TRƯỜNG ĐẠI HỌC PHENIKAA

KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO ĐỒ ÁN CƠ SỞ

Đề tài: Tìm hiểu về bài toán Sales Prediction trong lĩnh vực tài chính.

Giảng viên: TS. Lương Văn Thiện

Học phần: Đồ án cơ sở

Sinh viên: Trần Minh Hiếu

MSSV: 21011601

Năm học: 2023-2024

MỤC LỤC

Contents

DANH SÁCH HÌNH ẢNH

MỞ ĐẦU

Trong thời đại số ngày nay, khi doanh nghiệp phải đối mặt với sự biến động nhanh chóng và đa dạng của thị trường, việc phân tích thị trường, dự đoán doanh số bán hàng, … là một trong những chìa khóa quan trọng để tồn tại và phát triển. Việc dự đoán doanh số không chỉ giúp doanh nghiệp dự báo xu hướng mua sắm và xu hướng thị trường mà còn là bước đầu tiên để hiểu rõ hơn về mối quan tâm và sở thích của khách hàng.

Chúng ta đã học được nhiều từ thế giới số, từ cách mà dữ liệu có thể trở thành nguồn lợi thế quyết định và làm nên sự thành công. Trong bài báo cáo này, em sẽ nghiên cứu năm chủ đề liên quan đến bài toán Sales Prediction trên Kaggle, mỗi chủ đề là một khám phá sâu sắc vào các khía cạnh khác nhau của thị trường và mô hình kinh doanh: từ mua sắm trực tuyến đến bất động sản, từ ngành thức ăn nhanh đến lĩnh vực bảo hiểm và cuối cùng, bán lẻ.

NỘI DUNG

Chương 1: Tổng quan về bài toán Sales Prediction

1.1 Đặt vấn đề

Trong một thị trường đa dạng và biến động, khả năng dự đoán doanh số bán hàng là chìa khóa để tối ưu hóa chiến lược kinh doanh và thích nghi với nhu cầu thị trường.

1.2 Mục tiêu và ý nghĩa

a) Mục tiêu

- Dự đoán doanh số bán hàng: Sử dụng dữ liệu lịch sử để dự đoán doanh số bán hàng trong tương lai, giúp doanh nghiệp nắm bắt được xu hướng và biến động của thị trường.
- Hiểu rõ hơn về khách hàng: Từ việc phân tích doanh số bán hàng, doanh nghiệp có thể hiểu rõ hơn về ưu thích và hành vi mua sắm của khách hàng, giúp tối ưu hóa chiến lược tiếp thị và quảng cáo.
- Tối ưu hóa chiến lược kinh doanh: Dự đoán doanh số bán hàng cung cấp thông tin chi tiết về cách thức mà các biến như sản phẩm, giá cả, và chương trình khuyến mãi ảnh hưởng đến doanh số, từ đó giúp doanh nghiệp điều chỉnh chiến lược kinh doanh của mình.

b) Ý nghĩa

Tăng cường quyết định chiến lược: Cung cấp cái nhìn chi tiết và đáng tin cậy về thị trường, từ đó doanh nghiệp có thể ra quyết định chiến lược một cách chính xác và linh hoạt.

Tối ưu hóa kho hàng và nguồn lực: Hiểu rõ về xu hướng mua sắm giúp tối ưu hóa quản lý kho hàng, dự trữ, và sử dụng nguồn lực hiệu quả hơn.

Nâng cao trải nghiệm khách hàng: Thông qua việc đánh giá hành vi mua sắm và phản hồi từ khách hàng, doanh nghiệp có thể cải thiện trải nghiệm mua sắm, tăng cường sự hài lòng và trung thành của khách hàng.

Chương 2: Tập dữ liệu

2.1 BigBasket Entire Product List

2.1.1 Mô tả dữ liệu

Bộ dữ liệu có 27555 dòng và 10 cột. Truy cập tại đây.

Định nghĩa của từng cột:

- index: Mã định danh duy nhất cho mỗi mục nhập.
- product: Tiêu đề của sản phẩm như được liệt kê trong tập dữ liệu.
- category: Danh mục mà sản phẩm thuộc về.
- sub_category: Danh mục phụ phân loại thêm sản phẩm.
- brand: Thương hiệu gắn liền với sản phẩm.
- sale_price: Giá bán hiện tại của sản phẩm trên nền tảng.
- market_price: Giá thị trường hoặc giá bán lẻ tiêu chuẩn của sản phẩm.
- type: Loại hoặc phân loại của sản phẩm.
- rating: Sự đánh giá của người tiêu dùng đối với sản phẩm. Không phải tất cả các sản phẩm đều nhận được đánh giá.
- description: Mô tả chi tiết về sản phẩm.

2.1.2 Ưu điểm

Độ phong phú: Bộ dữ liệu chứa thông tin về nhiều loại sản phẩm từ nhiều danh mục khác nhau, tạo điều kiện cho việc xây dựng một hệ thống đề xuất đa dạng.

Thiết lập cho Content-Based Filtering: Có thông tin mô tả và thông tin chi tiết về sản phẩm, thuận lợi cho việc sử dụng Content-Based Filtering.

Số lượng lớn: Bộ dữ liệu có số lượng sản phẩm đủ lớn để tạo ra một mô hình có độ chính xác cao.

2.2 House price prediction

2.2.1 Mô tả dữ liệu

Bộ dữ liệu có 4500 dòng và 18 cột. Truy cập tại đây.

