

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



BÁO CÁO DỰ ÁN
NHẬP MÔN KHOA HỌC DỮ LIỆU

**Đề tài: “Phân tích sự phát triển của doanh nghiệp theo thời gian
và dự báo xu hướng trong tương lai”**

DANH SÁCH THÀNH VIÊN NHÓM 3 - LỚP N04

Ngô Văn Trọng	MSSV : B21DCCN726
Cam Hải Đăng	MSSV : B21DCCN027
Bùi Anh Tú	MSSV : B21DCCN743
Nguyễn Bá Hoàng Huynh	MSSV : B21DCCN447
Nguyễn Như Thiệu	MSSV : B21DCCN690

Giảng viên hướng dẫn : ThS. Đinh Xuân Trường

HÀ NỘI - 09/2024

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



BÁO CÁO DỰ ÁN
NHẬP MÔN KHOA HỌC DỮ LIỆU

***Đề tài: “Phân tích sự phát triển của doanh nghiệp theo thời gian
và dự báo xu hướng trong tương lai”***

Nhóm 03

Ngô Văn Trọng

Cam Hải Đăng

Bùi Anh Tú

Nguyễn Bá Hoàng Huynh

Nguyễn Như Thiệu

Lớp N04

MSSV : B21DCCN726

MSSV : B21DCCN027

MSSV : B21DCCN743

MSSV : B21DCCN447

MSSV : B21DCCN690

LỜI CẢM ƠN

Nhóm chúng em xin gửi lời cảm ơn chân thành đến thầy vì đã luôn sát sao, tận huyết trong quá trình giảng dạy, cho chúng em có nhiều cơ hội để tìm hiểu, thể hiện kiến thức và nhận ra được khoảng trống tri thức, ưu và nhược điểm của bản thân. Ngoài ra thầy còn giúp chúng em có cái nhìn sâu sắc, rộng mở hơn về lĩnh vực khoa học dữ liệu nói riêng, khoa học nói chung, gợi mở ra nhiều con đường, định hướng cho bản thân sau này.

Hà Nội, ngày 18 tháng 10 năm 2024

Đại diện nhóm

Ngô Văn Trọng

MỤC LỤC

LỜI CẢM ƠN.....	i
MỤC LỤC.....	ii
DANH MỤC CÁC HÌNH VẼ.....	iii
DANH MỤC CÁC BẢNG	v
PHẦN MỞ ĐẦU	vi
CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI.....	1
1.1 Tổng quan đề tài	1
1.2 Định hướng giải pháp và hạn chế.	2
CHƯƠNG 2. BÁO CÁO TIẾN ĐỘ TỪNG TUẦN	3
2.1 Tổng quan tiến độ công việc, phân công theo từng tuần.	3
2.2 Tuần 4: Thu thập và chuẩn bị dữ liệu (06/09 - 12/09)	3
2.3 Tuần 5: Tiền xử lý dữ liệu (13/09 - 19/09)	10
2.4 Tuần 6: Chuẩn hóa lại và thử nghiệm trên 1 số mô hình (20/09 - 26/09)	15
2.5 Tuần 7: Thử nghiệm trên một số mô hình (26/09 - 04/10).....	24
2.6 Tuần 8: Thử nghiệm mô hình và thực hiện nội suy trọng số(05/10 - 11/10).....	26
2.7 Tuần 9: Thử nghiệm mô hình Neural Network trên Orange Data Mining và chuyển đổi mô hình dữ liệu thành đồ thị(10/10 - 17/10).....	31

DANH MỤC HÌNH VẼ

1.1	Doanh số phát triển của các mặt hàng trên TikTok	1
1.2	Doanh nghiệp eHerb Viet Nam.	2
2.1	Tiến độ thực hiện.	3
2.2	Mở DevTool thì thấy website chỉ cung cấp phương thức POST.	4
2.3	DevTool của website không hỗ trợ phương thức GET với những dữ liệu cần lấy.	4
2.4	Tắt JavaScript của trang web.	5
2.5	Di chuột để hiển thị dữ liệu đồ thị.	5
2.6	Khai báo thư viện.	6
2.7	Giả lập đăng nhập.	6
2.8	Một số thao tác để đến với trang doanh số 30 ngày.	7
2.9	Script lấy tọa độ trên đồ thị.	8
2.10	Kết quả ví dụ với đồ thị về doanh thu.	8
2.11	Code tổng quát để lấy dữ liệu đồ thị và lưu vào file CSV.	9
2.12	Kết quả về dữ liệu các loại doanh số thu thập được trên máy(CSV).	9
2.13	Dữ liệu đã thu thập được	10
2.14	Import 2 thư viện cần dùng	10
2.15	Đặt tên, chuẩn hoá ngày tháng, bỏ kí hiệu tiền tệ	11
2.16	Trích xuất phần số trong 1 chuỗi.	11
2.17	Loại bỏ ký tự thừa.	12
2.18	Lưu vào file CSV.	12
2.19	Kết quả cuối cùng khi lưu vào file CSV.	13
2.20	Dữ liệu về các nhà sáng tạo theo sản phẩm.	14
2.21	Bảng quan hệ dữ liệu.	14
2.22	Kiểm tra dữ liệu thiếu.	15
2.23	Khai báo thư viện	15
2.24	Dữ liệu doanh thu	16
2.25	Dữ liệu về các nhà sáng tạo từ một sản phẩm	16
2.26	Bình thường hóa dữ liệu bằng MinMaxScaler	16
2.27	Đổi tên các cột và xuất file csv	17
2.28	Dữ liệu doanh thu sau khi bình thường hóa	17
2.29	Dữ liệu về các nhà sáng tạo từ một sản phẩm sau khi bình thường hóa	17
2.30	Chuẩn hóa dữ liệu bằng StandardScaler	18
2.31	Dữ liệu doanh thu	18
2.32	Dữ liệu về các nhà sáng tạo từ một sản phẩm	18
2.33	Dữ liệu doanh thu sau khi chuẩn hóa	19
2.34	Dữ liệu về các nhà sáng tạo từ một sản phẩm sau khi chuẩn hóa	19
2.35	Chi bộ dữ liệu	20

2.36	Chỉ bước thời gian.	20
2.37	Tạo mô hình	20
2.38	Huấn luyện mô hình và dự đoán	21
2.39	Huấn luyện mô hình và dự đoán	22
2.40	Dự đoán 3 ngày	22
2.41	Biểu đồ	23
2.42	Xây dựng mô hình	25
2.43	Dự đoán	25
2.44	Dữ liệu doanh thu	27
2.45	Sơ đồ thực hiện	28
2.46	Kết quả thu được	29
2.47	Bảng dữ liệu nhà sáng tạo	29
2.48	Code tìm trọng số	30
2.49	Sơ đồ thực hiện các mô hình Neural Network	32
2.50	Mô hình tổng quan về đồ thị	37
2.51	Đồ thị theo TimeSeries.	37

DANH MỤC BẢNG BIỂU

2.1	Forecasted Values from Prophet Model	25
2.2	So sánh các mô hình theo số lượng neuron cho mỗi hidden layer	33
2.3	So sánh các mô hình theo hàm kích hoạt	33
2.4	So sánh các mô hình theo thuật toán tối ưu	34
2.5	So sánh các mô hình theo Regularization	34
2.6	So sánh các mô hình theo số fold trong Cross validation	34
2.7	So sánh các mô hình theo việc lựa chọn các thuộc tính làm features	35

PHẦN MỞ ĐẦU

Mục tiêu và định hướng về môn khoa học dữ liệu: Mục tiêu bản thân trong quá trình học là xây dựng thức nền tảng thật tốt cho môn Khoa học dữ liệu, từ những kiến thức toán học, thống kê đến những kiến thức về thu thập, cấu trúc, tiền xử lý dữ liệu và cuối cùng là xây dựng mô hình học máy/học sâu, tạo tiền đề rộng mở cho nhiều định hướng sau này như Data Engineer/Analyst/Science, AL/ML Enggineer,.....

Vấn đề liên quan đến đề tài khoa học dữ liệu: Hiện nay, với sự phát triển nhanh chóng của các sàn thương mại điện tử như Shopee, Lazada mà đặc biệt là TikTokShop thì việc quảng cáo, mua bán online đang dần trở nên phổ biến bởi sự tiện lợi. Từ đó lưu lượng người mua cộng với doanh số của các sàn thương mại điện tử này ngày càng khổng lồ. Vậy nên việc phân tích dữ liệu của doanh nghiệp trên sàn để đưa ra phương hướng, chiến lược Marketing, phát triển đúng đắn cho doanh nghiệp là điều thực sự cần thiết

Các giải pháp hiện tại và hạn chế: Trong thực tế, việc thu thập, phân tích dữ liệu rồi xây dựng mô hình sẽ gặp nhiều khó khăn bởi dễ dẫn đến việc thiên kiến về mặt dữ liệu, những vấn đề khác trong dữ liệu, dẫn đến mô hình dự đoán không chính xác, phân tích chưa tốt dẫn đến chiến lược Marketing sai lầm. Ngoài ra còn nhiều yếu tố ảnh hưởng bên ngoài mà khả năng mô hình AI có thể len lỏi vào được là rất khó, cho nên cũng cần am hiểu rất nhiều về những yếu tố xã hội, sau đó mới kết hợp với AI thì mới có thể đưa ra được những sự phân tích, những insight đúng đắn nhất về dữ liệu cho doanh nghiệp.

Hơn nữa dữ liệu về doanh số của doanh nghiệp thường là bảo mật, vậy nên ở khía cạnh bài tập lớn, nhóm lại càng khó đưa ra được những insight đúng đắn nhất cho thực tế mà chỉ nằm ở mức lý thuyết.

Mục tiêu và hướng giải pháp: Phân loại, phân cụm được các sản phẩm, các nhà sáng tạo, các video, livestream có nội dung ở các mức độ tiềm năng để đưa ra các gợi ý về chiến lược Marketing cho doanh nghiệp. Đưa ra dự đoán về doanh số trong tương lai gần của doanh nghiệp.

Đóng góp của đề tài và bố cục của bài báo cáo: Hướng tới nhu cầu đó, đề tài có tên “*Phân tích sự phát triển của doanh nghiệp theo thời gian và dự báo xu hướng trong tương lai*”.

Nội dung trình bày trong báo cáo gồm 3 chương chính:

- Chương 1: Giới thiệu chung về đề tài.
- Chương 2: Báo cáo tiến độ từng tuần
 - Giới thiệu chung Trình bày tổng quan về đề tài, xác định được mục tiêu, đối tượng, phương hướng giải quyết và giới thiệu những kiến thức công nghệ liên quan.
 - Cơ sở lý thuyết.
 - Xây dựng chương trình.
- Chương 3: Kết luận về đề tài nghiên cứu bao gồm Bài học và kết quả đạt được từ đó rút ra những điều cần cải thiện trong tương lai.

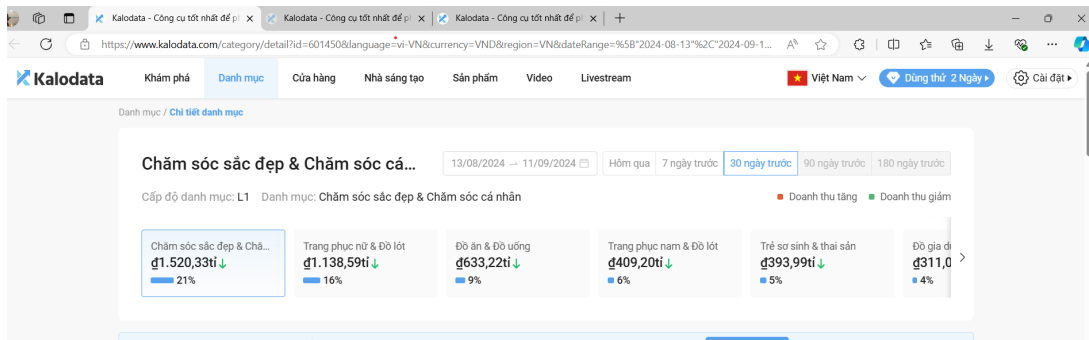
CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

Tóm tắt: Chương 1 giới thiệu bài toán Phân tích sự tăng trưởng của một doanh nghiệp trên sàn thương mại điện tử theo thời gian. Trong đó, nhóm sẽ phân tích các yếu tố tác động đến sự tăng trưởng ấy, phân loại hoặc phân cụm cho các yếu tố ấy để đưa ra định hướng phát triển cho doanh nghiệp. Ngoài ra nhóm còn đưa ra mô hình dự đoán doanh thu theo thời gian cho doanh nghiệp.

1.1 Tổng quan đề tài

Khảo sát hiện trạng

Hiện nay, với sự phát triển nhanh chóng của nền tảng TikTok thì việc quảng cáo, mua bán hàng qua các video ngắn, LiveStreams TikTok đang dần trở nên phổ biến bởi sự tiện lợi. Từ đó lưu lượng người mua cộng với doanh số của sàn thương mại điện tử TikTokShop ngày càng khổng lồ. Vậy nên việc phân tích dữ liệu của doanh nghiệp trên sàn để đưa ra phương hướng, chiến lược Marketing, phát triển đúng đắn cho doanh nghiệp là điều thực sự cần thiết.



Hình 1.1: Doanh số phát triển của các mặt hàng trên TikTok

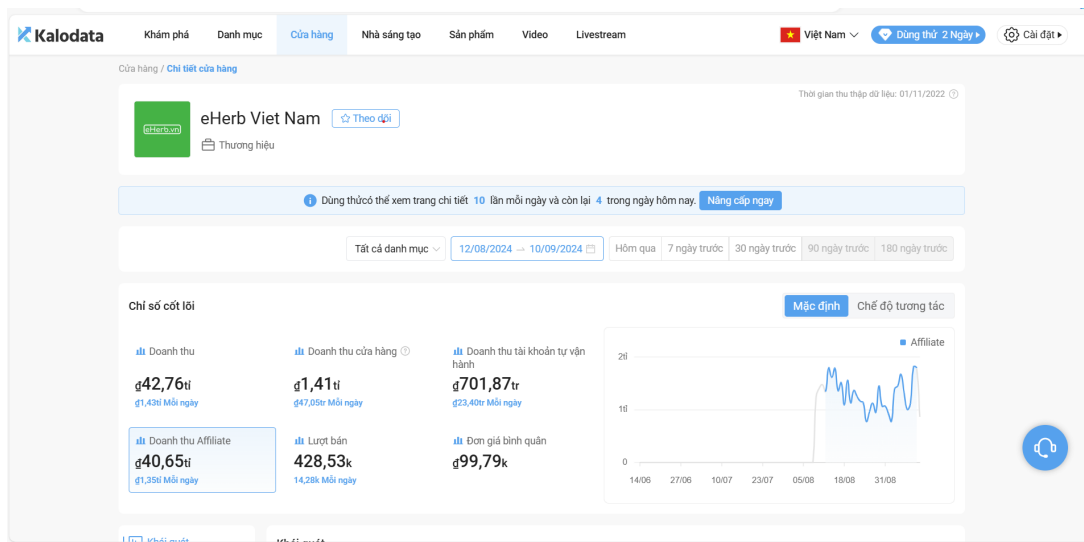
Hình trên mô tả doanh số phát triển trong 30 ngày của các mặt hàng trên TikTok qua website KaloData, điều đó chứng tỏ TikTok là một nền tảng rất tiềm năng cho các doanh nghiệp, cũng là cho các nhà phân tích dữ liệu để đưa ra những định hướng cho doanh nghiệp. Từ đó, nhóm chúng em lựa chọn mặt hàng tiềm năng nhất trong bảng xếp hạng này đó là chăm sóc sắc đẹp và cá nhân, chọn doanh nghiệp tiềm năng nhất trong mặt hàng đấy để phân tích, làm đầu vào cho bài toán.

Sơ lược về hệ thống

Bài toán: Phân tích các yếu tố liên quan đến sự tăng trưởng của doanh nghiệp eHerb Viet Nam - Doanh nghiệp tiềm năng nhất trong bảng xếp hạng của mặt hàng chăm sóc sắc đẹp và cá nhân. Dự đoán sự tăng trưởng ấy trong tương lai gần.

Nguồn dữ liệu: Dữ liệu được lấy từ website KaloData, cụ thể trong đó là tổng quan về các doanh số tăng trưởng trong 30 ngày của doanh nghiệp như: doanh thu, doanh thu cửa hàng, doanh thu tài khoản tự vận hành, doanh thu Affiliate, lượt bán, đơn giá bình quân.

Mục tiêu bài toán: Phân loại, phân cụm được các sản phẩm, các nhà sáng tạo, các video, livestream có nội dung ở các mức độ tiềm năng để đưa ra các gợi ý về chiến lược Marketing cho doanh nghiệp. Đưa ra dự đoán về doanh số trong tương lai gần của doanh nghiệp.



Hình 1.2: Doanh nghiệp eHerb Viet Nam.

1.2 Định hướng giải pháp và hạn chế.

Định hướng giải pháp.

1. Sử dụng các phương pháp Scraping để thu thập dữ liệu như dùng các thư viện Selenium, BeautifulSoup,...
2. Sử dụng các phương pháp tiền xử lý như trích xuất đặc trưng dữ liệu(feature selection)
3. Sử dụng các thuật toán phân loại, phân cụm như K-Nearest Neighbour (KNN), K-means clustering.
4. Sử dụng các mô hình dự báo dựa trên thời gian (Time forest) như Long Short Term Memory (LSTM).

