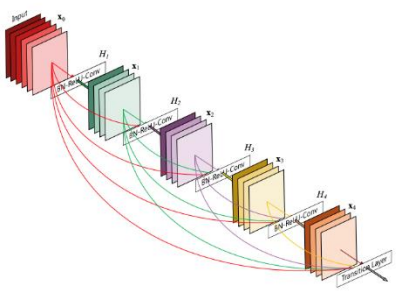
	movieid	title	genre
0	1650	Washington Square (1997)	Drama
1	185	Net, The (1995)	Sci-Fi|Thriller
2	1377	Batman Returns (1992)	Action|Adventure|Comedy|Crime
3	3204	Boys from Brazil, The (1978)	Thriller
7	2382	Police Academy 5: Assignment: Miami Beach (1988)	Comedy
...	...	...	...
3099	2921	High Plains Drifter (1972)	Western
3100	502	Next Karate Kid, The (1994)	Action|Children's
3101	2539	Analyze This (1999)	Comedy
3102	3038	Face in the Crowd, A (1957)	Drama
3105	1750	Star Kid (1997)	Adventure|Children's|Fantasy|Sci-Fi

2593 rows × 3 columns

Hình 2. Tập dữ liệu trước và sau khi lọc

2. Lựa chọn mô hình

Sau khi tìm hiểu và tham khảo các mô hình hiện có, 2 mô hình có kết quả khá tốt cho bài toán này là mô hình DenseNet121 và mô hình CNN tự xây dựng, tuy nhiên trong quá huấn luyện và thực nghiệm, mô hình DenseNet121 đưa ra kết quả đồng đều hơn cho độ chính xác của các hàm đánh giá. Do đó mô hình DenseNet121 được huấn luyện sẵn được sử dụng cho các đánh giá sau này.



Hình 3. Cấu tạo mô hình DenseNet121

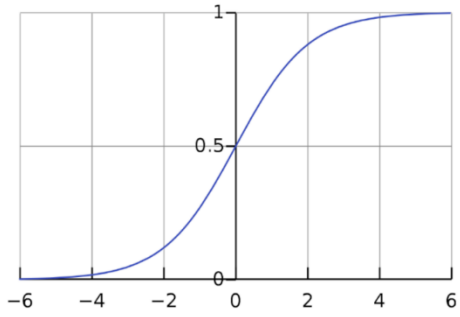
a. Một số điểm đặc trưng của mô hình

Dropout layers

Lớp này được sử dụng để ngẫu nhiên loại bỏ một phần các đơn vị đầu ra, giúp giảm overfitting bằng cách ngăn chặn mô hình phụ thuộc quá nhiều vào một số đặc trưng cụ thể. Trong trường hợp này, 50% số đơn vị sẽ bị loại bỏ ngẫu nhiên trong quá trình huấn luyện.

BinaryCrossEntropy và Sigmoid

Binary Cross Entropy thường được sử dụng kết hợp với hàm kích hoạt sigmoid ở lớp đầu ra để đo lường sự sai lệch giữa dự đoán và nhãn thực tế. Hàm sigmoid chuyển đổi giá trị đầu ra thành một phạm vi từ 0 đến 1, tương ứng với xác suất thuộc về mỗi lớp trong bài toán nhị phân.



Hình 4. Ảnh minh họa hàm sigmoid

b. Một số biện pháp giảm thiểu Overfit

EarlyStopping

Đây là một kỹ thuật để ngừng việc huấn luyện mô hình sớm khi một phần của mô hình bắt đầu overfitting. Bằng cách theo dõi metric được chọn trên tập validation (trong trường hợp của bạn là 'val_loss'), nếu không có sự cải thiện trong khoảng thời gian được xác định (patience) thì quá trình huấn luyện sẽ dừng lại. Điều này giúp tránh overfitting và tiết kiệm thời gian huấn luyện.