Định nghĩa của từng cột:

- date: Ngày bán nhà
- price: Giá bán từng căn nhà
- bedrooms: Số phòng ngủ
- bathrooms: Số lượng phòng tắm, trong đó 0,5 chiếm một phòng có nhà vệ sinh nhưng không có vòi sen
- sqft_living: Không gian sống bên trong căn hộ.
- sqft_lot: Không gian diện tích đất.
- floors: Số tầng.
- waterfront: Một biến giả cho biết căn hộ có nhìn ra bờ sông hay không.
- view: Chỉ số từ 0 đến 4 về mức độ xem của thuộc tính tốt như thế nào.
- condition: Chỉ số từ 1 đến 5 về tình trạng của căn hộ.
- sqft_above: Diện tích của không gian bên trong nhà ở cao hơn mặt đất.
- sqft_basement: Diện tích của không gian bên trong nhà ở dưới mặt đất.
- yr_build: Năm ngôi nhà được xây dựng lần đầu.
- yr_renovated: Năm sửa chữa gần đây nhất của ngôi nhà.
- statezip: Ngôi nhà nằm ở khu vực mã zip nào (77 giá trị duy nhất).
- city: thành phố nơi ngôi nhà tọa lạc (44 giá trị duy nhất).
- country: Hoa Kỳ (USA).

2.2.2 Ưu điểm

Dễ dàng triển khai mô hình dự đoán giá nhà.

2.3 McDonald's Store Reviews

2.3.1 Mô tả dữ liệu

Bộ dữ liệu có 33396 dòng và 10 cột. Truy cập tại đây.

Định nghĩa của từng cột:

- reviewer_id: Id duy nhất của mỗi người đánh giá.
- store_name: McDonald's.
- category: Chỉ có một giá trị duy nhất (Nhà hàng thức ăn nhanh).
- store_address: Địa chỉ của từng nhà hàng ở Hoa Kỳ.
- latitude: vĩ độ của mỗi nhà hàng.
- longitude: kinh độ của mỗi nhà hàng.
- rating_count: số lượng xếp hạng của mỗi nhà hàng.
- review_time: Khoảng thời gian khách hàng đánh giá tính đến thời điểm hiện tại. Có thể hữu ích nếu biết nhà hàng đã thay đổi như thế nào theo thời gian.
- review: đánh giá của khách hàng về nhà hàng (trải nghiệm của họ).
- rating: xếp hạng sao mà khách hàng dành cho nhà hàng.

2.3.2 Ưu điểm

Liên quan đến đề tài nghiên cứu: Bộ dữ liệu về đánh giá cửa hàng McDonald's cung cấp thông tin cần thiết để nghiên cứu về trải nghiệm và ý kiến của khách hàng đối với dịch vụ của nhà hàng.

Phong phú và đa dạng: Bộ dữ liệu bao gồm các yếu tố như vị trí địa lý, đánh giá sao, và văn bản đánh giá, tạo nên một nguồn thông tin đa chiều.

Khả năng Tổng hợp và Phân tích: Dữ liệu chứa thông tin địa lý cho phân tích geospatial và các đánh giá văn bản để thực hiện phân tích sentiment (cảm xúc), giúp ta có cái nhìn toàn diện về trạng thái của các cửa hàng McDonald's.

2.4 Health Insurance Cross Sell Prediction

2.4.1 Mô tả dữ liệu

Bộ dữ liệu có 381109 dòng và 12 cột. Truy cập tại đây.

- id: ID duy nhất cho khách hàng, đóng vai trò là mã định danh cho từng khách hàng trong tập dữ liệu.

- Gender: Giới tính của khách hàng, thường được phân loại là 'Nam' hoặc 'Nữ'.
- Age: Tuổi của khách hàng, thể hiện tuổi của khách hàng theo năm.
- Driving_License: Biến nhị phân cho biết khách hàng có giấy phép lái xe hay không. Nó được mã hóa là 0 đối với khách hàng không có giấy phép lái xe và 1 đối với khách hàng có giấy phép lái xe.
- Region_Code: Mã duy nhất đại diện cho khu vực của khách hàng. Mỗi mã tương ứng với một khu vực địa lý cụ thể.
- Previous_Insured: Biến nhị phân cho biết khách hàng đã có bảo hiểm xe hay chưa. Nó được mã hóa là 1 đối với khách hàng đã có bảo hiểm xe cộ và 0 đối với khách hàng không có bảo hiểm xe cộ.
- Vehicle_Age: Tuổi xe của khách hàng. Nó thường được phân loại thành các nhóm, chẳng hạn như '< 1 Năm', '1-2 Năm' và '> 2 Năm'.
- Vehicle_Damage: Biến nhị phân cho biết xe của khách hàng có bị hư hỏng trước đây hay không. Nó được mã hóa là 1 nếu xe của khách hàng bị hư hỏng và 0 nếu xe không bị hư hỏng.
- Annual_Premium: Số tiền khách hàng cần đóng để mua bảo hiểm xe trong một năm.
- Policy_Sales_Channel: Mã ẩn danh đại diện cho kênh hoặc phương thức được sử dụng để tiếp cận khách hàng để bán bảo hiểm. Điều này có thể bao gồm các đại lý khác nhau, thư từ, cuộc gọi điện thoại, thăm gặp trực tiếp và các phương pháp khác.
- Vintage: Số ngày khách hàng đã gắn kết với công ty bảo hiểm. Cột này đo lường độ dài mối quan hệ của khách hàng với công ty.
- Response: Biến mục tiêu cần dự đoán. Đó là biến nhị phân trong đó 1 cho biết khách hàng quan tâm đến bảo hiểm xe cộ và 0 cho biết khách hàng không quan tâm.