Hạn chế.

1. Nguồn dữ liệu trên KaloData bị giới hạn, số lượt tương tác để xem doanh thu giới hạn 10 lần/ngày, 7 ngày/tài khoản, 30 ngày thống kê / mỗi lần xem. Do đó việc thu thập dữ liệu rất khó khăn và phải canh đúng thời gian.
2. Với doanh số của 30 ngày/ lần xem, như vậy trong quá trình học chỉ có thể lấy được tối đa 3-4 tháng dữ liệu, điều đó quá ngắn ngủi để mô hình có thể đưa ra được dự báo chính xác.
3. Ngoài nguồn dữ liệu từ KaloData ra, hầu như không còn một bên nào cung cấp dữ liệu miễn phí cho các doanh nghiệp, mặt hàng cụ thể.
4. Đây là bài toán nội bộ, thường do các doanh nghiệp đặt hàng và cung cấp dữ liệu nội bộ cho các nhà phân tích nên ít phổ biến. Thứ năm: Để khắc phục được số lượng ngày giới hạn, bắt buộc nhóm phải thu thập rất nhiều trường dữ liệu liên quan để đưa ra quan hệ giữa chúng.

Kết luận chương

Chương này đã mô tả tổng quan về bài toán từ những bước đầu như khảo sát hiện trạng thực tế để đưa ra được những lý do và động lực để lựa chọn đề tài, từ đó đưa ra hệ thống cụ thể cần xây dựng gồm bài toán, dữ liệu, mục tiêu. Cuối cùng đưa ra hướng đi cho giải pháp kỹ thuật và nhận định sơ bộ về hạn chế của bài toán.

CHƯƠNG 2. BÁO CÁO TIẾN ĐỘ TỪNG TUẦN

Tóm tắt: Chương 2 là chi tiết nội dung và kết quả mà nhóm thực hiện được trong suốt quá trình, từ những bước thu thập, xử lý, chuẩn hóa dữ liệu đến các bước tự xây dựng, thử nghiệm mô hình; trực quan, so sánh mô hình trên Orange Data Mining; thậm chí thay đổi hướng đi của bài toán, tất cả đều được đề cập trong chương này.

2.1 Tổng quan tiến độ công việc, phân công theo từng tuần.

Tiến độ thực hiện báo cáo thể hiện trong bảng dưới đây:

Trong đó các tuần, nhóm trưởng phải thực hiện lên kế hoạch chi tiết và kiểm soát các công việc của các thành viên còn lại trong nhóm.

	Ngo Văn Trọng	Cam Hải Đăng	Nguyễn Bá Hoàng Huỳnh	Bùi Anh Tú	Nguyễn Như Thiệu
Tuần 4 (06/09)	Thu thập dữ liệu về Time Series trên Kalo Data bằng thư viện Selenium, gồm 3 bảng (cửa hàng, nhà sáng tạo, sản phẩm).				
Tuần 5 (13/09)	Liên kết, tìm ra quan hệ ẩn và tạo mô hình thực thể liên kết giữa các bảng dữ liệu.	Tổng hợp các trường dữ liệu, chuẩn hóa các đơn vị của các trường dữ liệu.			
Tuần 6 (20/09)	Xây dựng mô hình LSTM đơn giản để thử nghiệm với bộ dữ liệu của 1 Shop.		Chuẩn hóa giá trị dữ liệu bằng các phương pháp Scaler và so sánh, đưa ra kết luận.		
Tuần 7 (26/09)	Xây dựng, đánh giá mô hình mùa vụ Prophet với mô hình LSTM trước đó, đưa ra kết luận, hướng đi tiếp theo cho bài toán.			Tìm hiểu và xây dựng các mô hình mùa vụ như Prophet với bộ dữ liệu của 1 cửa hàng.	Tìm hiểu đặc trưng về các mô hình mùa vụ như Prophet.
Tuần 8 (05/10)	Thực hiện nội suy tuyến tính ra trọng số của các thuộc tính với bảng xếp hạng về các nhà sáng tạo (phương pháp Linear Regression)	Trực quan 1 số mô hình Time Series trên Orange Data Mining và nhận xét.			
Tuần 9 (10/10)	- Xây dựng ý tưởng chi tiết để thử nghiệm đánh giá mô hình Neural Network. - Phát triển bài toán chính: + Đặt vấn đề từ bài toán cũ -> Chuyển mô hình dữ liệu ban đầu sang mô hình đồ thị. + Thiết kế bài toán mới dựa trên mô hình đồ thị.		Thử nghiệm mô hình Neural Network trên Orange Data Mining (thay đổi trọng số và đưa ra đánh giá).		

Hình 2.1: Tiến độ thực hiện.

2.2 Tuần 4: Thu thập và chuẩn bị dữ liệu (06/09 - 12/09)

Chủ đề tìm hiểu tuần 4: Thu thập và chuẩn bị dữ liệu

Mục tiêu của tuần: Thu thập được dữ liệu trên KaloData gồm:

1. Dữ liệu về các doanh số chung của doanh nghiệp eHurb Viet Nam theo thời gian như doanh thu, doanh thu cửa hàng, doanh thu affiliate, doanh thu tài khoản tự vận hành, lượt bán, đơn giá bình quân.
2. Thu thập được các trường dữ liệu liên quan như: Sản phẩm bán chạy, các nhà sáng tạo, video, livestreams liên quan đến sản phẩm đó.

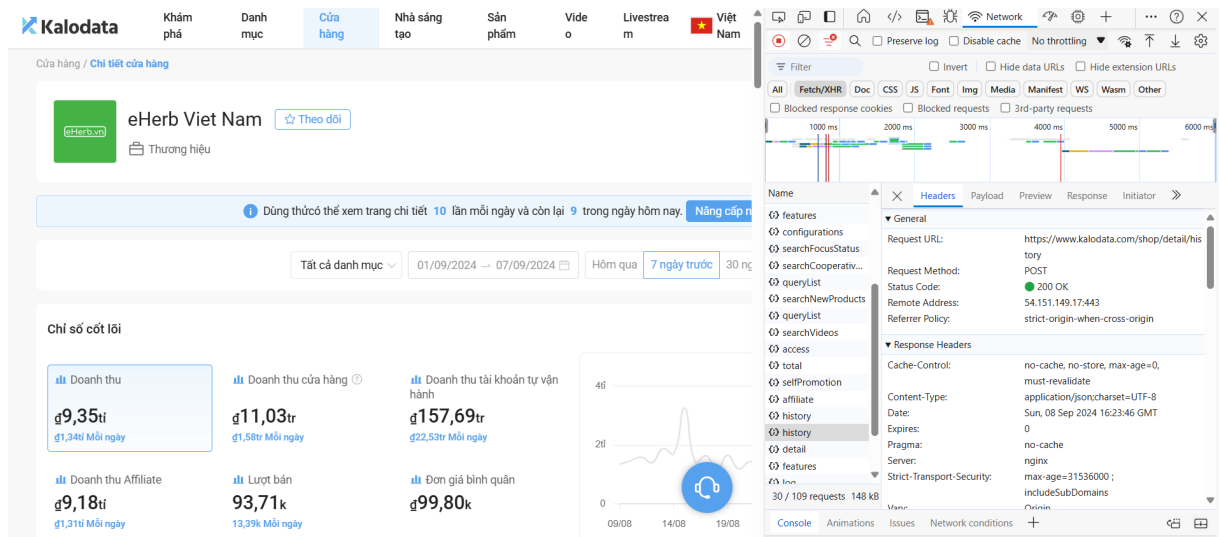
Thực hiện mục tiêu 1:

1. Xác định đặc điểm kỹ thuật của trang web cần thu thập.

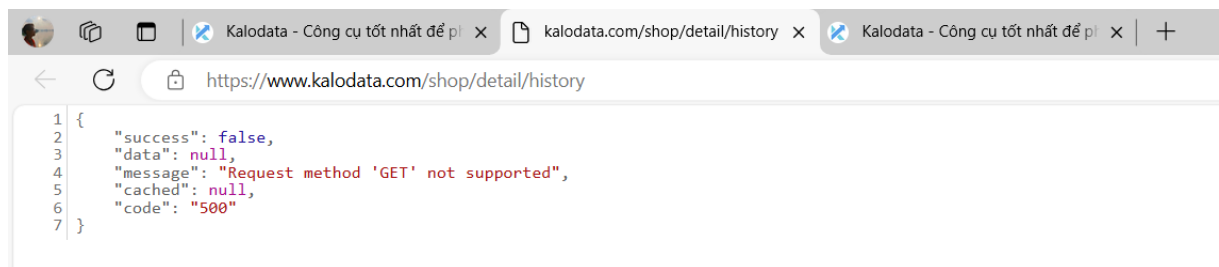
Đây là trang web không cung cấp phương thức GET nên nhóm em không thể sử dụng những phương pháp như lấy API như thông thường.

Sử dụng công cụ extension Toggle JavaScript để kiểm tra, ta nhận thấy đây là một trang web động do khi bật công cụ, dữ liệu không được hiển thị.

Cuối cùng, đây là trang web phải đăng nhập với nhiều động tác. Dữ liệu cần lấy là dữ liệu phải sử dụng nhiều động tác như : di chuột vào đồ thị thì mới hiện thẻ HTML, bấm chuột qua lại giữa các chức năng.



Hình 2.2: Mở DevTool thì thấy website chỉ cung cấp phương thức POST.



Hình 2.3: DevTool của website không hỗ trợ phương thức GET với những dữ liệu cần lấy.

2. Xác định phương pháp thu thập.

Với những đặc điểm như trên của website, nhóm em quyết định sử dụng phương pháp Scraping (thư viện Selenium) để thu thập dữ liệu với các ưu điểm sau:

Thứ nhất: Giúp giả lập thao tác người dùng, tự động hóa được quá trình đăng nhập, di chuyển qua lại giữa các chức năng, các trang web, lưu lại cookie đăng nhập.

Thứ hai: Giả lập thao tác di chuột qua đồ thị dựa trên tọa độ để hiển thị thẻ HTML.

Thứ ba: Sử dụng được các phương thức trong Selenium để tìm các thẻ và lấy nội dung trong đó.

Ngoài ra, nhóm em còn sử dụng thêm phương pháp chèn mã Script vào Console của DevTool trang web để tự động hóa việc lấy tọa độ từ đồ thị.

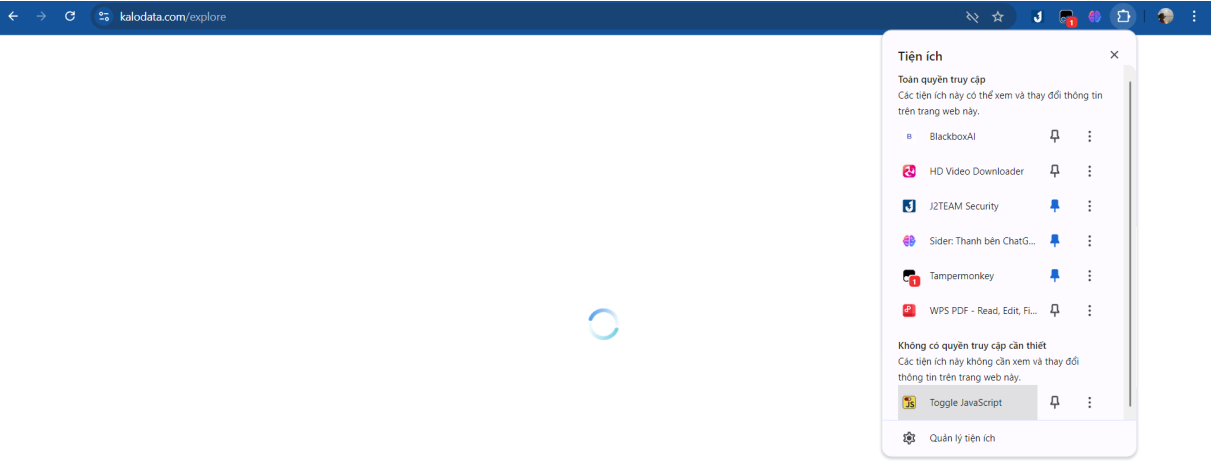
Chi tiết các phương pháp sẽ được trình bày trong phần thực hiện dưới đây

3. Quá trình thực hiện.

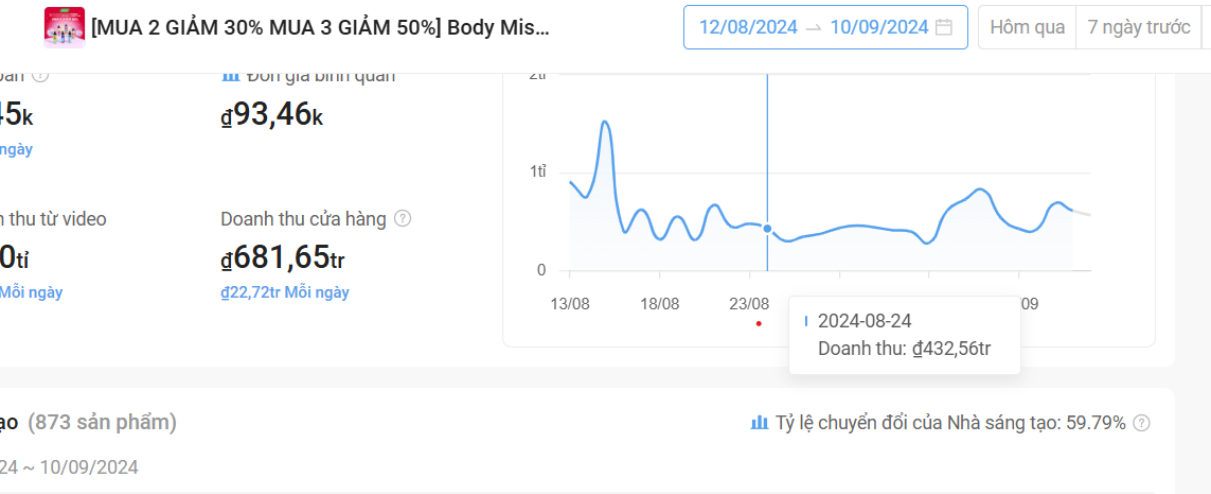
Khai báo các thư viện liên quan đến Selenium để khởi tạo trình duyệt (Edge), chờ đợi các phần tử xuất hiện và giả lập các thao tác.

Tiếp đến, nhóm tiến hành khởi tạo trình duyệt, điều hướng sang trang chi tiết để vào trang đăng nhập để đăng nhập, chờ các phần tử xuất hiện, giả lập các thao tác nhập mật khẩu và ấn đăng nhập.

Ở phần này, nhóm em sử dụng các phương thức trong Selenium như phương thức findElement() được sử dụng để



Hình 2.4: Tắt JavaScript của trang web.



Hình 2.5: Di chuột để hiển thị dữ liệu đồ thị.

định vị và lấy phần tử trên trang web dựa trên các tiêu chí như ID, tên Class hoặc XPath của Button.

Giả lập thêm một số thao tác để đến được với các doanh số cần lấy.

Điều hướng trình duyệt đến trang chi tiết.

Mỗi lần chuyển trang đều phải chờ đợi bằng hàm Sleep().