ModelCheckpoint

Khi huấn luyện mô hình, đôi khi bạn muốn lưu lại trọng số của mô hình tại các điểm mốc khi có sự cải thiện trên một metric nào đó. ModelCheckpoint cho phép bạn làm điều này. Trong trường hợp của bạn, nó sẽ lưu trọng số mô hình tốt nhất dựa trên metric 'val_f1_m' trên tập validation.

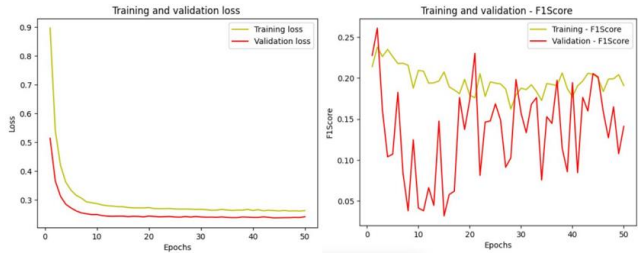
c. Các hàm đánh giá trên mô hình

Mô hình sẽ áp dụng các hàm đánh giá là map@k và f1_score để đánh giá độ chính xác của mô hình.

3. Huấn luyện và đánh giá mô hình

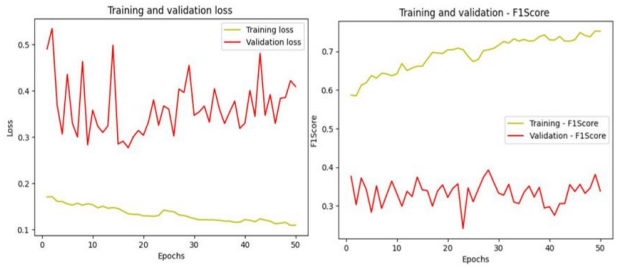
a. Huấn luyện mô hình

Mô hình DenseNet121 này sẽ được huấn luyện dựa trên phương pháp Transfer Learning (TL) và Fine-tuning (Ft) như được đề cập bên trên, và mô hình sẽ lần lượt được huấn luyện theo thứ tự TL → Ft →TL. Kết quả huấn luyện mô hình với giá trị tham chiếu là f1_score và loss là:



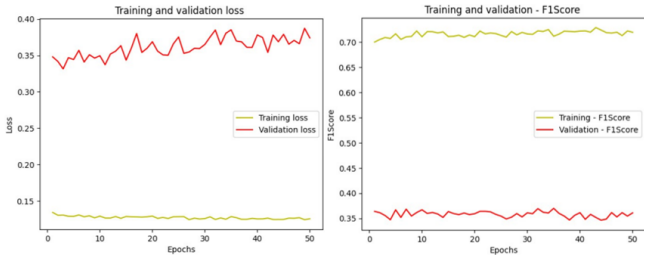
Hình 5. Kết quả huấn luyện lần 1 (TL)

Qua lần huấn luyện đầu tiên, các lớp đầu ra của mô hình được huấn luyện riêng để tăng khả năng tương thích với tập dữ liệu mới hơn. Lúc này độ chính xác của mô hình khá thấp, chỉ dao động quanh 20%.



Hình 6. Kết quả huấn luyện lần 2 (Ft)

Tại lần huấn luyện thứ 2, toàn bộ trọng số của mô hình sẽ được huấn luyện lại để có thể học được sâu hơn các đặc trưng của tập dữ liệu. Lúc này, độ chính xác của mô hình tăng lên đáng kể.



Hình 7. Kết quả huấn luyện lần 3 (TL)

Tại lần huấn luyện cuối cùng, các trọng số trong lớp base sẽ bị đóng băng, và chỉ huấn luyện lại các lớp đầu ra của mô hình. Lúc này mô hình có độ chính xác trên tập train khá ổn định, ở mức 70%.

b. Đánh giá mô hình

Mô hình sẽ được đánh giá qua giá trị của f1score và map@k với các trọng số tốt nhất đã được lưu lại qua mỗi lần huấn luyện dữ liệu.