2.4.2 Ưu điểm

Dữ liệu đa dạng: Các bộ dữ liệu cung cấp thông tin đa dạng về khách hàng, chiếc xe và lịch sử bảo hiểm, tạo điều kiện thuận lợi cho việc thực hiện phân tích chi tiết và phát triển mô hình phức tạp.

Kích thước phù hợp: Bộ dữ liệu có kích thước lớn đảm bảo tính đại diện và khả năng tổng quát của mô hình.

Thách thức tính ưng dụng: Dự án liên quan đến một thách thức thực tế trong ngành bảo hiểm, giúp áp dụng kiến thức và kỹ năng phân tích dữ liệu vào một bối cảnh ứng dụng thực tế.

2.5 SuperStore Sales

2.5.1 Mô tả dữ liệu

Bộ dữ liệu có 9800 dòng và 18 cột. Truy cập tại đây.

- Row ID: Mã định danh cho mỗi hàng trong tập dữ liệu.
- Order ID: Mã định danh duy nhất cho mỗi đơn hàng được đặt.
- Order Date: Ngày đặt hàng.
- Ship Date: Ngày đơn hàng được chuyển đi.
- Ship Mode: Phương thức vận chuyển được chọn cho đơn hàng (ví dụ: Hạng hai, Hạng tiêu chuẩn).
- Customer ID: Mã định danh duy nhất cho mỗi khách hàng.
- Customer Name: Tên khách hàng đặt hàng.
- Segment: Phân khúc thị trường mà khách hàng thuộc về (ví dụ: Người tiêu dùng, Doanh nghiệp).
- Country: Quốc gia nơi đặt hàng.
- City: Thành phố nơi khách hàng cư trú.
- State: Tiểu bang nơi khách hàng cư trú.
- Postal Code: Mã bưu chính nơi ở của khách hàng.
- Region: Khu vực nơi khách hàng cư trú.

- Product ID: Mã định danh duy nhất cho mỗi sản phẩm.
- Category: Danh mục rộng mà sản phẩm thuộc về (ví dụ: Nội thất, Vật tư văn phòng).
- Sub-Category: Danh mục phụ cụ thể của sản phẩm (ví dụ: Ghế, Nhãn).
- Product Name: Tên sản phẩm.
- Sales: Doanh thu bán hàng liên quan đến đơn hàng.

2.5.2 Ưu điểm

Đa dạng các khía cạnh: Tích hợp thông tin từ nhiều khía cạnh của mỗi giao dịch, từ đặt hàng đến thông tin vận chuyển và chi tiết sản phẩm, giúp tạo ra một hình ảnh toàn diện.

Ứng dụng thực tiễn: Dữ liệu phản ánh thực tế của môi trường kinh doanh bán lẻ, giúp kết quả phân tích có tính ứng dụng cao trong thực tế.

Khả năng dự đoán: Thông qua phân tích dòng thời gian, bộ dữ liệu này cung cấp cơ hội dự đoán xu hướng tương lai trong doanh số bán hàng.

Chương 3: Nền tảng lý thuyết

3.1 Term Frequency – Inverse Document Frequency (TF – IDF)

TF-IDF là một phương pháp được sử dụng trong xử lý ngôn ngữ tự nhiên và thông tin có mục tiêu chủ yếu là đo lường tầm quan trọng của một từ trong một tài liệu hoặc bộ sưu tập các tài liệu. Nó thường được sử dụng trong các bài toán tìm kiếm thông tin và phân loại văn bản.

3.1.1 Term Frequency (TF)

- Tần suất của một từ trong một tài liệu.
- Được tính theo công thức:

$$TF(t,d) = \frac{Số\ lần\ từ\ t\ xuất\ hiện\ trong\ tài\ liệu\ d}{Tổng\ số\ từ\ trong\ tài\ liệu\ d}$$

Phương trình 1: Công thức tính TF

3.1.2 Inverse Document Frequency (IDF)

- Đo lường mức độ quan trọng của một từ đối với toàn bộ tập tài liệu.
- Được tính theo công thức:

$$IDF(t,D) = \log\left(\frac{Tổng\ số\ văn\ bản\ trong\ tập\ mẫu\ D}{Số\ văn\ bản\ có\ chứa\ từ\ t}\right)$$

Phương trình 2: Công thức tính IDF

3.1.3 TF – IDF Score

- Kết hợp giữa TF và IDF để đánh giá mức độ quan trọng của một từ trong một tài liệu trong ngữ cảnh toàn bộ tập tài liệu.
- Công thức:

$$TF - IDF(t,d,D) = TF(t,d) \times IDF(t,D)$$

Phương trình 3: Công thức tính TF - IDF

3.1.4 Ví dụ

Giả sử rằng một tài liệu có 20 từ và 5 trong số đó là từ "tuyệt vời". Ta có thể tính TF và IDF như sau:

- TF:

$$TF = \frac{5}{20} = 0.25$$

- IDF (giả sử có 5 văn bản và chỉ có 2 văn bản chứa từ "tuyệt vời":

$$IDF = \log\left(\frac{5}{2}\right) = 0.398$$

- Vậy giá trị TF – IDF sẽ là:

$$TF - IDF = TF \times IDF = 0.25 \times 0.398 = 0.0995$$

3.2 Cosine Similarity

Cosine Similarity là một phương pháp đo lường độ tương tự giữa hai vector trong không gian nhiều chiều. Mỗi sản phẩm được biểu diễn bằng một vector TF-IDF trong không gian nhiều chiều, với mỗi chiều tương ứng với một từ trong từ điển của tất cả các mô tả sản phẩm.