Nhấn thêm một vài Button như 'Đã hiểu', chuyển tab, '30 ngày' để đến với doanh thu 30 ngày.

```
import pickle
import csv
from selenium import webdriver
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from selenium.webdriver.edge.service import Service
from selenium.webdriver.common.by import By
from time import sleep
from selenium.webdriver.common.action_chains import ActionChains
```

Hình 2.6: Khai báo thư viện.

```
11 # Đường dẫn đến msedgedriver
12 edge_driver_path = "B:\\Documents\\DataScience Research(09 2024)\\edgedriver_win64\\msedgedriver.exe"
13 # Khởi tạo trình duyệt Edge
14 service = Service(executable_path=edge_driver_path)
15 browser = webdriver.Edge(service=service)
16 # 1. Mở trang đăng nhập
17 browser.get("https://www.kalodata.com/video/detail?id=7396229557373766919&language=vi-VN&currency=VND&region=VN&dateRange=%5B%222024-08-30%22%2C%222024-09-05%22%5D")
18 # Đợi để trang tải xong
19 sleep(2)
20 # 3. Nhập số điện thoại
21 try:
22     phone_field = browser.find_element(By.ID, "register_phone")
23     phone_field.clear()
24     phone_field.send_keys("0904708498") # Thay đổi số điện thoại nếu cần
25 except Exception as e:
26     print(f"Không tìm thấy trường số điện thoại: {e}")
27 # 4. Nhập mật khẩu
28 try:
29     password_field = browser.find_element(By.ID, "register_password")
30     password_field.clear()
31     password_field.send_keys("trongtiktok") # Thay đổi mật khẩu nếu cần
32 except Exception as e:
33     print(f"Không tìm thấy trường mật khẩu: {e}")
34 # 5. Nhấn vào nút đăng nhập
35 try:
36     submit_button = browser.find_element(By.XPATH, "//*[@type='submit' and contains(text(), 'Log in')]")
37     submit_button.click()
38     sleep(2) # Đợi trang tải xong
39 except Exception as e:
40     print(f"Không tìm thấy nút đăng nhập (submit): {e}")
41
```

Hình 2.7: Giả lập đăng nhập.

```
# 6. Điều hướng đến trang shop chi tiết sau khi đăng nhập
try:
    browser.get("https://www.kalodata.com/shop/detail?id=7494529979361168222&language=vi-VN&currency=VND&region=VN")
    sleep(1) # Đợi trang shop tải xong
except Exception as e:
    print(f"Không thể điều hướng đến trang shop: {e}")

# 7. Nhấn vào nút "Đã hiểu" để tắt thông báo
try:
    da_hieu_button = browser.find_element(By.XPATH, "//div[text()='Đã hiểu']")
    da_hieu_button.click()
    sleep(1) # Đợi thông báo tắt

    # Chuyển sang tab mới sau khi nhấn 'Đã hiểu'
    browser.switch_to.window(browser.window_handles[-1]) # Chuyển sang tab mới (tab cuối cùng mở)
    print("Đã chuyển sang tab mới.")

    sleep(1) # Đợi trang mới tải xong

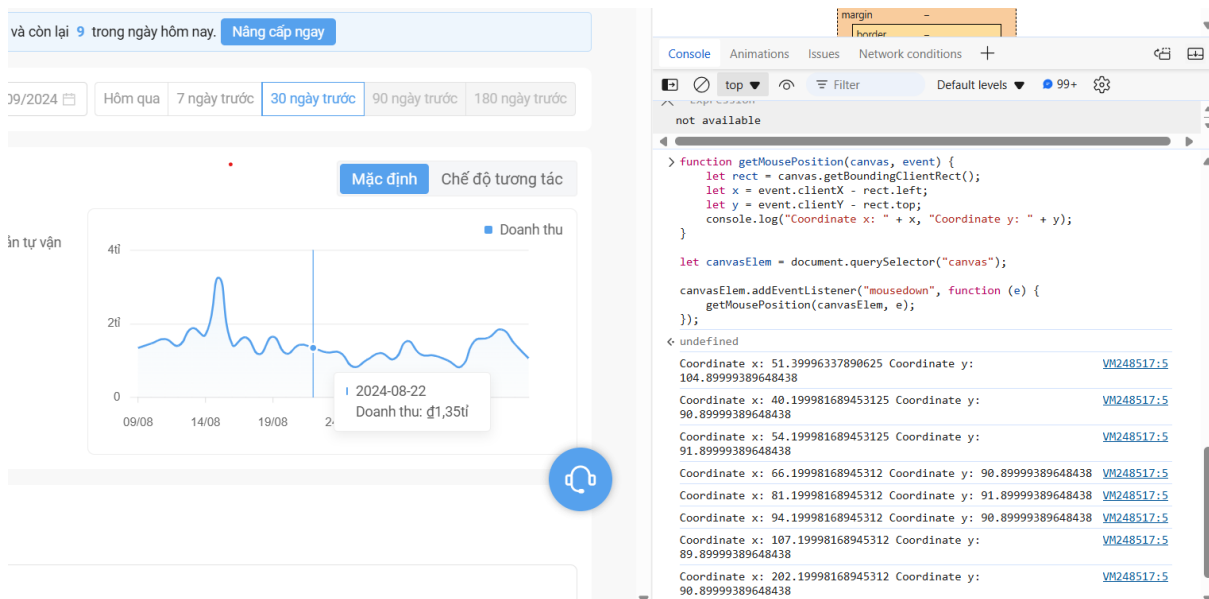
    # Quay lại tab cũ
    browser.switch_to.window(browser.window_handles[0]) # Chuyển lại về tab đầu tiên
    print("Đã quay lại tab cũ.")
    sleep(1)

except Exception as e:
    print(f"Không tìm thấy nút 'Đã hiểu': {e}")

try:
    # Chờ đến khi phân tử '30 ngày trước' hiển thị và nhấp
    button_30_days = WebDriverWait(browser, 10).until(
        EC.element_to_be_clickable((By.XPATH, "//span[contains(text(),'30 ngày trước')]"))
    )
    button_30_days.click()
    print("Đã nhấp vào nút '30 ngày trước'.")
    sleep(2)
except Exception as e:
    print(f"Không thể nhấp vào nút '30 ngày trước': {e}")
```

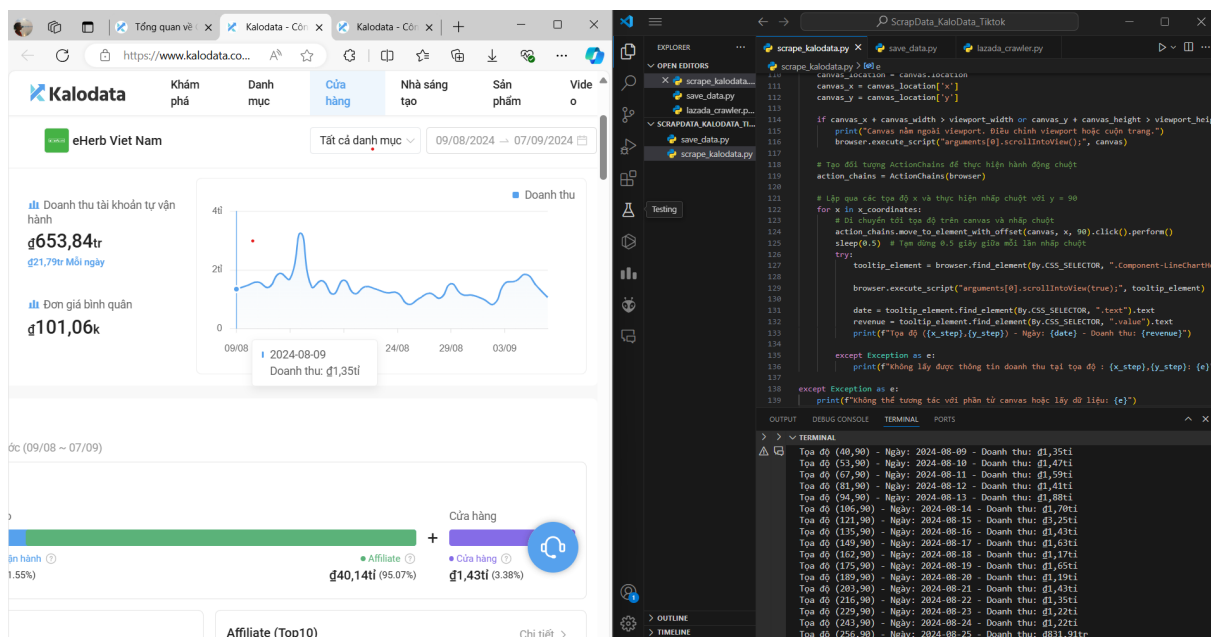
Hình 2.8: Một số thao tác để đến với trang doanh số 30 ngày.

Do đồ thị hiển thị dữ liệu chỉ khi ta di chuột vào nên phải xác định được tọa độ của đường đồ thị, lúc này nhóm em viết đoạn mã Script vào Console của website để tự động hóa trong việc lấy tọa độ đồ thị.



Hình 2.9: Script lấy tọa độ trên đồ thị.

Sau cùng là kết quả chạy được khi sử dụng Selenium để lấy dữ liệu trên đồ thị từ tọa độ lấy được.



Hình 2.10: Kết quả ví dụ với đồ thị về doanh thu.

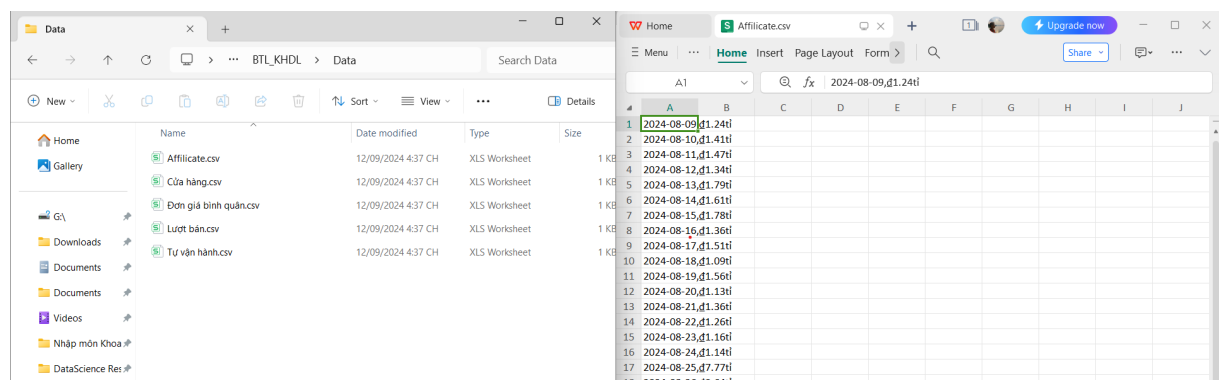
Với mỗi thông số khác nhau như doanh thu, doanh thu cửa hàng, doanh thu affiliate, doanh thu tài khoản tự vận hành, lượt bán, đơn giá bình quân lại có những đồ thị khác nhau, vì vậy nhóm em sẽ đưa ra đoạn code tổng quát để lấy toàn bộ dữ liệu và lưu vào file CSV.


```

114 action_chains = ActionChains(browser)
115 # Lặp qua các tọa độ x và thực hiện nhấp chuột với y = 90
116 for x in x_coordinates:
117     # Di chuyển tới tọa độ trên canvas và nhấp chuột
118     action_chains.move_to_element_with_offset(canvas, x, 90).click().perform()
119     sleep(0.5) # Tạm dừng 0.5 giây giữa mỗi lần nhấp chuột
120     try:
121         tooltip_element = browser.find_element(By.CSS_SELECTOR, ".Component-LineChartHoverTip")
122
123         browser.execute_script("arguments[0].scrollIntoView(true);", tooltip_element)
124
125         date = tooltip_element.find_element(By.CSS_SELECTOR, ".text").text
126         Value = tooltip_element.find_element(By.CSS_SELECTOR, ".value").text
127         Label = tooltip_element.find_element(By.CSS_SELECTOR, ".label").text
128         print(f"Tọa độ ({x_step},{y_step}) - Ngày: {date} - {Label}: {revenue}")
129
130         folder_path = r"B:\Documents\DataScience Research(09 2024)\BTL_KHDL\Data"
131         filename = f"{Label}.csv"
132         file_path = os.path.join(folder_path, filename)
133         with open(file_path, mode='a', newline='', encoding='utf-8') as file:
134             writer = csv.writer(file)
135             writer.writerow([date, Value])
136

```

Hình 2.11: Code tổng quát để lấy dữ liệu đồ thị và lưu vào file CSV.



Hình 2.12: Kết quả về dữ liệu các loại doanh số thu thập được trên máy(CSV).

Thực hiện mục tiêu 2:

Tương tự tuần 1 với các bảng dữ liệu khác.

2.3 Tuần 5: Tiền xử lý dữ liệu (13/09 - 19/09)

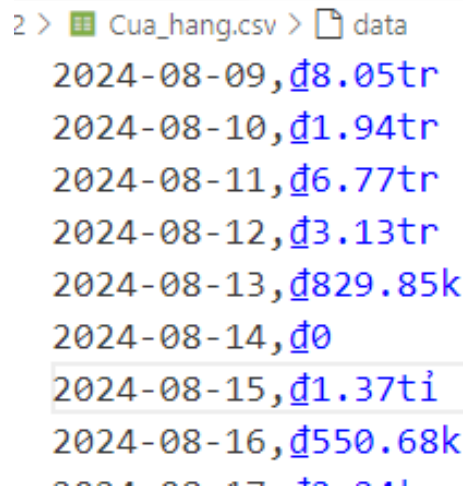
Chủ đề tìm hiểu tuần 5: Tiền xử lý dữ liệu.

Mục tiêu của tuần: Chuẩn hoá, tổng hợp dữ liệu từ các phần đã thu thập gồm:

1. Chuẩn hoá các thuộc tính của trường dữ liệu như ngày tháng, giá trị kiểu số.
2. Tổng hợp các dữ liệu thành một bảng thống nhất.
3. Tìm mối quan hệ và vẽ sơ đồ thực thể liên kết của các bảng dữ liệu như Cửa hàng, Sản phẩm, Nhà Sáng Tạo.

Thực hiện mục tiêu 1:

1. Xác định dữ liệu sắp xử lý thuộc kiểu dữ liệu nào và chuẩn hoá như thế nào.



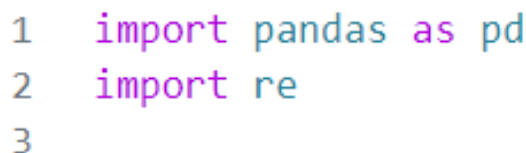
```

2 > Cua_hang.csv > data
2024-08-09, 8.05tr
2024-08-10, 1.94tr
2024-08-11, 6.77tr
2024-08-12, 3.13tr
2024-08-13, 829.85k
2024-08-14, 0
2024-08-15, 1.37tỷ
2024-08-16, 550.68k
2024-08-17, 0
    
```

Hình 2.13: Dữ liệu đã thu thập được

Như hình, dữ liệu được thu thập lưu trong csv gồm 2 cột ngày và doanh số, nhưng khi lưu chúng đều ở dạng chuỗi kí tự. Nhiệm vụ của nhóm em sẽ là chuẩn hoá ngày tháng về kiểu date và doanh số về kiểu float, đơn vị tính theo nghìn đồng.

Trước hết, để thực hiện việc chuẩn hoá, nhóm em sử dụng 2 thư viện gồm pandas để đọc ghi file csv và re để phục vụ chuẩn hoá doanh thu.



```

1 import pandas as pd
2 import re
3
    
```

Hình 2.14: Import 2 thư viện cần dùng

Sau đó tiến hành đọc file csv và đặt tên cho 2 cột để thuận tiện cho việc xử lý. Ở cột đầu tiên (cột Date) nhóm em đưa dữ liệu về format YYYY-mm-dd. Ở cột thứ hai, do các bản ghi đều có kí tự tiền tệ nên nhóm em sẽ loại bỏ tất cả.

```

35 #path
36 pathCuaHang = 'Chapter2\Cua_hang.csv'
37 data1 = pd.read_csv(pathCuaHang)
38
39 #Đặt tên cho 2 cột
40 data1.columns = ['Date', 'Cuahang']
41
42 #Chuẩn hoá ngày tháng năm theo format
43 data1['Date'] = pd.to_datetime(data1['Date'], format= '%Y-%m-%d')
44 #Bỏ toàn bộ ký tự tiền tệ trước các bản ghi
45 data1['Cuahang'] = data1['Cuahang'].replace({'đ': ''}, regex = True)
46

```

Hình 2.15: Đặt tên, chuẩn hoá ngày tháng, bỏ kí hiệu tiền tệ

Để dễ dàng chuẩn hoá, nhóm em viết riêng 1 hàm lấy toàn bộ ký tự số và dấu thập phân để loại bỏ ký tự chữ (k, tr, tỉ).

```

#Trích xuất phần số trong 1 chuỗi
def extract_numbers(text):
    numbers = re.findall(r'\d+\.\d+', text)
    if numbers:
        return float(numbers[0])
    else:
        return 0

```

Hình 2.16: Trích xuất phần số trong 1 chuỗi.

Sau đó nhóm em tiến hành loại bỏ các ký tự thừa, đơn vị sẽ là nghìn đồng để tránh tràn số. Sau khi chuẩn hoá, lưu kết quả thu được vào 1 danh sách.

```
tmp_list = list(data1['Cuahang'])
cuahang_List = []
#Do phần chữ có k, tr, tỉ nên không thể replace all như phần trước, làm thủ công
for i in tmp_list:
    if(i[-2:] == 'tỉ'):
        cuahang_List.append(extract_numbers(i)* 1e6)
    elif(i[-2:] == 'tr'):
        cuahang_List.append(extract_numbers(i)* 1e3)
    else:
        cuahang_List.append(extract_numbers(i))
```

Hình 2.17: Loại bỏ ký tự thừa.

Tương tự, nhóm em chuẩn hoá các dữ liệu về doanh thu tự vận hành, doanh thu affiliate, đơn giá trung bình và lượt bán.

Thực hiện mục tiêu 2:

1. Tổng hợp các dữ liệu đã chuẩn hoá thông qua danh sách đã tạo Sau khi hoàn tất chuẩn hoá, dữ liệu được lưu vào danh sách. Sau đó, tạo một từ điển lưu trữ tất cả các dữ liệu đã chuẩn hoá và đưa nó vào một Dataframe. Cuối cùng xuất ra thành một file csv hoàn chỉnh. .

```
full_data = {
    'Ngày: ': date_List,
    'Doanh thu Affiliate (nghìn đồng)': affiliate_List,
    'Doanh thu cửa hàng (nghìn đồng)': cuahang_List,
    'Đơn giá bình quân (nghìn đồng)': dongia_List,
    'Lượt bán (nghìn lượt)': luotban_List,
    'Doanh thu tài khoản tự vận hành (nghìn đồng)': tvh_List,
}

df = pd.DataFrame(full_data)
```

Hình 2.18: Lưu vào file CSV.