DenseNet121	Epochs	mAP@k	F1score
	50	0.512	0.357
	40	0.509	0.341
	20	0.531	0.343

C. MÔ HÌNH XỬ LÝ USER VÀ RATINGS

I. Ý TƯỞNG

Mô hình được xây dựng để dự đoán đầu ra dựa trên Users và Ratings được dựa trên ý tưởng của thuật toán KNN và Recommender System.

Mô tả về dữ liệu Users:

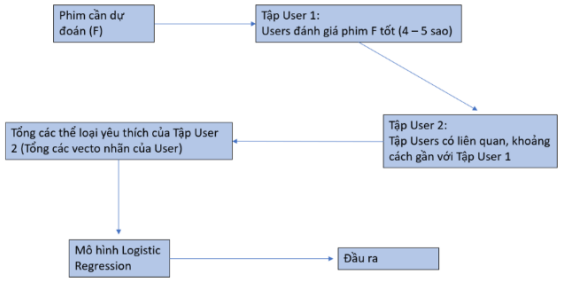
- Người dùng được coi như tọa độ, các thông tin đặc trưng được chuyển thành các vecto số học.
- Tập tổng tích lũy các thể loại mà người dùng đánh giá từ 4 đến 5 sao được coi là nhãn.

Ví dụ: [50, 100, 36,….15]

Số 50 thể hiện người dùng đã đánh giá tốt (4 - 5 sao) cho 50 bộ phim thuộc loại 1

Về cơ bản, nhãn có thể hiểu là đặc trưng sở thích của người dùng

Phương pháp tiếp cận:



- Với phim cần dự đoán thể loại F, tìm tập những người dùng đã đánh giá phim F tốt (Tập User 1)
- Với mỗi người dùng trong Tập User 1, lấy K người liên quan, có khoảng cách gần nhất, thu được Tập User 2
- Thực hiện lấy tổng các thể loại yêu thích của Tập User 2, thu được vecto tương tự nhãn được mô tả ở cho User ở trên.
- Đây được coi tập sở thích chung của cộng đồng gần những người đã đánh tốt cho phim F. Kì vọng rằng, thể loại của phim F gần với sở thích của cộng đồng.
- Đưa dữ liệu vừa thu được qua mô hình Logistic để phần trăm hóa đầu ra cho mỗi loại nhãn, đồng thời có thể học thêm được một số dữ liệu cho từng loại nhãn. Ví dụ, những loại nhãn cho dù ít xuất hiện cũng có thể được xác nhận là thể loại của phim.

II. Experiment

Chỉ số Map@K của mô hình: 0.2

D. MÔ HÌNH TỔNG HỢP CUỐI CÙNG

1. Các lớp, thư viện sử dụng:

Bayesian Optimization là một phương pháp tối ưu hóa được sử dụng để tìm kiếm giá trị tối ưu của một hàm mục tiêu không đồng thời cho trước trong một không gian tìm kiếm. Phương pháp này kết hợp việc sử dụng mô hình thống kê (thường là Gaussian Process) với kiến thức cũng như phân phối xác suất của các điểm tìm kiếm để dự đoán điểm tiếp theo sẽ tìm ra giá trị tối ưu của hàm mục tiêu.

Cụ thể, Bayesian Optimization áp dụng các kiến thức thu được từ các điểm đã thử nghiệm để tập trung tìm kiếm ở những vùng tiềm năng tối ưu hơn, thay vì dựa vào tìm kiếm ngẫu nhiên. Điều này giúp tăng tốc quá trình tối ưu hóa và giảm số lần thử nghiệm cần thiết.

Bayesian Optimization thường được sử dụng trong các bài toán tối ưu siêu tham số cho mô hình học máy, tối ưu các siêu tham số cho bộ phân loại hoặc bộ hồi quy, và các bài toán tối ưu hóa khácnh trong lĩnh vực machine learning và optimization.