Cosine Similarity giữa hai vector được tính bằng cách chia tích vô hướng của hai vector cho tích của độ lớn của mỗi vector. Giá trị của Cosine Similarity nằm trong khoảng từ -1 đến 1. Khi hai vector giống hệt nhau, Cosine Similarity sẽ bằng 1. Khi hai vector hoàn toàn không liên quan (vuông góc với nhau), Cosine Similarity sẽ bằng 0. Khi hai vector đối nhau, Cosine Similarity sẽ bằng -1.

Trong bài báo cáo này em có sử dụng Cosine Similarity để đo lường mức độ tương tự giữa các sản phẩm dựa trên mô tả sản phẩm của chúng. Sản phẩm có độ tương tự cao sẽ có khả năng được đề xuất cho người dùng.

Ví dụ:

Ta có 2 sản phẩm với các vector TF – IDF như sau:

Sản phẩm A: (0.1, 0, 0.2, 0.3, 0)

Sản phẩm B: (0, 0.1, 0.1, 0.2, 0.1)

Để đo độ tương tự giữa 2 sản phẩm A và B. Ta có thể làm như sau:

Tính tích vô hướng của 2 vector:

$$dot(A,B) = A \times B = 0.1 \times 0 + 0 \times 0.1 + 0.2 \times 0.1 + 0.3 \times 0.2 + 0 \times 0.1 = 0.08$$

Tính độ lớn của mỗi vector:

$$||A|| = \sqrt{0.1^2 + 0^2 + 0.2^2 + 0.3^2 + 0^2} = \sqrt{0.14}$$

$$||B|| = \sqrt{0^2 + 0.1^2 + 0.1^2 + 0.2^2 + 0.1^2} = \sqrt{0.07}$$

Cuối cùng, áp dụng công thức cosine similarity:

$$cosine\ similarity(A,B) = \frac{dot(A,B)}{||A|| \times ||B||} = \frac{0.08}{\sqrt{014} \times \sqrt{0.07}} = 0.808$$

Vậy, độ tương đồng giữa sản phẩm A và B là 0.808 theo cosine similarity. Điều này có nghĩa là hai sản phẩm này rất giống nhau dựa trên mô tả sản phẩm của chúng.

3.3 Thuật toán Decision Tree

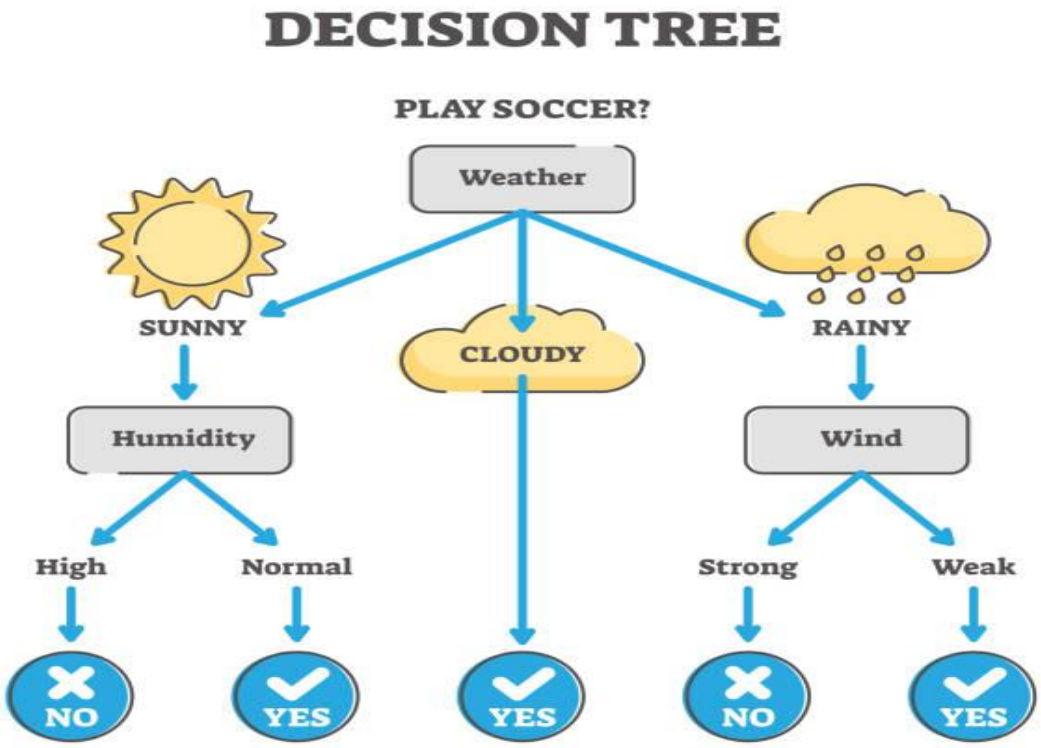
Decision Tree được xây dựng bằng các luật phân chia các phân lớp dữ liệu. Decision Tree được xây dựng trên ba thành phần chính:

- Nút (Node): mỗi node thể hiện đặc trưng (thuộc tính, tính chất của dữ liệu). Trong đó, nút gốc là nút trên cùng của cây.
- Nhánh (Branch): mỗi nhánh mô tả một quy luật của dữ liệu.
- Lá (Leaf): mỗi là biểu diễn một kết quả của phân lớp.