Kết quả cuối cùng thu được như sau:

	A	B	C	D	E	F	G	H
1		Ngày:	Doanh thu Affiliate (nghìn đồng)	Doanh thu cửa hàng (nghìn đồng)	Đơn giá bình quân (nghìn đồng)	Lượt bán (nghìn lượt)	Doanh thu tài khoản tự vận hành (nghìn đồng)	
2	0	2024-08-10	1410000	1940	118,91	12,35	10440	
3	1	2024-08-11	1470000	6770	119,14	13,32	12500	
4	2	2024-08-12	1340000	3130	104,27	13,5	13060	
5	3	2024-08-13	1790000	829,85	117,69	15,97	16180	
6	4	2024-08-14	1610000	0	114,85	14,81	17820	
7	5	2024-08-15	1780000	1370000	114,68	28,35	24310	
8	6	2024-08-16	1360000	550,68	88,45	16,14	27180	
9	7	2024-08-17	1510000	2240	99,19	16,46	27210	
10	8	2024-08-18	1090000	277,33	85,92	13,65	13670	
11	9	2024-08-19	1560000	333,37	97,1	17,01	19440	
12	10	2024-08-20	1130000	138,42	84,37	14,14	16180	
13	11	2024-08-21	1360000	734,76	107,54	13,28	19850	
14	12	2024-08-22	1260000	192,21	100,15	13,43	30490	
15	13	2024-08-23	1160000	456,41	99,74	12,28	14570	
16	14	2024-08-24	1140000	176,16	98,57	12,35	19820	
17	15	2024-08-25	7770000	0	94,8	8,78	26990	
18	16	2024-08-26	9640000	0	98,44	10,36	18710	
19	17	2024-08-27	1120000	1020000	95,91	12,57	34350	
20	18	2024-08-28	9490000	1240000	110,83	9,4	34600	
21	19	2024-08-29	1450000	11380000	85,77	17,82	20440	
22	20	2024-08-30	1070000	554,77	91,08	12,81	38430	
23	21	2024-08-31	1070000	139,7	91,33	12,45	26340	
24	22	2024-09-01	9290000	0	90,78	11,06	13330	
25	23	2024-09-02	7830000	442,21	95,46	8,83	10730	
26	24	2024-09-03	1400000	1740000	93,68	16,41	19390	
27	25	2024-09-04	1450000	0	107,94	15,01	41880	
28	26	2024-09-05	1670000	8850000	109,51	16,85	25640	
29	27	2024-09-06	1330000	0	98,83	14,57	28140	
30	28	2024-09-07	1010000	0	96,8	10,99	18580	

Hình 2.19: Kết quả cuối cùng khi lưu vào file CSV.

Tương tự với các bộ dữ liệu khác như nhà sáng tạo theo sản phẩm.

Thực hiện mục tiêu 3:

1. Dữ liệu về Các Nhà sáng tạo của 1 sản phẩm.

A	B	C	D	E	F	G	H
	Nhà sáng tạo	Số người theo dõi (nghìn người)	Doanh thu (nghìn đồng)	Lượt bán (nghìn lượt)	Doanh thu từ video (nghìn đồng)	Doanh thu Live (nghìn đồng)	
0	@tiemtaphoa_so37	6	2500000	27,35	2500000	0	
1	@haithichreview	2,89	1450000	15,86	1450000	0	
2	@nhunhoi22	49,53	1020000	11,26	1020000	0	
3	@thaiyublackbi149	3020	1010000	11,01	1010000	0	
4	@cocomanshop	2,6	905850	9,87	905850	0	
5	@is_yangg.22	347,14	867040	9,44	867040	0	
6	@chuyen.thang.thanh	145,62	864160	9,36	864160	0	
7	@chucareviewkhongbooking	734,88	759150	8,29	759150	0	
8	@annie.riviu	31,48	660960	7,3	657080	3880	
9	@vietgreview	106,15	647460	7	647460	0	
10	@nemdila	108,46	541160	5,8	541160	0	
11	@hocphatcungphu	11,87	368940	4,02	368940	0	
12	@chichichanhchanh04	33,23	302500	3,29	302500	0	
13	@dung_review_8396	56,84	279370	3,1	0	279370	
14	@chiemreview43	1,92	264390	2,86	264390	0	
15	@tiembodyst5	4,66	178000	1,93	835,01	177170	
16	@herb.vietnam	114,66	157450	1,71	116730	40720	
17	@lingg2809	7660	115950	1,28	115950	0	
18	@tu_guyen1401	714,96	102510	1,09	35520	66990	
19	@taphoathomphuc777	7,46	92710	975	92710	0	
20	@review.cnh.p0	91,23	85800	996	0	85800	
21	@hula.nkc	1,95	82260	910	82260	0	
22	@ngocgan1194	16,6	79370	764	79370	0	
23	@trongaden.99	1200	73200	779	73200	0	
24	@haithuan68	1,64	66450	728	148,81	66300	
25	@nuochoanmunchinhhang1	19,13	66320	734	0	66320	
26	@lthoi.offical	10,88	58820	665	58820	0	
27	@nhatanhreview_	752,62	56490	568	56490	0	
28	@degreyyn	403,05	47940	568	47940	0	
29	@toantungtung20	189,31	45370	503	45370	0	
30	@ngochuyen0595	283,16	45070	511	45070	0	
31	@_baby20233	15,7	38430	420	0	38430	

Hình 2.20: Dữ liệu về các nhà sáng tạo theo sản phẩm.

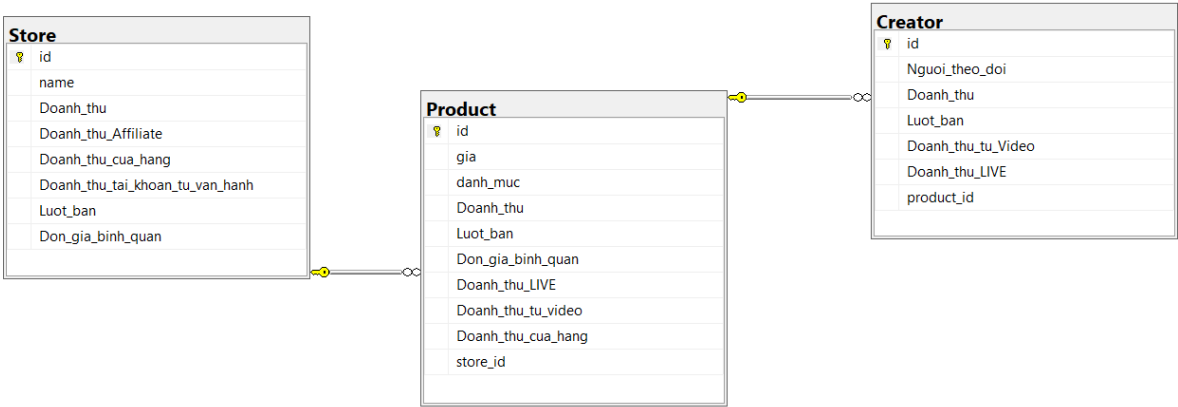
2. Sơ đồ bảng quan hệ dữ liệu.

Các bảng đều có đặc điểm chung là liên quan đến doanh thu.

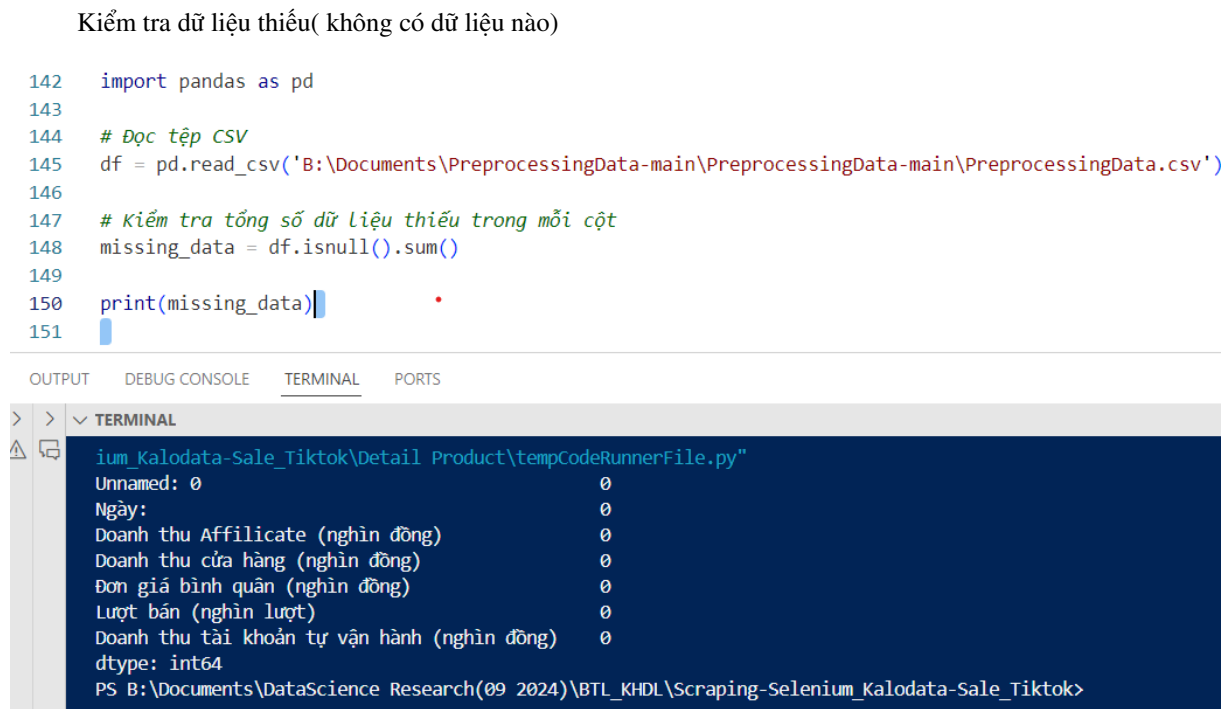
Các loại doanh thu từ bảng Store phụ thuộc vào doanh thu từ các Product, cụ thể là phụ thuộc vào tổng doanh thu từ các bảng Product.

Các loại doanh thu từ bảng Product lại phụ thuộc vào tổng các loại doanh thu từ các Creator theo sản phẩm đó.

Từ đó nhóm em đưa ra được những mối quan hệ ẩn trong các bảng dữ liệu.



Hình 2.21: Bảng quan hệ dữ liệu.



Hình 2.22: Kiểm tra dữ liệu thiếu.

2.4 Tuần 6: Chuẩn hóa lại và thử nghiệm trên 1 số mô hình (20/09 - 26/09)

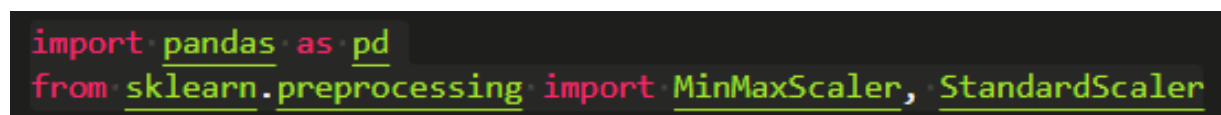
Chủ đề tìm hiểu tuần 6: Chuẩn hóa và thử nghiệm trên mô hình.

Mục tiêu của tuần:

1. Bình thường hóa dữ liệu bằng phương pháp MinMaxScaler.
2. Chuẩn hóa dữ liệu bằng phương pháp StandardScaler.
3. Phân tích các phương pháp để đưa ra đầu vào cuối cùng cho mô hình dự đoán.
4. Thử nghiệm việc dự đoán doanh thu trên mô hình LSTM.

Thực hiện mục tiêu 1:

1. Đầu tiên, để thực hiện việc bình thường hóa, nhóm em sử dụng 2 thư viện pandas để xử lý dữ liệu và MinMaxScaler để bình thường hóa dữ liệu.



Hình 2.23: Khai báo thư viện

2. Sau khi đọc dữ liệu từ hai file csv và loại bỏ cột số thứ tự (cột dư thừa), hiển thị toàn bộ dữ liệu như sau:

	Ngày:	Doanh thu Affilicate (nghìn đồng)	Doanh thu cửa hàng (nghìn đồng)	Đơn giá bình quân (nghìn đồng)	Lượt bán (nghìn lượt)	Doanh thu tài khoản tự vận hành (nghìn đồng)
0	2024-08-10	1410000.0	1940.00	118.91	12.35	10440.0
1	2024-08-11	1470000.0	6770.00	119.14	13.32	12500.0
2	2024-08-12	1340000.0	3130.00	104.27	13.50	13060.0
3	2024-08-13	1790000.0	829.85	117.69	15.97	16180.0
4	2024-08-14	1610000.0	0.00	114.85	14.81	17820.0
5	2024-08-15	1780000.0	1370000.00	114.68	28.35	24310.0
6	2024-08-16	1360000.0	550.68	88.45	16.14	27180.0
7	2024-08-17	1510000.0	2240.00	99.19	16.46	27210.0
8	2024-08-18	1090000.0	277.33	85.92	13.65	13670.0
9	2024-08-19	1560000.0	333.37	97.10	17.01	19440.0
10	2024-08-20	1130000.0	138.42	84.37	14.14	16180.0
11	2024-08-21	1360000.0	734.76	107.54	13.28	19850.0
12	2024-08-22	1260000.0	192.21	100.15	13.43	30490.0
13	2024-08-23	1160000.0	456.41	99.74	12.28	14570.0
14	2024-08-24	1140000.0	176.16	98.57	12.35	19820.0
15	2024-08-25	7770000.0	0.00	94.80	8.78	26990.0
16	2024-08-26	9640000.0	0.00	98.44	10.36	18710.0

Hình 2.24: Dữ liệu doanh thu

	Nhà sáng tạo	Số người theo dõi (nghìn người)	Doanh thu (nghìn đồng)	Lượt bán (nghìn lượt)	Doanh thu từ video (nghìn đồng)	Doanh thu Live (nghìn đồng)
0	@tiemtaphoa_so37	6.00	2500000.00	27.35	2500000.00	0.00
1	@haithichreview	2.89	1450000.00	15.86	1450000.00	0.00
2	@nhunhoi22	49.53	1020000.00	11.26	1020000.00	0.00
3	@thaiublackbi149	3020.00	1010000.00	11.01	1010000.00	0.00
4	@cocomanshop	2.60	905850.00	9.87	905850.00	0.00
...
267	@thao.ntbt	7.63	355.07	6.00	355.07	0.00
268	@trangsdnb9c	80.00	354.69	4.00	354.69	0.00
269	@agnthu_06	1.24	353.89	4.00	353.89	0.00
270	@tainucrosp2	9.99	353.80	4.00	0.00	353.80
271	@moc_green	12.66	353.20	4.00	88.44	264.76

Hình 2.25: Dữ liệu về các nhà sáng tạo từ một sản phẩm

Như hình trên, loại trừ cột ngày và tên nhà sáng tạo, các cột còn lại đều hiển thị giá trị có đơn vị nghìn đồng, nghìn người và nghìn lượt. Nhiệm vụ của nhóm là bình thường hóa dữ liệu sao cho các giá trị nằm trong phạm vi [0, 1].