2. Phương pháp:

- Đưa dữ liệu tập huấn luyện đi qua 3 mô hình xử lý Rating, Title và Image. Thu được 3 ma trận đầu ra 18 nhãn tương ứng
- Đưa dữ liệu tập huấn luyện đi qua 3 mô hình xử lý Rating, Title và Image. Thu được 3 ma trận đầu ra 18 nhãn tương ứng

Khi huấn huyện mô hình cho từng đầu vào, nhận thấy rằng, một số phim bị thiếu tiêu đề, ảnh và có thể chưa được người dùng nào đánh giá. Vì vậy, đối với các vecto đầu ra, khi dữ liệu bị thiếu (ví dụ như không có tiêu đề) thì đầu ra phần bị thiếu được thay bằng 0.

- Thực hiện kết nối 3 vecto đầu ra , đưa đi qua Bayesian Optimization, với nhãn là tập nhãn thực tế của phim. Tại bước này, dựa vào việc lấy mẫu các trọng số ngẫu nhiên w1, w2, w3 trong khoảng cho trước (0, 1) cho các mô hình, thu được trọng số phù hợp, tối đa độ chính xác cho Mô hình tổng hợp cuối cùng.
- Bài toán:

  Tối ưu cho hàm mục tiêu f(w1,w2,w3):

  Maxf(w1,w2,w3)=Map@k(w1*model1+w2*model2+w3* model3)

  Điều khiện w1+w2+w3≤1

3. Kết quả:

Mô hình Tổng hợp cuối cùng đạt độ chính xác trên tập huấn luyện với chỉ số Map@K = 94.4%

Mô hình Tổng hợp cuối cùng đạt độ chính xác trên tập kiểm thử với chỉ số Map@K = 57.7%

References

[1] https://aws.amazon.com/vi/what-is/nlp/#:~:text=Natural language processing (NLP) is,manipulate%2C and comprehend human language.

[2] https://www.analyticsvidhya.com/blog/2022/03/a-brief-overview-of-recurrent-neural-networks-rnn/#:~:text=Recurrent Neural Networks (RNNs) are,them to capture temporal dependencies.

[3] Transformer Explained | Papers With Code

[4] https://stackoverflow.com/questions/39142778/how-to-determine-the-language-of-a-piece-of-text

[5] What is imbalanced data? Simply explained (stephenallwright.com)

[6] https://neptune.ai/blog/data-augmentation-nlp

[7] https://www.coursera.org/learn/nlp-sequence-models/lecture/6Oq70/word-representation

[8]https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/

[9]https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html

[10] Transfer learning - Wikipedia

[11]machine learning - Understanding Precision@K, AP@K, MAP@K - Stack Overflow

[12] Micro vs Macro F1 score, what's the difference? (stephenallwright.com)

[13] Micro vs Macro F1 score, what's the difference? (stephenallwright.com)

[14] https://huggingface.co/docs/transformers/model_doc/bert

[15]https://colab.research.google.com/github/abhimishra91/transformers-tutorials/blob/master/transformers_multi_label_classification.ipynb

[16]Hyperparameter tuning

https://www.geeksforgeeks.org/hyperparameter-tuning/

https://www.vebuso.com/2020/03/svm-hyperparameter-tuning-using-gridsearchcv/

https://www.youtube.com/watch?v=5nYqK-HaoKY

[17]Bayes

https://github.com/bayesian-optimization/BayesianOptimization

https://www.run.ai/guides/hyperparameter-tuning/bayesian-hyperparameter-optimization

https://github.com/bayesian-optimization/BayesianOptimization/blob/master/examples/constraints.ipynb

[18]Bayesian Optimization: bayes_opt or hyperopt

https://www.analyticsvidhya.com/blog/2021/05/bayesian-optimization-bayes_opt-or-hyperopt/

[19]Using Gradient Boosting

https://www.youtube.com/watch?v=StWY5QWMXCw