Có hai câu hỏi, em nghĩ mình cần đưa ra khi nói về Decision Tree:

a) Tại sao chúng ta lại chọn thuật toán Decision Tree?

Hình 1 (Trang 14) là một ví dụ minh họa về mô hình thuật toán Decision Tree về việc quyết định có đi chơi bóng đá hay không?



Hình 1: Minh họa cách thức hoạt động Decision Tree về dữ liệu chơi bóng đá.

- Nút gốc ở đây là đặc trưng (Weather) của tập dữ liệu
- Chia thành ba nhánh (Sunny, Cloudy, Rainy, ba giá trị của đặc trưng weather)
- Nhánh Sunny, Rainy chưa quyết định đi chơi được nên sẽ quyết định tiếp dựa trên đặc trưng Humidity hay Wind, nhánh Cloudy dẫn đến đi chơi luôn.
- Tiếp túc từ nút (đặc trưng) chia ra các nhánh chứa giá trị đặc trưng của chúng cho đến khi quyết định được đi chơi hay không (yes/no)

Có thể thấy, Decision Tree bắt trước mức độ suy nghĩ của con người nên nó đơn giản và dễ hiểu. Điều quan trọng là nó giúp ta thấy được logic từ dữ liệu.

b) Làm thế nào để chọn đặc trưng cho các nút?

Tại mỗi nút, các đặc trưng sẽ được đánh giá dựa trên việc chia các lớp mục tiêu của toàn bộ dữ liệu huấn luyện. Ở đây, một đơn vị đo sẽ được sử dụng: hỗn tạp (Impurity). Một số thuật toán để đo Impurity hay nói cách khác là để tạo cây:

- ID3(Iteractive Dichotomiser 3) dùng Entropy function và Information Gain (IG) để kiểm tra.
- CART (Classification and Regression Trees) dùng Gini index để kiểm tra.

Ví dụ ta sử dụng IG để xây dựng cây. Giá trị IG sẽ tỉ lệ thuận với độ trong suốt trung bình của các tập con mà đặc trưng tạo ra. Khi đi xây dựng mô hình, ta sẽ phát triển từ gốc đến nhánh mà tại mỗi nút ta sẽ tính IG rồi chọn đặc trưng mà tạo ra IG lớn nhất. Các bước lần lượt là:

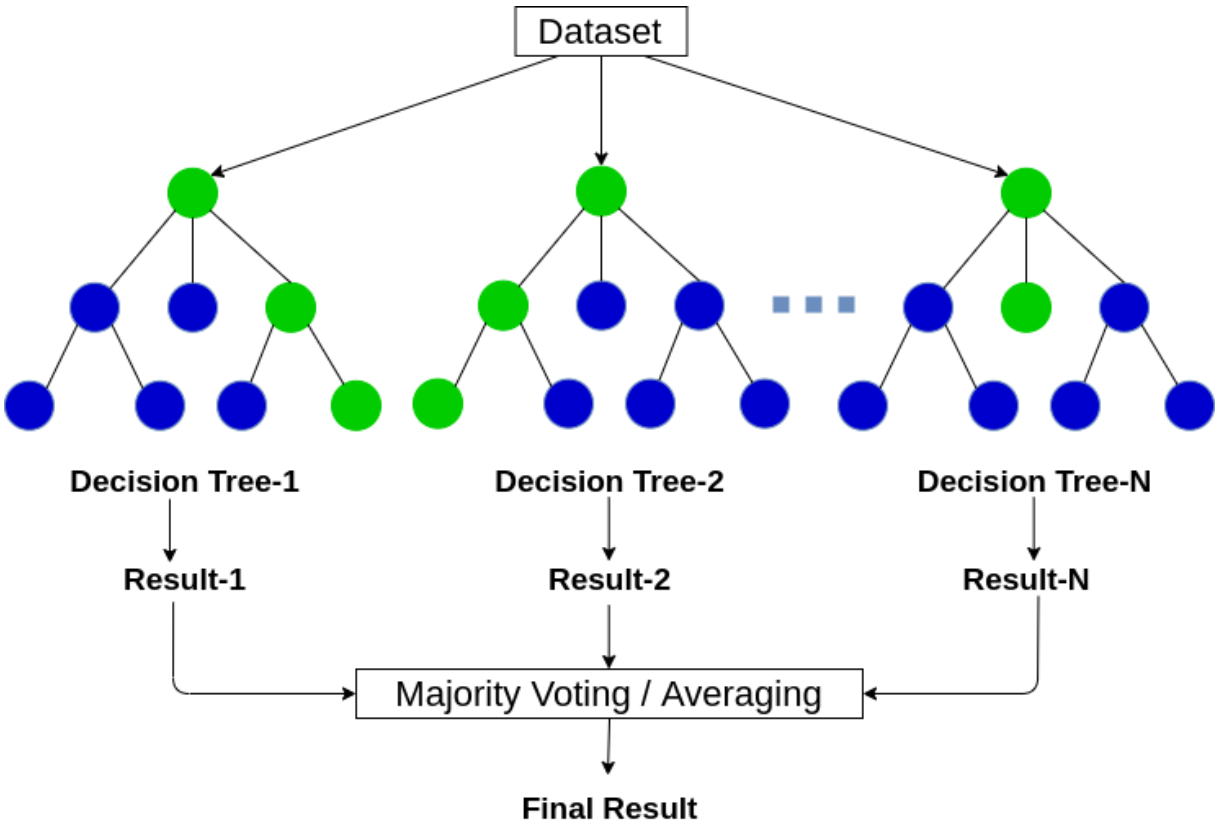
1. Chọn thuộc tính "tốt" nhất bằng một độ đo đã định trước.
2. Phát triển cây bằng việc thêm các nhánh tương ứng với từng giá trị của thuộc tính đã chọn.
3. Sắp xếp, phân chia tập dữ liệu đào tạo tới node con.
4. Nếu các samples được phân lớp rõ ràng thì dừng.
5. Ngược lại: lặp lại bước 1 tới bước 4 cho từng node con.