Tiếp theo, nhóm em thực hiện bình thường hóa dữ liệu các giá trị liên tục (trừ cột Ngày đối với dữ liệu doanh thu và Nhà sáng tạo đối với dữ liệu nhà sáng tạo từ một sản phẩm) bằng cách sử dụng MinMaxScaler. MinMaxScaler được dùng để đưa các giá trị của các cột được chọn vào phạm vi [0, 1], làm cho dữ liệu dễ dàng so sánh và sử dụng trong các bước phân tích hoặc mô hình hóa tiếp theo.

```
# Bình thường hóa dữ liệu theo MinMaxScaler
min_max_scaler = MinMaxScaler()
normalized_revenue_data = df1.copy()
normalized_revenue_data[['Doanh thu Affilicate (nghìn đồng)', 'Doanh thu cửa hàng (nghìn đồng)', 'Đơn giá bình quân (nghìn đồng)',
                          'Lượt bán (nghìn lượt)', 'Doanh thu tài khoản tự vận hành (nghìn đồng)']] = min_max_scaler.fit_transform(
    normalized_revenue_data[['Doanh thu Affilicate (nghìn đồng)', 'Doanh thu cửa hàng (nghìn đồng)', 'Đơn giá bình quân (nghìn đồng)',
                              'Lượt bán (nghìn lượt)', 'Doanh thu tài khoản tự vận hành (nghìn đồng)']]
normalized_product_creator_data = df2.copy()
normalized_product_creator_data[['Số người theo dõi (nghìn người)', 'Doanh thu (nghìn đồng)', 'Lượt bán (nghìn lượt)',
                                'Doanh thu từ video (nghìn đồng)', 'Doanh thu Live (nghìn đồng)']] = min_max_scaler.fit_transform(
    normalized_product_creator_data[['Số người theo dõi (nghìn người)', 'Doanh thu (nghìn đồng)', 'Lượt bán (nghìn lượt)',
                                     'Doanh thu từ video (nghìn đồng)', 'Doanh thu Live (nghìn đồng)']]
```

Hình 2.26: Bình thường hóa dữ liệu bằng MinMaxScaler

Cuối cùng, nhóm em đổi tên các cột và xuất dữ liệu sau khi bình thường hóa ra file csv.

```
normalized_revenue_data.columns = ['Ngày', 'Doanh thu Affiliate', 'Doanh thu cửa hàng', 'Đơn giá bình quân', 'Lượt bán', 'Doanh thu tài khoản tự vận hành']
normalized_product_creator_data.columns = ['Nhà sáng tạo', 'Số người theo dõi', 'Doanh thu (nghìn đồng)', 'Lượt bán', 'Doanh thu từ video', 'Doanh thu Live']
normalized_revenue_data.to_csv('Normalized_revenue_data.csv')
normalized_product_creator_data.to_csv('Normalized_product_creator_data.csv')
```

Hình 2.27: Đổi tên các cột và xuất file csv

Kết quả cuối cùng sau khi xuất ra file csv:

	Ngày	Doanh thu Affiliate	Doanh thu cửa hàng	Đơn giá bình quân	Lượt bán	Doanh thu tài khoản tự vận hành
0	2024-08-10	0.046349942	0.000170475	0.993385102	0.182422075	0
1	2024-08-11	0.053302433	0.000594903		1	0.231987736
2	2024-08-12	0.038238702	0.000275044	0.572332471	0.241185488	0.083333333
3	2024-08-13	0.090382387	7.29218E-05	0.958297383	0.36739908	0.182569975
4	2024-08-14	0.069524913	0	0.876617774	0.308124681	0.234732824
5	2024-08-15	0.089223638	0.120386643	0.871728502	1	0.441157761
6	2024-08-16	0.040556199	4.83902E-05	0.117342537	0.376085846	0.532442748
7	2024-08-17	0.057937428	0.000196837	0.426229508	0.392437404	0.533396947
8	2024-08-18	0.009269988	2.43699E-05	0.04457866	0.248850281	0.102735369
9	2024-08-19	0.06373117	2.92944E-05	0.366120219	0.420541645	0.286259542
10	2024-08-20	0.013904983	1.21634E-05	0	0.273888605	0.182569975
11	2024-08-21	0.040556199	6.45659E-05	0.666379062	0.229943792	0.299300254
12	2024-08-22	0.028968714	1.68902E-05	0.453839517	0.237608585	0.637722646
13	2024-08-23	0.017381228	4.01063E-05	0.442047742	0.178845171	0.131361323
14	2024-08-24	0.015063731	1.54798E-05	0.408398044	0.182422075	0.298346056
15	2024-08-25	0.783314021	0	0.299971124	0	0.526399491
16	2024-08-26	1	0	0.404659189	0.08073582	0.263040712
17	2024-08-27	0.012746234	0.089630931	0.331895312	0.193663771	0.760496183
18	2024-08-28	0.982618772	0.108963093	0.761000863	0.031681145	0.768447837
19	2024-08-29	0.050984936	1	0.040264596	0.461931528	0.318066158
20	2024-08-30	0.008952491	4.87496E-05	0.192982456	0.20592744	0.890267176
21	2024-08-31	0.006952491	1.22759E-05	0.200172563	0.187531937	0.505725191
22	2024-09-01	0.959443801	0	0.184354328	0.116504854	0.09192112
23	2024-09-02	0.790266512	3.88585E-05	0.318953121	0.002554931	0.009223919
24	2024-09-03	0.045191194	0.152899824	0.267759563	0.389882473	0.284669211
25	2024-09-04	0.050984936	0	0.677883233	0.318344405	1
26	2024-09-05	0.076477404	0.777680141	0.723037101	0.412365866	0.48346056
27	2024-09-06	0.037079954	0	0.415875755	0.295861012	0.562977099
28	2024-09-07	0	0	0.357492091	0.112927951	0.258905852

Hình 2.28: Dữ liệu doanh thu sau khi bình thường hóa

	Nhà sáng tạo	Số người theo dõi	Doanh thu	Lượt bán	Doanh thu từ video	Doanh thu Live
0	@tiemaphoa_soc37	0.000650217		1	0.026394347	1
1	@haithichreview	0.000244158	0.579940654		0.014845564	0.58
2	@nhunhoi22	0.006333742	0.40791635		0.01022203	0.408
3	@thaiavublackbi149	0.394175203	0.403915785		0.009970751	0.404
4	@cocomanshop	0.000206294	0.362249899		0.008824919	0.36234
5	@ig_yangg_22	0.045191396	0.346723705		0.008392719	0.346816
6	@chuyen.thang.thanh	0.018879799	0.345571542		0.00631231	0.345664
7	@chucareviewkhongbooking	0.095816936	0.303561607		0.007236835	0.30366
8	@annie_rvnu	0.003977031	0.264280057		0.006241771	0.262832
9	@vietcgreview	0.013726371	0.258879294		0.005940236	0.258984
10	@nemdlam	0.014027978	0.216353286		0.004734097	0.216464
11	@hocphatcungnhu	0.001416638	0.147455553		0.00294499	0.147576
12	@chichichanhchanh04	0.004205521	0.120875797		0.002211255	0.121
13	@dung_review_8396	0.007288177	0.11162249		0.002020283	0
14	@chiemreview43	0.000117509	0.105629643		0.001779055	0.105756
15	@tiembodystmist5	0.000475259	0.071068761		0.000844297	0.000334004
16	@eherb_vietnam	0.014837485	0.062847599		0.000623172	0.046692
17	@lingg2809	1	0.046245254		0.000190972	0.04638
18	@tu_quyen1401	0.093216068	0.040868494		0	0.014208
19	@taphoathomphuc777	0.000840843	0.03694794		0.978892563	0.037084
20	@review_cnh.p0	0.011778331	0.034183549		1	0
21	@hula_nkc	0.000121426	0.032767349		0.913560021	0.032904
22	@ngocngan1194	0.002034213	0.031611186		0.766813079	0.031748
23	@trongaden_99	0.15654565	0.029142837		0.781889819	0.02928
24	@haithuan68	8.09507E-05	0.026442456		0.730628901	0.000059524
25	@nuochoanamnuchinhhang1	0.002364545	0.026390448		0.736659597	0
26	@thoai_official	0.001287378	0.023390025		0.667306591	0.023528
27	@nhathanhreview_	0.098133172	0.022457893		0.569810335	0.022596
28	@degreyvn	0.052491324	0.01903741		0.569810335	0.019176
29	@toantungtung20	0.024584214	0.018009264		0.504477792	0.018148
30	@ngochuyen0595	0.036837803	0.017889247		0.51251872	0.018028

Hình 2.29: Dữ liệu về các nhà sáng tạo từ một sản phẩm sau khi bình thường hóa

Thực hiện mục tiêu 2:

1. Nhóm em thực hiện chuẩn hóa dữ liệu các giá trị liên tục (trừ cột Ngày đối với dữ liệu doanh thu và Nhà sáng tạo đối với dữ liệu nhà sáng tạo từ một sản phẩm) bằng cách sử dụng StandardScaler. StandardScaler được dùng để đưa các giá trị về phân phối chuẩn với trung bình là 0 và độ lệch chuẩn là 1.

```
# Chuẩn hóa dữ liệu theo StandardScaler
standard_scaler = StandardScaler()
standardized_revenue_data = df1.copy()
standardized_revenue_data[['Doanh thu Affiliate (nghìn đồng)', 'Doanh thu cửa hàng (nghìn đồng)', 'Đơn giá bình quân (nghìn đồng)',
                           'Lượt bán (nghìn lượt)', 'Doanh thu tài khoản tự vận hành (nghìn đồng)']] = standard_scaler.fit_transform(
    standardized_revenue_data[['Doanh thu Affiliate (nghìn đồng)', 'Doanh thu cửa hàng (nghìn đồng)', 'Đơn giá bình quân (nghìn đồng)',
                                'Lượt bán (nghìn lượt)', 'Doanh thu tài khoản tự vận hành (nghìn đồng)']]
)
standardized_product_creator_data = df2.copy()
standardized_product_creator_data[['Số người theo dõi (nghìn người)', 'Doanh thu (nghìn đồng)', 'Lượt bán (nghìn lượt)',
                                   'Doanh thu từ video (nghìn đồng)', 'Doanh thu Live (nghìn đồng)']] = standard_scaler.fit_transform(
    standardized_product_creator_data[['Số người theo dõi (nghìn người)', 'Doanh thu (nghìn đồng)', 'Lượt bán (nghìn lượt)',
                                        'Doanh thu từ video (nghìn đồng)', 'Doanh thu Live (nghìn đồng)']]
)
```

Hình 2.30: Chuẩn hóa dữ liệu bằng StandardScaler

2. Cuối cùng, nhóm em đổi tên các cột và xuất dữ liệu sau khi chuẩn hóa ra file csv.

```
standardized_revenue_data.columns = ['Ngày', 'Doanh thu Affiliate', 'Doanh thu cửa hàng', 'Đơn giá bình quân', 'Lượt bán', 'Doanh thu tài khoản tự vận hành']
standardized_product_creator_data.columns = ['Nhà sáng tạo', 'Số người theo dõi', 'Doanh thu', 'Lượt bán', 'Doanh thu từ video', 'Doanh thu Live']
standardized_revenue_data.to_csv('Standardized_revenue_data.csv')
standardized_product_creator_data.to_csv('Standardized_product_creator_data.csv')
```

Hình 2.31: Dữ liệu doanh thu

	Nhà sáng tạo	Số người theo dõi (nghìn người)	Doanh thu (nghìn đồng)	Lượt bán (nghìn lượt)	Doanh thu từ video (nghìn đồng)	Doanh thu Live (nghìn đồng)
0	@tiemtaphoa_so37	6.00	2500000.00	27.35	2500000.00	0.00
1	@haithichreview	2.89	1450000.00	15.86	1450000.00	0.00
2	@nhunhoi22	49.53	1020000.00	11.26	1020000.00	0.00
3	@thaiublackbi149	3020.00	1010000.00	11.01	1010000.00	0.00
4	@cocomanshop	2.60	905850.00	9.87	905850.00	0.00
...
267	@thao.ntbt	7.63	355.07	6.00	355.07	0.00
268	@trangsdnb9c	80.00	354.69	4.00	354.69	0.00
269	@agnthu_06	1.24	353.89	4.00	353.89	0.00
270	@taiucrosp2	9.99	353.80	4.00	0.00	353.80
271	@moc_green	12.66	353.20	4.00	88.44	264.76

Hình 2.32: Dữ liệu về các nhà sáng tạo từ một sản phẩm

Kết quả cuối cùng sau khi xuất ra file csv:

	Ngày	Doanh thu Affiliate	Doanh thu cửa hàng	Đơn giá bình quân	Lượt bán	Doanh thu tài khoản tự vận hành
0	2024-08-10	-0.432856418	-0.341972634	1.831159097	-0.444625803	-1.413559059
1	2024-08-11	-0.411714147	-0.340098802	1.853918071	-0.177375294	-1.163356049
2	2024-08-12	-0.457490523	-0.341510965	0.382500903	-0.127782416	-1.095339696
3	2024-08-13	-0.299128419	-0.342403325	1.710437581	0.552742077	-0.716391447
4	2024-08-14	-0.36247326	-0.342725271	1.429413724	0.23314353	-0.5172007
5	2024-08-15	-0.302647577	0.188775765	1.412591873	3.963630021	0.271060242
6	2024-08-16	-0.450452207	-0.342511631	-1.182920724	0.599579795	0.619644049
7	2024-08-17	-0.397664839	-0.341856247	-0.120175574	0.687744912	0.623287782
8	2024-08-18	-0.545469469	-0.342617679	-1.433269442	-0.086455017	-1.021250455
9	2024-08-19	-0.38006905	-0.342595937	-0.326985384	0.839278706	-0.320439109
10	2024-08-20	-0.531392838	-0.34267157	-1.586645139	0.048547817	-0.716391447
11	2024-08-21	-0.450452207	-0.342440215	0.706074147	-0.188395933	-0.270641422
12	2024-08-22	-0.485643786	-0.342650701	-0.025181594	-0.147068535	1.021669275
13	2024-08-23	-0.520835364	-0.342548203	-0.065751939	-0.463911922	-0.91193846
14	2024-08-24	-0.52787368	-0.342656928	-0.181525852	-0.444625803	-0.274285155
15	2024-08-25	1.805327986	-0.342725271	-0.554575128	-1.428217884	0.596567072
16	2024-08-26	2.463410507	-0.342725271	-0.194389621	-0.992902621	-0.409103283
17	2024-08-27	-0.534911996	0.052990829	-0.444738338	-0.384012285	1.490496276
18	2024-08-28	2.410623139	0.13834136	1.031626433	-1.25739797	1.520860719
19	2024-08-29	-0.418779786	4.072224938	-1.448112251	1.062446657	-0.198981337
20	2024-08-30	-0.552507785	-0.342510044	-0.9226768	-0.317888448	1.986043987
21	2024-08-31	-0.552507785	-0.342671073	-0.897938784	-0.417074204	0.51761952
22	2024-09-01	2.340239982	-0.342725271	-0.952362419	-0.800041429	-1.062546098
23	2024-09-02	1.826442933	-0.342553712	-0.489266767	-1.414442084	-1.378336305
24	2024-09-03	-0.436375576	0.33231984	-0.665401438	0.673969112	-0.326511997
25	2024-09-04	-0.418779786	-0.342725271	0.745654972	0.288246728	2.405073301
26	2024-09-05	-0.341358313	3.090693829	0.90100971	0.795196148	0.432599079
27	2024-09-06	-0.461009681	-0.342725271	-0.155798316	0.167019693	0.73624351
28	2024-09-07	-0.573622732	-0.342725271	-0.356671003	-0.819327548	-0.424892793

Hình 2.33: Dữ liệu doanh thu sau khi chuẩn hóa

	Nhà sáng tạo	Số người theo dõi	Doanh thu	Lượt bán	Doanh thu từ video	Doanh thu Live
0	@tiemtaphoa_so37	-0.277479653	10.71194165	-0.291291034	10.74055864	-0.227358139
1	@haithichreview	-0.283158202	6.11273048	-0.363624859	6.138492237	-0.227358139
2	@nhunhoi22	-0.197998232	4.229243999	-0.392583571	4.25383647	-0.227358139
3	@thaiublackbi149	5.225782543	4.185441988	-0.394157414	4.210007266	-0.227358139
4	@cocomanshop	-0.283687713	3.729244041	-0.401334138	3.753526108	-0.227358139
5	@ig_yangg_22	0.345407827	3.559248436	-0.404041148	3.583424968	-0.227358139
6	@chuyen.thang.thanh	-0.022547514	3.546633457	-0.40544778	3.570802157	-0.227358139
7	@chucareviewkhongbooking	1.053382241	3.086668538	-0.411280826	3.110551687	-0.227358139
8	@annie.riviu	-0.230955725	2.65657659	-0.417513244	2.663187003	-0.051345926
9	@vietcgreview	-0.094615782	2.597443875	-0.419401856	2.621023309	-0.227358139
10	@nemdila	-0.090397953	2.131828496	-0.426956303	2.155118872	-0.227358139
11	@hocphatcungnhu	-0.266761621	1.37747026	-0.438162065	1.400292323	-0.227358139
12	@chichichanhchanh04	-0.2277604	1.086449697	-0.442757687	1.109091092	-0.227358139
13	@dung_review_8396	-0.184650904	0.985135645	-0.443953807	-0.216742325	12.44597481
14	@chiemreview43	-0.284929325	0.919520233	-0.445464697	0.942057996	-0.227358139
15	@tiembodystmist5	-0.279926359	0.541114658	-0.451319393	-0.213082543	7.809776838
16	@eherb.vietnam	-0.079077374	0.451101525	-0.452704375	0.294875972	1.619862815
17	@lingg2809	13.69795792	0.269323179	-0.455411385	0.291457294	-0.227358139
18	@tu_quyen1401	1.017010316	0.210453276	-0.456607505	-0.061060993	2.811574368
19	@taphoathomphuc777	-0.274813839	0.167527305	5.674518347	0.189598224	-0.227358139
20	@review_cnh.p0	-0.121858208	0.137260115	5.806721162	-0.216742325	3.664870687
21	@hula.nkc	-0.284874548	0.121754203	5.26531916	0.143796706	-0.227358139
22	@ngocngan1194	-0.258125115	0.109095422	4.346194832	0.131130066	-0.227358139
23	@trongaden.99	1.902644787	0.082069581	4.440625414	0.104087447	-0.227358139
24	@haithuan68	-0.285440577	0.052503224	4.119561436	-0.216090103	2.780273227
25	@nuochoanamnuchinhhang1	-0.253505588	0.051933797	4.157333669	-0.216742325	2.781180506
26	@thoai.offical	-0.268569262	0.019082289	3.722952993	0.041061052	-0.227358139
27	@nhatanhreview_	1.085773705	0.00887642	3.112301898	0.030848848	-0.227358139
28	@degreyvn	0.447493889	-0.028574299	3.112301898	-0.00625122	-0.227358139
29	@toantungtung20	0.057226051	-0.039831416	2.703102711	-0.017889227	-0.227358139
30	@ngochuyen0595	0.228586754	-0.041145476	2.753465688	-0.019204103	-0.227358139

Hình 2.34: Dữ liệu về các nhà sáng tạo từ một sản phẩm sau khi chuẩn hóa

Thực hiện mục tiêu 3:

- Mục tiêu thứ nhất của bài toán “Phân tích tăng trưởng của doanh nghiệp theo thời gian và dự báo xu hướng trong tương lai” là dự đoán các loại doanh thu trong tương lai dựa trên các giá trị quá khứ, sử dụng mô hình LSTM.
- Ưu và nhược điểm của hai phương pháp: Vì dữ liệu về doanh thu là không thể âm, việc chuẩn hóa theo phương pháp
 - StandardScaler sẽ đưa ra giá trị âm, phân phối chuẩn là 0 và độ lệch chuẩn là 1. Dữ liệu sẽ xoay quanh phân phối chuẩn.

- Còn theo phương pháp MinMaxScaler thì sẽ không có giá trị âm, nằm trong khoảng từ [0;1], dữ liệu không cần theo phân phối chuẩn, phù hợp với mạng Nơron như LSTM, đặc biệt với dữ liệu về doanh thu.

Kết luận: MinMaxScaler sẽ là lựa chọn tốt hơn trong các bài toán Time Series với dữ liệu doanh thu, đặc biệt là khi sử dụng các mô hình học sâu vì chúng yêu cầu dữ liệu phải nằm trong một khoảng nhất định để giúp các mạng nơ-ron hoạt động hiệu quả hơn. Vậy dữ liệu của phương pháp này sẽ làm đầu vào cho bài toán.

Thực hiện mục tiêu 4:

1. Sau khi dữ liệu được chuẩn hóa theo phương pháp MinMaxScaler, lựa chọn mô hình LSTM để thực hiện dự đoán. Thực hiện chia bộ dữ liệu theo kiểu, lấy tất cả các cột để huấn luyện và dự đoán tất cả các cột.

```
# Tạo tập dữ liệu với nhiều biến đầu vào
def create_dataset(data, time_step=1):
    X, y = [], []
    for i in range(len(data) - time_step):
        X.append(data[i:(i + time_step), :]) # Lấy tất cả các cột
        y.append(data[i + time_step, :]) # Dự đoán tất cả các cột
    return np.array(X), np.array(y)
```

Hình 2.35: Chi bộ dữ liệu

Chia số bước thời gian.

```
# Số bước thời gian của chuỗi
time_step = 10 # Có thể điều chỉnh nếu cần
X, y = create_dataset(data, time_step)
X = X.reshape(X.shape[0], X.shape[1], X.shape[2]) # Giữ nguyên số lượng cột
```

Hình 2.36: Chi bước thời gian.

Tạo mô hình và chia dữ liệu.

```
# Tạo mô hình LSTM
class LSTMModel(nn.Module):
    def __init__(self, input_size, output_size, hidden_size=50, num_layers=1):
        super(LSTMModel, self).__init__()
        self.lstm = nn.LSTM(input_size, hidden_size, num_layers, batch_first=True)
        self.fc = nn.Linear(hidden_size, output_size) # Số cột output có thể khác với số cột input

    def forward(self, x):
        out, _ = self.lstm(x)
        out = self.fc(out[:, -1, :]) # Lấy đầu ra của bước cuối cùng
        return out

# Khởi tạo mô hình, loss và optimizer
input_size = X.shape[2] # Số lượng cột đầu vào
output_size = X.shape[2] # Dự đoán tất cả các cột
model = LSTMModel(input_size=input_size, output_size=output_size)
criterion = nn.MSELoss()
optimizer = optim.Adam(model.parameters(), lr=0.001)

# Chia dữ liệu thành tập huấn luyện và kiểm tra
train_size = int(len(X) * 0.8)
X_train, X_test = X[:train_size], X[train_size:]
y_train, y_test = y[:train_size], y[train_size:]
```

Hình 2.37: Tạo mô hình

Huấn luyện mô hình và dự đoán cho tập kiểm tra.

```
# Huấn luyện mô hình
num_epochs = 100
for epoch in range(num_epochs):
    model.train()
    optimizer.zero_grad()
    outputs = model(torch.FloatTensor(X_train))
    loss = criterion(outputs, torch.FloatTensor(y_train))
    loss.backward()
    optimizer.step()
    if (epoch + 1) % 10 == 0:
        print(f'Epoch [{epoch + 1}/{num_epochs}], Loss: {loss.item():.4f}')

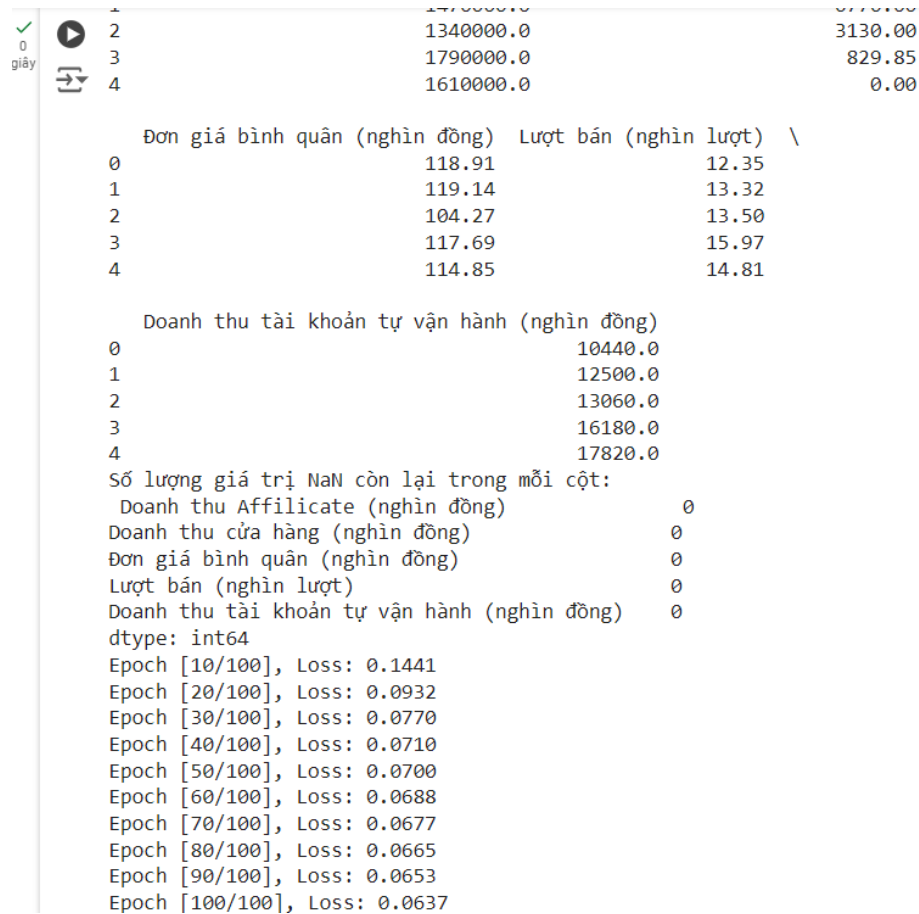
# Dự đoán cho tập kiểm tra
model.eval()
test_predict = model(torch.FloatTensor(X_test)).detach().numpy()
```

Hình 2.38: Huấn luyện mô hình và dự đoán

Tham số mất mát cho việc dự đoán trên dữ liệu kiểm tra.

Kết luận : Tham số mất mát giảm dần theo số vòng epoch, mô hình đang được huấn luyện đúng.

Dự đoán doanh thu trong 3 ngày tới.



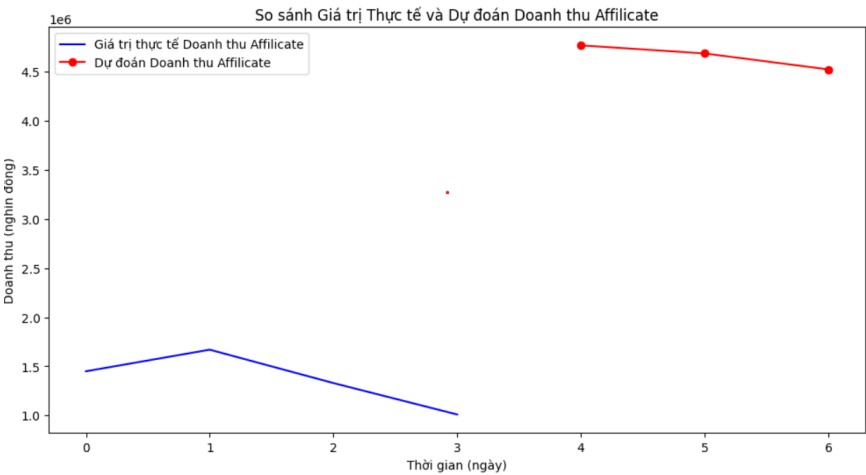
Hình 2.39: Huấn luyện mô hình và dự đoán

```
# Dự đoán 3 ngày tiếp theo (dựa trên dữ liệu cuối cùng trong tập test)
input_data = X_test[-1].reshape(1, time_step, input_size) # Dữ liệu cuối cùng trong tập test
predictions = []
for _ in range(3): # Dự đoán cho 3 ngày tiếp theo
    pred = model(torch.FloatTensor(input_data)).detach().numpy()
    predictions.append(pred)
    # Cập nhật input_data với dự đoán mới
    input_data = np.append(input_data[:, 1:, :], pred.reshape(1, 1, input_size), axis=1)

# Chuyển predictions thành numpy array và ngược lại thang đo gốc
predictions = np.array(predictions).reshape(3, output_size) # 3 ngày, số lượng cột bằng với output
predictions_original_scale = scaler.inverse_transform(predictions)

# In dự đoán đã được chuyển đổi về thang đo gốc
print("Dự đoán cho 3 ngày tiếp theo (giá trị gốc): ", predictions_original_scale)
```

Hình 2.40: Dự đoán 3 ngày



Hình 2.41: Biểu đồ

2.5 Tuần 7: Thử nghiệm trên một số mô hình (26/09 - 04/10)

Chủ đề tìm hiểu tuần 7: Thử nghiệm trên một số mô hình

1. Thử nghiệm việc dự đoán doanh thu trên mô hình Prophet của Facebook

Prophet là một mô hình dự báo chuỗi thời gian do Facebook phát triển, được thiết kế để xử lý dữ liệu có xu hướng, tính theo mùa và giá trị ngoại lệ. Mô hình này đặc biệt phù hợp với các ứng dụng thực tế khi dữ liệu có thể bị thiếu, xu hướng có thể thay đổi và có thể có các thành phần theo mùa quan trọng.

Mô hình Prophet được biết đến bởi các tính chất:

- **Xu hướng (Trend):** Dữ liệu có thể có xu hướng tăng hoặc giảm theo thời gian, và Prophet cho phép mô hình hóa xu hướng này dưới dạng tuyến tính hoặc phi tuyến (logistic growth model).
- **Mùa vụ (Seasonality):** Prophet xử lý tốt các thành phần mùa vụ (seasonal components), tức là các mô hình có chu kỳ lặp lại hàng năm, hàng tuần, hoặc hàng ngày. Các thành phần này có thể có biên độ thay đổi và không cố định.
- **Ngày đặc biệt (Holidays/Events):** Mô hình Prophet có khả năng xử lý các sự kiện đặc biệt như ngày lễ hoặc sự kiện ngoài dự tính, những ngày này có thể ảnh hưởng mạnh đến dữ liệu.
- **Outliers:** Prophet khá linh hoạt trong việc xử lý dữ liệu có nhiều hoặc điểm ngoại lai (outliers), giúp mô hình ít bị ảnh hưởng bởi các giá trị bất thường.
- **Dữ liệu không đều:** Prophet có thể xử lý chuỗi thời gian với khoảng cách thời gian không đều nhau giữa các điểm dữ liệu.

Dự đoán với mô hình Prophet:

```
import pandas as pd
from prophet import Prophet
import plotly.graph_objects as go

# Đường dẫn đến file CSV
file_path = '/content/Normalized_revenue_data.csv'

# Đọc file CSV
df = pd.read_csv(file_path)

# Chuyển đổi cột 'Ngày' thành kiểu datetime
df['Ngày'] = pd.to_datetime(df['Ngày'])

# Chuẩn bị dữ liệu cho Prophet - Ví dụ: dự đoán 'Doanh thu Affilicate'
prophet_data = df[['Ngày', 'Doanh thu Affilicate']].rename(columns={'Ngày': 'ds', 'Doanh thu Affilicate': 'y'})

# Khởi tạo và huấn luyện mô hình Prophet
model = Prophet()
model.fit(prophet_data)

# Dự đoán cho tương lai (thêm 3 ngày)
future = model.make_future_dataframe(periods=3)
forecast = model.predict(future)
```

Hình 2.42: Xây dựng mô hình

```
# In dự đoán
print(forecast[['ds', 'yhat', 'yhat_lower', 'yhat_upper']].tail(3))

# Vẽ biểu đồ của dự đoán và dữ liệu lịch sử bằng Plotly
fig = go.Figure()

# Thêm dữ liệu lịch sử
fig.add_trace(go.Scatter(x=prophet_data['ds'], y=prophet_data['y'],
                        mode='lines+markers', name='Doanh thu Affilicate (Actual)'))

# Thêm dự đoán (yhat)
fig.add_trace(go.Scatter(x=forecast['ds'], y=forecast['yhat'],
                        mode='lines', name='Dự đoán (Forecast)', line=dict(color='blue'))))

# Thêm dải tin cậy (confidence interval)
fig.add_trace(go.Scatter(x=forecast['ds'], y=forecast['yhat_upper'],
                        mode='lines', name='Upper Bound', line=dict(color='lightblue'),
                        fill=None))
fig.add_trace(go.Scatter(x=forecast['ds'], y=forecast['yhat_lower'],
                        mode='lines', name='Lower Bound', line=dict(color='lightblue'),
                        fill='tonexty'))

# Cập nhật tiêu đề và nhãn trục
fig.update_layout(title='Dự đoán Doanh thu Affilicate sử dụng Prophet',
                  xaxis_title='Ngày',
                  yaxis_title='Doanh thu Affilicate',
                  showlegend=True)

# Hiển thị biểu đồ
fig.show()
```

Hình 2.43: Dự đoán

Kết quả dự đoán:

ds	yhat	yhat_lower	yhat_upper
2024-09-08	0.674152	0.350259	0.978669
2024-09-09	0.695935	0.373381	1.004961
2024-09-10	0.263465	-0.041800	0.616759

Bảng 2.1: Forecasted Values from Prophet Model

Kết luận: Mô hình Prophet hiện tại, với bộ dữ liệu doanh thu có tính chất khá đơn điệu, chủ yếu dựa vào mùa vụ như các yếu tố ngày, tháng, năm, và các sự kiện lễ hội. Điều này thể hiện rõ qua khả năng của Prophet trong việc xử lý các thành phần xu hướng và mùa vụ theo thời gian.

2.6 Tuần 8: Thử nghiệm mô hình và thực hiện nội suy trọng số(05/10 - 11/10)

Chủ đề tìm hiểu tuần 8: Thử nghiệm mô hình trên Orange Data Mining và thực hiện nội suy trọng số

Mục tiêu của tuần:

1. Thử nghiệm thêm mô hình bằng Orange Data Mining và so sánh, đưa ra nhận xét.
2. Nội suy tuyến tính ra trọng số từ bảng Ranking nhà sáng tạo.

Thực hiện mục tiêu 1:

1. Thử nghiệm việc dự đoán doanh thu trên mô hình ARIMA và VAR trong Orange Data Mining

(a) Cơ sở lý thuyết

i. Mô hình ARIMA

- ARIMA (AutoRegressive Integrated Moving Average) là một trong những phương pháp phổ biến nhất được sử dụng trong phân tích và dự báo chuỗi thời gian đơn biến. ARIMA kết hợp 3 thành phần AR, I và MA:

- AR: là tự hồi quy, dùng để mô hình hoá mối quan hệ giữa giá trị hiện tại của chuỗi thời gian và các giá trị trước đó. Công thức đặc trưng:

$$\mathcal{X}_t = c + \Phi_1 \mathcal{X}_{t-1} + \Phi_2 \mathcal{X}_{t-2} + \dots + \Phi_p \mathcal{X}_{t-p} + \epsilon_t$$

với:

\mathcal{X}_t : Là giá trị tại thời điểm t

c : Là hằng số

Φ_i : Là các hệ số AR

ϵ_t : Là nhiễu trắng

- I: là tích hợp, liên quan đến lấy sự khác biệt của chuỗi thời gian để khiến nó ổn định, được đặc trưng bởi tham số d là số lần khác biệt cần thực hiện.
- MA: là trung bình động, dùng để mô hình hoá mối quan hệ giữa giá trị hiện tại và các lỗi đã dự báo trước đó. Công thức tính:

$$\mathcal{X}_t = c + \epsilon_t + \Theta_1 \epsilon_{t-1} + \dots + \Theta_q \epsilon_{t-q}$$

với Θ_i là các hệ số MA.