3.4 Thuật toán Random Forest

Khi đọc tên thuật toán "Rừng ngẫu nhiên" chắc hẳn ta cũng nghĩ nó là tập hợp nhiêu cây để tạo nên một cánh rừng. Đúng vậy, Random Forest sử dụng kỹ thuật lấy mẫu Boostraps. Từ đó kết hợp nhiều mô hình Decision Tree để tạo ra một "rừng cây".

Để tạo nên một mô hình Random Forest, ta tuân theo các bước như sau:

1. Từ tập dữ liệu ban đầu, dùng phương pháp lấy mẫu Boostraps tạo thành một K tập dữ liệu con.
2. Với mỗi dữ liệu con tạo một cây quyết định. Tại mỗi nốt chia, chọn ngẫu nhiên số lượng x feature (x= sqrt(m), m là số lượng các đặc trưng). Với K cây quyết định thì tạo thành một Rừng.



Hình 2: Minh họa cách thức hoạt động của một mô hình Random Forest dùng voting hay lấy kết quả trung bình để đưa ra kết quả cuối cùng.

3.5 Thuật toán ARIMA

ARIMA, viết tắt của Auto-Regression Integrated Moving Average, là một mô hình được sử dụng để phân tích thống kê dữ liệu chuỗi thời gian. Nó giúp hiểu rõ hơn về dữ liệu và dự đoán xu hướng trong tương lai. Mô hình này thường được gọi là ARIMA(p, d, q), với các số nguyên tương ứng với các phần trung bình tự hồi quy, tích hợp và chuyển động của tập dữ liệu một cách tương ứng.

Các bước thực hiện thuật toán ARIMA như sau:

- Kiểm tra tính dừng: Đầu tiên, ta cần kiểm tra xem chuỗi thời gian có đứng yên hay không. Một chuỗi thời gian đứng yên nếu giá trị trung bình, phương sai và cấu trúc tự tương quan không thay đổi theo thời gian.
- Xử lý chuỗi không dừng: Nếu chuỗi thời gian không đứng yên, ta cần phải biến đổi nó để trở nên đứng yên.

- Chọn bậc AR (p) tối ưu: Chọn bậc p tối ưu cho phần tự hồi quy của mô hình.
- Chọn bậc MA (q) tối ưu: Tiếp theo, chọn bậc q tối ưu cho phần trung bình động của mô hình.
- Ước lượng mô hình ARIMA (p, d, q) và chọn mô hình tối ưu: Sau khi đã xác định được các bậc p, d, q, tiến hành ước lượng mô hình ARIMA và chọn mô hình tối ưu.
- Dự báo: Cuối cùng, ta sử dụng mô hình đã được ước lượng để dự báo chuỗi thời gian trong tương lai.

Chương 4: Phương pháp nghiên cứu

4.1 Hệ thống đề xuất sản phẩm

4.1.1 Thu thập dữ liệu

Dữ liệu được tải về từ Kaggle và chứa thông tin chi tiết về hơn 27,000 sản phẩm có sẵn trên trang web, bao gồm tên, danh mục, giá bán, giá thị trường, thương hiệu, loại, đánh giá, và mô tả.

4.1.2 Tiền xử lý dữ liệu

Loại bỏ dữ liệu trùng lặp và không dùng đến: Các sản phẩm trùng lặp được loại bỏ để đảm bảo tính chính xác của dữ liệu.

Tiền xử lý văn bản: Mô tả sản phẩm (category, sub_category, brand, type) được tiền xử lý để loại bỏ các ký tự đặc biệt, chuyển đổi về chữ thường, và thực hiện các bước khác để chuẩn hóa văn bản.

4.1.3 Xây dựng hệ thống đề xuất sản phẩm

TF-IDF và Cosine Similarity: Sử dụng phương pháp Content-Based Filtering với TF-IDF và cosine similarity để đo lường sự tương đồng giữa các sản phẩm dựa trên mô tả của chúng.

Loại bỏ stop_words: giúp loại bỏ các từ không có nhiều ý nghĩa như: a, the,…

Vector hóa dữ liệu: Dữ liệu mô tả sản phẩm được biểu diễn dưới dạng vector để tính toán sự tương đồng.

Xây dựng hệ thống đề xuất sản phẩm: Dựa trên sự tương đồng tính toán được, hệ thống đề xuất 10 sản phẩm tương tự cho mỗi sản phẩm đầu vào.