- Quy trình thực hiện mô hình ARIMA

- Xác định tính ổn định bằng cách kiểm tra sự ổn định của các tham số thống kê như trung bình và phương sai.
- Ổn định hoá chuỗi thời gian.
- Xác định các tham số p (trong AR), d (trong I), q (trong MA).
- Xây dựng mô hình, kiểm tra và đánh giá.

ii. Mô hình VAR

- VAR (Vector AutoRegression) là mô hình thống kê được sử dụng để mô hình hóa và dự báo các hệ thống biến thời gian đa biến. Cấu trúc của mô hình VAR được đặc trưng qua công thức:

$$\mathcal{Y}_t = c + A_1\mathcal{Y}_{t-1} + \dots + A_p\mathcal{Y}_{t-p} + \epsilon_t$$

với

\mathcal{Y}_t : Là vector các biến tại thời điểm t

c : Là vector hằng số

A_i : Là ma trận hệ số độ trễ i

ϵ_t : Là vector nhiễu ngẫu nhiên

- Quy trình thực hiện mô hình VAR:
 - Xác định các biến.
 - Kiểm tra tính dừng của chuỗi thời gian.
 - Xác định độ trễ tối ưu.
 - Ước lượng mô hình VAR qua công thức đặc trưng.
 - Xây dựng mô hình, kiểm tra và đánh giá.

(b) Thử nghiệm trên Orange Data Mining

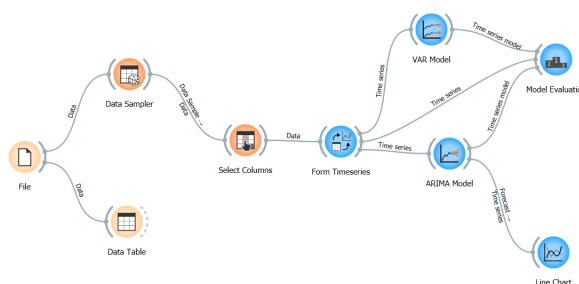
- Dữ liệu doanh thu được chuẩn bị gồm 3 cột doanh thu gồm doanh thu affiliate, doanh thu cửa hàng, doanh thu tài khoản tự vận hành.

Ngày	Doanh thu Affiliate (nguyên đồng)	Doanh thu cửa hàng (nguyên đồng)	Doanh thu tài khoản tự vận hành (nguyên đồng)
8/7/2024	1470000	1940	118.91
8/12/2024	1470000	6770	119.14
8/17/2024	1490000	3130	104.27
8/18/2024	1700000	829.85	117.89
8/24/2024	1610000	0	114.85
8/31/2024	1780000	1170000	114.68
9/10/2024	1860000	350.68	88.95
9/17/2024	1510000	2740	98.19
9/18/2024	1090000	277.33	85.92

Hình 2.44: Dữ liệu doanh thu

- Mục tiêu là sử dụng mô hình TimeSeries để phân tích, đánh giá xu hướng của dữ liệu Doanh thu trong chuỗi thời gian của tập dữ liệu. Ta sẽ thử từng thuộc tính và thống kê các thông số đánh giá mô hình chuỗi thời gian gồm:

- RMSE: là căn bậc 2 của giá trị trung bình của bình phương sai số giữa các giá trị dự đoán và giá trị thực tế. RMSE cho biết mức độ sai lệch của các dự đoán với thực tế (giá trị thấp hơn → mô hình dự đoán chính xác hơn).
 - MAE: là giá trị trung bình của các sai số tuyệt đối giữa giá trị thực tế và giá trị dự đoán. MAE cho biết sai số trung bình mà mô hình mắc phải trong các dự đoán.
 - MAPE: giá trị trung bình của sai số tuyệt đối giữa giá trị thực tế và giá trị dự đoán, được tính theo phần trăm. MAPE giúp đánh giá độ chính xác của mô hình dự đoán theo phần trăm.
 - POCID: là một chỉ số thể hiện điểm tại đó hai mô hình dự đoán cho kết quả tương đương. POCID thường được sử dụng trong các mô hình đối sánh và giúp xác định mô hình nào có hiệu suất tốt hơn ở mức độ nhất định.
 - R^2 : là hệ số xác định, đo lường tỷ lệ phương sai của biến phụ thuộc được giải thích bởi mô hình hồi quy. R^2 có giá trị từ 0 đến 1. Giá trị cao hơn cho thấy mô hình giải thích tốt hơn về sự biến động của dữ liệu.
 - AIC: là một tiêu chí thông tin được sử dụng để chọn mô hình. Nó đánh giá chất lượng mô hình bằng cách tính toán độ phù hợp và độ phức tạp của mô hình. Giá trị AIC thấp hơn cho thấy mô hình tốt hơn.
 - BIC: tương tự như AIC nhưng có hình phạt mạnh hơn đối với số lượng tham số. Giá trị BIC thấp hơn cho thấy mô hình tốt hơn, và BIC thường được ưa chuộng hơn AIC khi kích thước mẫu lớn.
- Sơ đồ thực hiện kiểm tra mô hình như sau: Trong đó:



Hình 2.45: Sơ đồ thực hiện

- File dữ liệu không nhất thiết phải được chuẩn hoá. Dữ liệu được đưa đến Select Column để chọn 1 mục Target là một cột doanh thu bất kì, còn mục Feature chỉ có thuộc tính ngày.
- Form Timeseries: chọn thuộc tính ngày làm chuỗi thời gian, dữ liệu target được phân bổ theo chuỗi thời gian đó.
- VAR Model và ARIMA Model: 2 mô hình Timeseries có trong Orange Data Mining, cùng nối với Model Evaluation để đánh giá và so sánh.

(c) Phân tích kết quả Kết quả thu được sau khi thực hiện:

Target	Model	RMSE	MAE	MAPE	POCID	R ²	AIC	BIC
Doanh thu Affiliate	ARIMA(1,1,0)	4011593,27	473314	0,868	44,4	-0,521	1542,9	1546,6
	VAR(1,n)	3603258,72	1582211	0,939	44,4	-0,227	41,5	41,8
Doanh thu Cửa hàng	ARIMA(1,1,0)	480150,09	4296,7	1,963	33,3	-0,675	1555,1	1558,8
	VAR(1,n)	390295,44	132031	1,956	55,6	-0,107	41,8	42,2
Doanh thu đơn giá Tự vận hành	ARIMA(1,1,0)	10736	8541,4	0,341	44,4	-0,827	1028,8	1032,6
	VAR(1,n)	10477	7013,7	0,314	44,4	-0,74	30,7	31,1

Hình 2.46: Kết quả thu được

(d) Nhận xét, đánh giá dựa trên các thông số:

- RMSE: VAR có giá trị RMSE thấp hơn ARIMA trong cả 3 trường hợp, điều đó cho thấy tỉ lệ dự báo chính xác của VAR cao hơn (gần thực tế hơn).
- MAE: tương tự RMSE. Tuy nhiên ở target "Doanh thu Affiliate", mặc dù VAR có RMSE thấp hơn, MAE của nó cao hơn đáng kể so với ARIMA, điều này có thể cho thấy VAR có thể dự đoán tốt hơn trong tổng thể nhưng vẫn tồn tại sai số lớn ở một số điểm dữ liệu cụ thể.
- MAPE: Với các target "Doanh thu Cửa hàng", "Doanh thu đơn giá tự vận hành", ARIMA có MAPE cao hơn, cho thấy tỷ lệ sai số dự báo tương đối cao hơn so với VAR.
- POCID: Tỷ lệ phần trăm dự báo đúng hướng dao động tương đương giữa hai mô hình trong hầu hết các trường hợp, trừ target "Doanh thu Cửa hàng", trong đó VAR có POCID cao hơn (55.6 so với 33.3), cho thấy mô hình VAR dự đoán xu hướng tốt hơn.
- AIC và BIC: AIC và BIC đều chỉ ra mô hình VAR đơn giản hơn và có độ khớp tốt hơn so với ARIMA cho hầu hết các target.
- Hệ số R^2 mang giá trị âm, nên việc sử dụng 2 mô hình này không quá phù hợp với bài toán. Tuy nhiên hệ số R^2 của VAR gần 0 hơn, cho thấy nó vẫn chính xác hơn.

⇒ Như vậy, mô hình VAR cho kết quả tốt hơn cho việc dự đoán doanh thu trong bài toán.

Thực hiện mục tiêu 2: Nội suy trọng số

1. Cơ sở lý thuyết

Bảng dưới đây mô tả các thuộc tính của nhà sáng tạo:

Rank	Nhà sáng tạo	Số người theo dõi(nghìn người)	Doanh thu(nghìn đồng)	Lượt bán(nghìn lượt)	Doanh thu từ video(nghìn đồng)	Doanh thu Live(nghìn đồng)
1	@tiemtaphoa_so37	6	2500000	27,35	2500000	0
2	@haithichreview	2,89	1450000	15,86	1450000	0
3	@nhunhoi22	49,53	1020000	11,26	1020000	0
4	@thaivublackbi149	3020	1010000	11,01	1010000	0
5	@cocomanshop	2,6	905850	9,87	905850	0
6	@ig_yangg.22	347,14	867040	9,44	867040	0
7	@chuyen.thang.thar	145,62	864160	9,36	864160	0
8	@chucareviewkhong	734,88	759150	8,29	759150	0
9	@annie.riviu	31,48	660960	7,3	657080	3880
10	@vietcgreview	106,15	647460	7	647460	0

Hình 2.47: Bảng dữ liệu nhà sáng tạo

Ta coi như mỗi thuộc tính (trừ Rank) là cột feature từ X_1 đến X_n .

Mục tiêu của ta là từ các cột còn lại, có thể suy ra được Rank. Đúng hơn là khi có n điểm dữ liệu, ta có thể xếp hạng cho nó. Để làm được điều ấy, ta phải suy ra được hệ số cụ thể với mỗi thuộc tính, bởi không phải thuộc tính

nào cũng có hệ số giống nhau.

Giả sử các cột X_1, X_2, \dots, X_n có các hệ số là w_1, w_2, \dots, w_n . Đương nhiên, những hệ số này càng tốt thì Rank sẽ càng cao, cho nên W sẽ luôn dương. Trừ cột *id* nhà sáng tạo ra, bởi vì nó không ảnh hưởng đến rank ($W_1 = 0$).

X_0 chính là cột mục tiêu (Target) mà ta muốn cuối cùng.

Với một điểm dữ liệu mới với bộ giá trị thuộc tính là (x_1, x_2, \dots, x_n) , ta sẽ tìm ra được trọng số cuối cùng của nó là:

$$W = x_1 \cdot w_1 + x_2 \cdot w_2 + \dots + x_n \cdot w_n$$

Công việc của ta là tìm ra các hệ số w_1, \dots, w_n này để đưa ra trọng số cuối cùng cho các điểm dữ liệu, từ đó xếp hạng cho chúng.

2. Ý tưởng ban đầu

Số Rank càng nhỏ thì trọng số càng cao, có nghĩa nếu đảo ngược số Rank lại, ta sẽ được trọng số cuối cùng cho các điểm dữ liệu. Với x_1, x_2, \dots, x_n là các thuộc tính (Feature), ta sẽ có Rank là mục tiêu (target).

Từ đó, ta đưa về bài toán hồi quy tuyến tính (Linear Regression). Việc xây mô hình sẽ từ bảng Rank để suy ra các hệ số của thuộc tính, rồi lại test thử với một vài điểm dữ liệu xem rank có đúng như vậy không.

```
else:
    # Nếu không có cột nào thiếu, tiếp tục quá trình
    X = df[X_columns]
    y = df['Rank']

    # Bước 5: Chia dữ liệu thành tập huấn luyện và kiểm tra
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

    # Bước 6: Khởi tạo và huấn luyện mô hình hồi quy tuyến tính
    model = LinearRegression()
    model.fit(X_train, y_train)

    # Bước 7: In ra trọng số của các thuộc tính
    print("Trọng số của các thuộc tính:")
    for feature, coef in zip(X.columns, model.coef_):
        print(f"{feature}: {coef:.4f}")

    # Bước 8: Đánh giá mô hình
    score = model.score(X_test, y_test)
    print(f"Độ chính xác của mô hình (R^2 score): {score:.4f}")
```

Hình 2.48: Code tìm trọng số

2.7 Tuần 9: Thử nghiệm mô hình Neural Network trên Orange Data Mining và chuyển đổi mô hình dữ liệu thành đồ thị(10/10 - 17/10)

Chủ đề tìm hiểu tuần 9: Thử nghiệm mô hình Neural Network trên Orange Data Mining và chuyển đổi mô hình dữ liệu thành đồ thị.

Mục tiêu của tuần 9:

1. Thử nghiệm mô hình Neural Network trên Orange Data Mining và đưa ra nhận xét.
2. Chuyển đổi mô hình dữ liệu thành mô hình đồ thị.

Thực hiện mục tiêu 1:

1. Cơ sở lý thuyết:

Neural Network bao gồm các lớp neurons (nút) được kết nối với nhau. Mỗi neuron nhận các đầu vào, tính toán và chuyển tiếp đầu ra thông qua hàm kích hoạt (activation function).

- **Input Layer:** Là nơi nhận các đặc trưng (features) từ dữ liệu đầu vào. Số lượng neurons ở lớp này tương ứng với số lượng đặc trưng.
- **Hidden Layers:** Đây là các lớp trung gian giữa input layer và output layer. Mỗi neuron trong hidden layers thực hiện phép nhân có trọng số (weights) với đầu vào, cộng thêm bias, sau đó áp dụng một hàm kích hoạt để tạo ra output. Việc tăng số lượng hidden layers và neurons giúp mô hình học được các mẫu phức tạp hơn, nhưng cũng tăng nguy cơ overfitting.
- **Output Layer:** Đây là lớp cuối cùng của mạng, cung cấp dự đoán hoặc phân loại dựa trên bài toán cụ thể.

Các hàm kích hoạt:

- **Identity:** Là hàm kích hoạt đơn giản trong mạng Neural, trong đó đầu ra của neuron bằng với đầu vào của nó.
- **ReLU (Rectified Linear Unit):** Là hàm phổ biến, chuyển tất cả các giá trị âm thành 0 và giữ nguyên các giá trị dương.
- **Sigmoid:** Là hàm giúp biến đổi đầu ra thành giá trị trong khoảng từ 0 đến 1, phù hợp cho các bài toán nhị phân.
- **Tanh:** Là hàm kích hoạt tương tự Sigmoid nhưng chuyển đổi đầu ra trong khoảng từ -1 đến 1.

Thuật toán tối ưu:

- **Adam (Adaptive Moment Estimation):** Một trong những thuật toán tối ưu phổ biến nhất cho Neural Networks. Adam tự động điều chỉnh tốc độ học cho từng thông số dựa trên các biến động của gradient.
- **SGD (Stochastic Gradient Descent):** Là thuật toán truyền thống, điều chỉnh trọng số từng bước một dựa trên mỗi điểm dữ liệu hoặc từng batch nhỏ.
- **L-BFGS-B:** Là một thuật toán tối ưu hóa số học sử dụng cho các bài toán tối ưu không bị ràng buộc hoặc có các ràng buộc về miền giá trị của biến số (ví dụ: giá trị của biến bị giới hạn trong một khoảng nào đó).

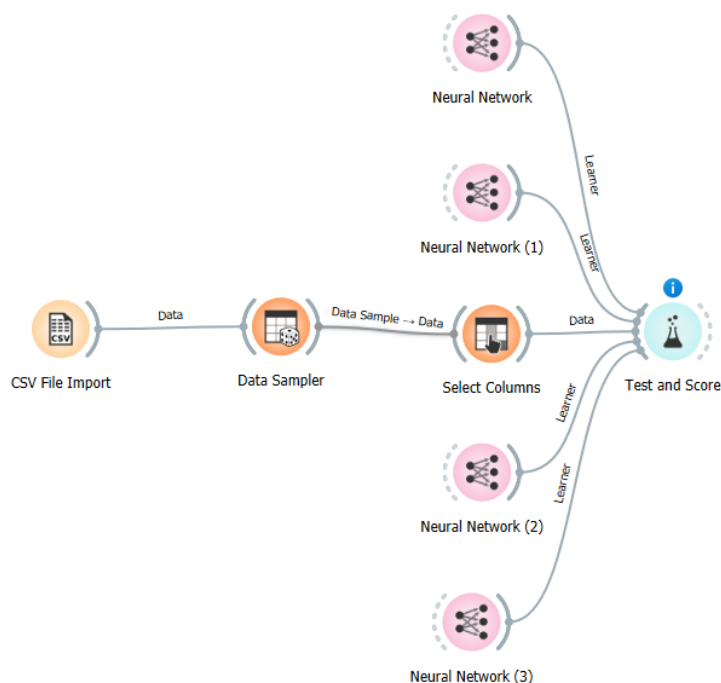
Regularization: Trọng số alpha trong đó có ý nghĩa là trọng số phạt của mô hình lên các thuộc tính nhiễu. Nếu nó quá nhỏ, mô hình dễ học những thuộc tính nhiễu này và dẫn đến overfitting. Nếu nó quá lớn, mô hình dễ loại trừ đi nhiễu thuộc tính và dẫn đến underfitting. Cho nên nên chọn thông số nào ở mức vừa phải, trọng số này phải thay đổi cụ thể trong mô hình.

Max Iterations: Số vòng lặp mà mô hình thực hiện để cập nhật trọng số qua các sai số để đưa đến cái kết quả chính xác hơn. Trọng số này là trọng số kết nối giữa các điểm neuron qua các lớp (các neuron qua các trọng số này, đưa vào hàm kích hoạt để cho ra được neuron mới).

Cross-validation: Kỹ thuật chia dữ liệu để tránh bị overfitting cho mô hình (nằm trong phần test and scores). Ví dụ 1000 điểm dữ liệu chia thành 5 folds (k), mỗi fold có 200 điểm dữ liệu, mô hình sẽ lấy (k-1=4) folds = 800 điểm dữ liệu để train, 1 fold còn lại là 200 để test. Quá trình cứ lặp lại như vậy, sẽ được k lần lấy (k bộ dữ liệu), mỗi bộ chứa k-1 phần để train và 1 phần để test.

2. Quy trình thực hiện mô hình Neural Network:

- Xác định các feature và target để xây dựng.
- Xác định số lượng layer, số lượng neuron cho mỗi hidden layer.
- Lựa chọn hàm kích hoạt, thuật toán tối ưu, L2 Regularization, số vòng lặp tối đa và xác định số k để thực hiện cross-validation.
- Xây dựng mô hình, kiểm tra và đánh giá.



Hình 2.49: Sơ đồ thực hiện các mô hình Neural Network

3. Thử nghiệm trên Orange Data Mining:

Dữ liệu đưa vào gồm các features (Ngày, Đơn giá bình quân, Lượt bán) để dự đoán target cho Doanh thu Affilicate.

Mục tiêu là so sánh giữa các loại mô hình khác nhau để từ đó đưa ra một mô hình tối ưu nhất. Ta sẽ xây dựng mô hình và so sánh dựa trên các thuộc tính sau:

- **MSE (Mean Squared Error):** Trung bình của bình phương của sự chênh lệch giữa giá trị dự đoán và giá trị thực tế. Giá trị MSE càng nhỏ, mô hình càng chính xác.
- **RMSE (Root Mean Squared Error):** Căn bậc hai của MSE, cho giá trị trung bình của sự chênh lệch giữa giá trị dự đoán và giá trị thực tế.
- **MAE (Mean Absolute Error):** Đo sự chênh lệch tuyệt đối trung bình giữa giá trị thực và dự đoán. MAE nhỏ chỉ ra rằng mô hình ít sai số hơn.
- **R-squared:** Phần trăm phương sai của biến phụ thuộc mà mô hình có thể giải thích được. Giá trị càng gần 1 càng tốt.

(a) *Giữ nguyên cách chọn feature và target, thay đổi thông số của mô hình*

Ta sẽ thiết lập thông số và so sánh giữa các mô hình với đầu vào là thuộc tính Ngày để dự đoán đầu ra Doanh thu Affilicate.

i. *Chỉ thay đổi mỗi cách chọn các hidden layer của mô hình:* Các mô hình sử dụng cùng hàm kích hoạt Logistic, thuật toán tối ưu Adam, trọng số Regularization 0.001, số vòng lặp tối đa 100.

Số lượng neuron cho mỗi hidden layer (mỗi layer là một số)	MSE	RMSE	MAE	R^2
5	0.053	0.231	0.174	-0.086
5, 10, 20	0.051	0.225	0.153	-0.032
10, 20, 30, 100	0.049	0.222	0.115	-0.003
20, 40, 60, 80, 100	0.060	0.246	0.115	-0.231

Bảng 2.2: So sánh các mô hình theo số lượng neuron cho mỗi hidden layer

Mô hình có cách chọn layer (10, 20, 30, 100) có hiệu quả tốt nhất về các chỉ số MSE, RMSE và MAE, cho thấy khả năng dự đoán Doanh thu Affilicate tương đối ổn định hơn so với các mô hình khác.

MAPE vẫn rất cao ở tất cả các mô hình, cho thấy độ sai lệch trong dự đoán còn lớn.

ii. *Chỉ thay đổi cách chọn các hàm kích hoạt:* Các mô hình sử dụng cùng cách chọn layer (10, 20, 30, 100), thuật toán tối ưu Adam, trọng số Regularization 0.001, số vòng lặp tối đa 100.

Hàm kích hoạt	MSE	RMSE	MAE	R^2
Identity	4311540177850352.5	65662319.315192886	65662295.45065042	-8.773737661217957e+16
Logistic	0.049	0.222	0.115	-0.003
tanh	0.058	0.242	0.201	-0.190
ReLu	137186538125.5623	370387.011	370386.868	-2791667586324.205

Bảng 2.3: So sánh các mô hình theo hàm kích hoạt

Mô hình sử dụng hàm kích hoạt Logistic có hiệu quả tốt nhất trong tất cả các hàm kích hoạt, với MSE, RMSE và MAE thấp nhất.

MAPE vẫn cao, cho thấy khả năng dự đoán không ổn định.

Identity và ReLU đều cho kết quả rất kém, với các chỉ số sai số rất cao, đặc biệt là R2 âm lớn, cho thấy chúng không phù hợp với bài toán dự đoán này.

iii. *Chỉ thay đổi cách chọn các thuật toán tối ưu:* Các mô hình sử dụng cùng cách chọn layer (10, 20, 30, 100), hàm kích hoạt Logistic, trọng số Regularization 0.001, số vòng lặp tối đa 100.

Thuật toán tối ưu	MSE	RMSE	MAE	R^2
L-BFGS-B	0.049	0.222	0.123	-0.002
SGD	0.055	0.235	0.184	-0.122
Adam	0.049	0.222	0.115	-0.003

Bảng 2.4: So sánh các mô hình theo thuật toán tối ưu

Mô hình sử dụng thuật toán Adam tối ưu nhất trong ba thuật toán, với các chỉ số MSE, RMSE, MAE thấp nhất, cho thấy khả năng học và dự đoán tốt nhất.

Mô hình sử dụng thuật toán L-BFGS-B cũng cho kết quả gần tương đương với Adam, nhưng MAE và MAPE hơi cao hơn một chút.

Mô hình sử dụng thuật toán SGD có hiệu suất kém hơn hẳn, với các chỉ số sai số cao hơn và R2 âm lớn hơn.

iv. *Chỉ thay đổi trọng số phạt (regularization):* Các mô hình sử dụng cùng cách chọn layer (10, 20, 30, 100), hàm kích hoạt Logistic, thuật toán tối ưu Adam, số vòng lặp tối đa 100.

Regularization	MSE	RMSE	MAE	R^2
0.0001	0.049	0.222	0.115	-0.003
0.001	0.049	0.222	0.115	-0.003
0.01	0.049	0.222	0.115	-0.003
0.1	0.049	0.222	0.123	-0.002
1	0.049	0.222	0.123	-0.002
10	0.049	0.222	30.123	-0.002

Bảng 2.5: So sánh các mô hình theo Regularization

Với việc thay đổi các giá trị Regularization từ 0.0001 đến 10, các chỉ số của mô hình hầu như không thay đổi đáng kể. Điều này cho thấy các mô hình có khả năng ổn định và không quá nhạy cảm với mức độ regularization trong khoảng này. Trong trường hợp này, regularization từ 0.0001 đến 0.01 có thể là lựa chọn phù hợp nhất, vì nó cho các chỉ số sai số thấp và độ chính xác cao hơn.

v. *Chỉ thay đổi cách chia dữ liệu với cross validation:* Các mô hình sử dụng cùng cách chọn layer (10, 20, 30, 100), hàm kích hoạt Logistic, thuật toán tối ưu Adam, trọng số Regularization 0.001, số vòng lặp tối đa 100.

Number of folds (Cross validation)	MSE	RMSE	MAE	R^2
2	0.049	0.222	0.120	-0.007
3	0.050	0.223	0.121	-0.016
5	0.049	0.222	0.115	-0.003
10	0.050	0.224	0.119	-0.021
20	0.050	0.223	0.117	-0.016

Bảng 2.6: So sánh các mô hình theo số fold trong Cross validation

Số fold nhỏ (2, 5): Mô hình với số fold 2 và 5 có hiệu suất tốt hơn với các chỉ số sai số (MSE, RMSE, MAE, MAPE) thấp hơn, đặc biệt số fold 5 cho kết quả tối ưu nhất về MAE và R2, cho

thấy độ lệch trung bình thấp và khả năng dự đoán tốt hơn.

Số fold lớn (10, 20): Khi số fold tăng lên 10 và 20, các chỉ số sai số như MAE và MAPE tăng lên nhẹ, điều này cho thấy rằng mô hình có thể gặp khó khăn trong việc duy trì độ chính xác khi số fold quá cao. Việc chia nhỏ dữ liệu có thể gây ra sự không đồng đều trong quá trình huấn luyện và đánh giá mô hình.

vi. *Kết luận về các thông số tốt nhất cho mô hình*

Tóm lại, cấu hình đề xuất cho mô hình dự đoán là:

- Hàm kích hoạt: Logistic
- Thuật toán tối ưu: Adam
- Regularization: 0.001
- Cross-validation folds: 5
- Hidden layers: (10, 20, 30, 100)

Thiết lập này sẽ giúp đạt hiệu suất cao nhất với các chỉ số lỗi MSE, RMSE, MAE thấp, đảm bảo mô hình có độ chính xác cao khi dự đoán doanh thu từ dữ liệu ngày. Tuy nhiên, chỉ số R2 âm cho thấy mô hình này vẫn chưa phù hợp với dữ liệu và cần tối ưu thêm.

- (b) *Giữ nguyên các thông số của mô hình, thay đổi cách chọn các feature* Với cấu hình đề xuất cho mô hình như trên, ta sẽ xây dựng cho mô hình với các feature khác nhau (luôn giữ nguyên cột ngày làm đặc trưng) để dự đoán Doanh thu Affiliate.

Features	MSE	RMSE	MAE	R^2
Ngày	0.049	0.222	0.115	-0.003
Ngày + Lượt bán	0.053	0.229	0.116	-0.071
Ngày + Đơn giá bình quân	0.053	0.229	0.116	-0.071
Ngày + Đơn giá bình quân + Lượt bán	0.051	0.225	0.153	-0.071
Ngày + Đơn giá bình quân + Doanh thu cửa hàng	0.051	0.225	0.153	-0.031
Ngày + Lượt bán + Doanh thu cửa hàng	0.051	0.225	0.153	-0.031
Ngày + Lượt bán + Doanh thu cửa hàng tự vận hành	0.051	0.225	0.153	-0.031

Bảng 2.7: So sánh các mô hình theo việc lựa chọn các thuộc tính làm features

Feature "Ngày" khi sử dụng đơn lẻ, cho thấy mô hình hoạt động khá ổn định với lỗi nhỏ nhất và giá trị R2 cao nhất trong các mô hình (tuy nhiên vẫn là âm).

Thêm các feature bổ sung như "Lượt bán", "Đơn giá bình quân", "Doanh thu cửa hàng" hoặc "Doanh thu cửa hàng tự vận hành" không cải thiện đáng kể hiệu quả mô hình. Hầu hết các chỉ số lỗi đều tăng nhẹ hoặc không thay đổi, và giá trị R2 vẫn ở mức âm.

Kết luận chung: Cấu hình tốt nhất cho bài toán này không cần phức tạp mà nên duy trì ở mức đơn giản, tập trung vào các tham số tối ưu như hàm kích hoạt Logistic và thuật toán tối ưu Adam. Không cần thiết phải thêm các feature khác vào mô hình hiện tại, vì chúng không mang lại giá trị dự báo bổ sung đáng kể cho Doanh thu Affiliate.

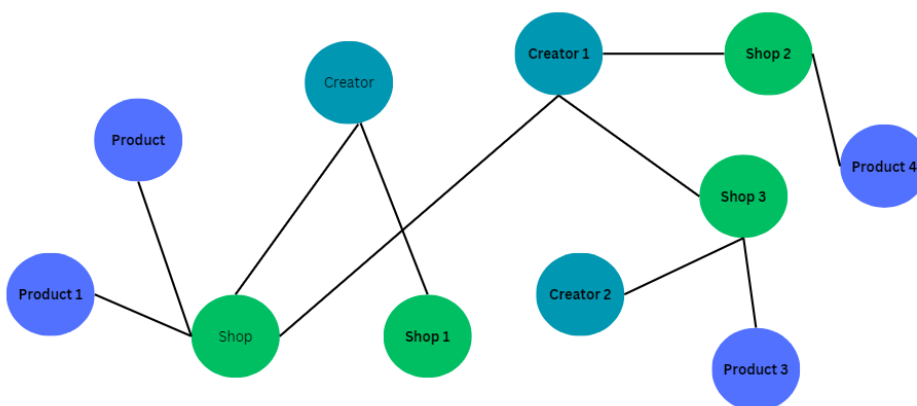
Thực hiện mục tiêu 2: Chuyển đổi mô hình dữ liệu thành mô hình đồ thị

1. Đặt vấn đề cho bài toán:

- Với bộ dữ liệu TimeSeries tuyến tính và đơn giản về các bảng đơn như Cửa hàng, nhà sáng tạo, sản phẩm. Việc sử dụng các mô hình phù hợp với bộ dữ liệu mùa vụ trên như Prophet, ARIMA là hoàn toàn phù hợp (nếu lượng dữ liệu đủ nhiều), hoặc thậm chí mới mô hình phức tạp thuộc Neural Network như LSTM vẫn hợp lý. Tuy nhiên bài toán này chỉ phù hợp với việc dự đoán trong tương lai gần, theo xu hướng mùa vụ về mặt thời gian, các sự kiện.
- Để phát triển bài toán sâu sắc hơn, học và dự đoán một cách toàn diện hơn, các bảng dữ liệu về Cửa hàng, Nhà sáng tạo, Sản phẩm phải được liên kết với nhau. Vấn đề khó khăn ở đây là bảng dữ liệu về Cửa hàng, sản phẩm là bảng dữ liệu về TimeSeries, mỗi dòng là doanh thu theo từng ngày ; bảng Nhà sáng tạo, sản phẩm thì mỗi dòng lại là tổng doanh thu trên 1 tháng, điều quan trọng là mỗi dòng trên bảng cửa hàng lại có thể chứa rất nhiều sản phẩm, nhà sáng tạo liên quan đến nó, vậy nên để liên kết quan hệ vào bảng chung theo kiểu 1-1 thì không thể. Bởi nếu có thể liên kết, cách biểu diễn mỗi quan hệ như trên cũng không đúng và dẫn đến nhiều thuộc tính nhiễu.
- Từ những vấn đề trên, nhóm quyết định phát triển mô hình dữ liệu thành mô hình đồ thị. Mục đích là để biểu diễn đúng đắn hơn các mối quan hệ của các bảng dữ liệu. Từ đó đưa vào mô hình Neural Network để dự đoán một cách chính xác.

2. Chuyển đổi mô hình dữ liệu thành mô hình đồ thị như sau:

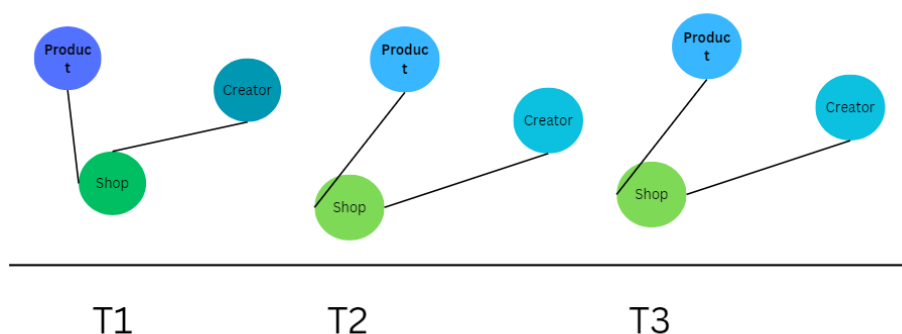
- Các node chia ra ba loại cửa hàng, nhà sáng tạo và sản phẩm. Đường nhiên mạng lưới đồ thị sẽ gồm nhiều node với mỗi loại như vậy như: Các cửa hàng, các sản phẩm khác nhau trong cùng một danh mục, và các nhà sáng tạo khác nhau.
- Quan hệ giữa các node: Cửa hàng chính là Shop, Shop sẽ có hợp đồng với nhiều nhà Creator để giới thiệu nhiều sản phẩm (Product) khác nhau của Shop. Các Product trong Shop thì sẽ có quan hệ với Creator thông qua Shop. Ngoài ra với những Shop khác, các Creator hợp đồng với Shop cũ cũng có thể hợp đồng với các Shop khác để giới thiệu những Product khác. Như vậy các Creator, Product, Shop khác nhau có thể có mối liên kết với nhau.



Hình 2.50: Mô hình tổng quan về đồ thị

3. Mô hình TimeSeries với đồ thị.

Mỗi ngày, đồ thị sẽ ở một điểm TimeSeries. Như vậy với bộ dữ liệu của nhiều ngày, ta sẽ được một bộ đồ thị ứng với số ngày đó.



Hình 2.51: Đồ thị theo TimeSeries.

4. Thiết kế bài toán.

- Bài toán 1: Dự đoán doanh thu của Shop hiện tại, mô hình đồ thị có thể dựa vào sự biến động theo thời gian của Shop hiện tại, của các sản phẩm (Product) và các nhà sáng tạo (Creator) và các Shop khác nhau, các Creator, Product khác nhau để đưa ra dự đoán chuẩn hơn cho doanh thu hiện tại của Shop.

- Bài toán 2: Mô hình đồ thị có thể đề xuất các sản phẩm, nhà sáng tạo tiềm năng mà trước đó chưa liên quan đến Shop để đưa ra chiến lược kinh doanh cho Shop rằng: Shop đây có thể thuê thêm nhà sáng tạo nào với sả