4.2 Dự đoán giá nhà (House price prediction)

4.2.1 Thu thập dữ liệu

Dữ liệu được tải về trên Kaggle và chứa thông tin chi tiết cho hơn 4000 điểm dữ liệu về những thứ ảnh hưởng đến giá nhà: vị trí địa lý, diện tích đất đai, …

4.2.2 Tìm hiểu và trực quan hóa dữ liệu

Phân tích thống kê: Số lượng phân phối, giá trị mean, giá trị tứ phân vị của các Features (cột trong tập dữ liệu). Giúp ta có cái nhìn bề nổi của tập dữ liệu.

Trực quan hóa dữ liệu: Giúp ta có cái nhìn sâu về dữ liệu, góp phần quyết định vào việc xử lý dữ liệu hay cải tiến mô hình.

4.2.3 Tiền xử lý dữ liệu và Feature Engineering

Loại bỏ outliers: Outliers có thể tạo ra thiên kiến tiêu cực cho toàn bộ kết quả của một phân tích. Việc loại bỏ outliers giúp tránh thiên kiến này, đảm bảo rằng kết quả phân tích phản ánh chính xác dữ liệu. tăng độ chính xác cho kết quả phân tích và mô hình dự đoán.

Làm giàu dữ liệu: Tạo thêm các Features cải thiện độ chính xác của mô hình.

4.2.4 Xây dựng mô hình dự đoán

Sử dụng thuật toán: Linear Regression và Random Forest để dự đoán giá nhà dựa trên các feature đã qua xử lý. Tinh chỉnh các tham số của mô hình để đưa ra kết quả tốt nhất cho mô hình.

Đánh giá bằng các chỉ số: Sử dụng các chỉ số R squared (R2), Root Mean Squared Error (RMSE) để đánh giá mô hình.

4.3 McDonald’s Store Reviews

4.3.1 Thu thập dữ liệu

Bộ dữ liệu được tải từ Kaggle với các văn bản đánh giá, thông tin địa lý, ratings của người tiêu dùng về các cửa hàng McDonald’s tại Hoa Kỳ.

4.3.2 Tiền xử lý dữ liệu

Xử lý Nan (missing values): địa chỉ nhà hàng bị lỗi (có thể do lỗi font chữ khi encode ra file csv) dẫn đến vĩ độ (latitude) và kinh độ (longitude) không xác định được. Ta có thể tìm tên địa chỉ hoặc thành phố nơi nhà hàng tọa lạc từ các văn bản review rồi tìm kiếm trên google để chỉnh sửa lại.

Sửa lại kiểu dữ liệu: chỉnh sửa kiểu dữ liệu category (rating) sang dạng numeric thuận tiện cho việc phân tích và xây dựng mô hình.

Xử lý văn bản: loại bỏ các từ không có nghĩa hay các ký tự không cần thiết có thể gây ra nhầm lẫn cho mô hình. Sử dụng kỹ thuật pos_tag và lemmazitation để cho chuyển chữ về đúng định dạng ngữ pháp.

Phân tích sentiment dựa trên rating và TextBlob.

4.3.3 Xây dựng mô hình dự đoán

Vector hóa dữ liệu bằng TF-IDF. Sử dụng các thuật toán quen thuộc trong lĩnh vực phân tích sentiment của văn bản là Multinomial Naïve Bayes, SVC, Decision Tree, Random Forest.

Sử dụng độ chính xác (accuracy score) để đánh giá mô hình.

Giảm chiều dữ liệu cho ma trận TF – IDF để làm mất đi những dữ liệu nhiễu.

4.3.4 Trực quan hóa dữ liệu và phân tích không gian địa lý

Vẽ biểu đồ WordCloud để có cái nhìn về từng vấn đề mà cửa hàng gặp phải, thậm chí là trải nghiệm của khách hàng như nào qua biểu đồ.

Dựa vào kinh độ và vĩ độ để vẽ bản đồ mật độ với giá trị rating trung bình mà từng cửa hàng nhận được. Giúp ta có cái nhìn tổng quát về trải nghiệm của khách hàng với khu vực nhà hàng. Qua đó cải thiện

4.4 Dự đoán bán chéo về bảo hiểm

4.4.1 Thu thập dữ liệu

Dữ liệu được tải từ Kaggle, được cung cấp bởi một công ty bảo hiểm với mục đích dự đoán khả năng quan tâm của khách hàng đối với bảo hiểm xe hơi. Thông tin chi tiết của khách hàng, lịch sử bảo hiểm, và chi tiết về phương tiện đã được cung cấp để hỗ trợ quá trình phân tích.

4.4.2 Phân tích và trực quan hóa dữ liệu

Vẽ các biểu đồ như countplot, histogram, boxplot, scatterplot,… để hiểu các phân phối, ý nghĩa của feature.

4.4.3 Feature Engineering

Feature Binning: Phân chia nhóm tuổi (Age) thành các nhóm nhỏ hơn để giảm ảnh hưởng.

Feature Interaction: Kết hợp hai hoặc nhiều tính năng để tạo các thuật ngữ tương tác. Ví dụ: ta có thể nhân "Age" và "Annual_Premium" để nắm bắt sự tương tác giữa tuổi và khoản thanh toán phí bảo hiểm.

Feature Scaling: chuẩn hóa các Features như “Age”, “Annual_Premium” để chúng có tỷ lệ tương tự.

Feature Aggreation: Tổng hợp các Features, chẳng hạn như tính toán “Response” trung bình hay “Annual_Premium” cho từng khu vực (Region_Code).

4.4.4 Xây dựng mô hình

Sử dụng thuật toán Random Forest và tỉnh chỉnh các tham số để đưa ra được mô hình có kết quả tốt nhất.

Đánh giá mô hình dựa trên chỉ số ROC AUC.

4.5 Dự đoán giá bán lẻ dựa trên chuỗi thời gian

4.5.1 Thu thập dữ liệu

Bộ dữ liệu được tải từ Kaggle, chứa thông tin chi tiết về các giao dịch bán lẻ trong suốt 4 năm từ 2015 – 2018.

4.5.2 Tiền xử lý dữ liệu

Xử lý dữ liệu bị thiếu (Postal Code), tra trên google mã Postal của thành phố để điền vao.

4.5.3 Phân tích và trực quan hóa dữ liệu

Hiểu rõ đặc điểm chuỗi thời gian, biến động ví dụ như xu hướng, mùa vụ hay các biến động ngắn hạn.

Nắm bắt được xu hướng dài hạn giúp trong việc dự đoán và kế hoạch chiến lược, trong khi việc xác định các mùa vụ ngắn hạn giúp điều chỉnh chiến lược ngắn hạn.

4.5.4 Xây dựng mô hình

Sử dụng thuật toán ARIMA để dự đoán giá bán lẻ dựa trên chuỗi thời gian.

Tinh chỉnh tham số p và q dựa vào biểu đồ ACF (autocorrelation) và PACF (Partial autocorrelation) để cho ra mô hình dự đoán tốt nhất. Sử dụng chỉ số RMSE để đánh giá mô hình.

Dựa vào mô hình dự đoán tốt nhất, dự đoán giá bán lẻ 7 ngày tiếp theo.

Chương 5: Kết quả và thảo luận

5.1 Hệ thống đề xuất sản phẩm

5.1.1 Kết quả

```
def recommendations(title, cosine_sim = cos_sim):
    recommended_product = []
    index = indices[indices == title].index[0]
    similarity_scores = pd.Series(cosine_sim[index]).sort_values(ascending = False)
    top_10_product = list(similarity_scores.iloc[1:11].index)
    for i in top_10_product:
        recommended_product.append(list(df2['product'])[i])
    return recommended_product
```

```
recommendations("Turmeric Powder/Arisina Pudi")
```

```
['Powder - Chilli',
 'Combo Pack - Chilli, Turmeric & Coriander (200g Each)',
 'Compounded Asafoetida - Cake',
 'Asafoetida Powder',
 'Punjabi Chole Masala',
 'Paneer Masala',
 'Biriyani masala',
 'Meat/Mutton Masala',
 'Red Chilli Powder 200G +Coriander/Dhania Powder 200G +Turmeric/Haldi Powder 200G',
 'Chicken Tandoori Masala']
```

Hình 3: Hàm trả về các đề xuất sản phẩm tương tự

- Hệ thống đề xuất đã được xây dựng thành công sử dụng phương pháp Content-Based Filtering dựa trên TF-IDF và cosine similarity.
- Mỗi sản phẩm đầu vào được kết quả với danh sách 10 sản phẩm tương tự dựa trên mô tả.

5.1.2 Nhận xét

Ý nghĩa:

- Kết quả cho thấy hệ thống đề xuất là một công cụ mạnh mẽ để tăng cường trải nghiệm mua sắm và thúc đẩy doanh thu trong môi trường thương mại điện tử.
- Người dùng nhận được các gợi ý mua sắm chính xác và phù hợp với sở thích và nhu cầu của họ.

Thách thức và giải pháp:

- Đối mặt với sự hạn chế của Content-Based Filtering trong việc khám phá các sản phẩm mới và đa dạng.
- Kết hợp với Collaborative Filtering để cải thiện khả năng khám phá và đề xuất sản phẩm dựa trên hành vi mua sắm của người dùng khác.

5.1.3 App Demo

Hình 4: App demo hệ thống đề xuất sản phẩm

Em đã xây dựng một app demo và deploy trên website để tăng tính trải nghiệm người dùng. Truy cập tại đây.

5.2 Dự đoán giá nhà

5.2.1 Kết quả

	Model	R2	Cross-Validated R2	RMSE
0	Linear Regression	-102547175773427008.0000	0.6784	69458632859158.9375
1	Random Forest Regression	0.6728	0.6780	124063.7977
2	RF (drop sqft_above)	0.6729	0.6751	124042.8914
3	RF best params	0.6833	0.6958	122066.3784

Hình 5: Kết quả chỉ số đánh giá mô hình dự đoán giá nhà

5.2.2 Nhận xét

Linear Regression: R2 có giá trị âm, cho biết mô hình đang hoạt động kém. RMSE cũng cực kỳ cao, cho thấy dự đoán của mô hình còn rất xa so với giá trị thực tế. Mô hình này có vẻ không phù hợp với dữ liệu.

Random Forest Regression: Mô hình này có điểm R2 là 0.6728 và điểm Cross – Validated R2 là 0.6780, cho thấy rằng mô hình này giải thích được khoảng 67% phương sai trong dữ liệu. RMSE là 124063.7977, thấp hơn đáng kể so với mô hình Linear Regression.

RF (dropsqft_above): Mô hình này hoạt động tương tự như mô hình Hồi quy rừng ngẫu nhiên, với điểm R2 cao hơn một chút là 0.6729 và điểm Cross – Validated R2 thấp hơn một chút là 0.6751. RMSE thấp hơn một chút ở mức 124042.8914.

RF best params: Mô hình này hoạt động tốt nhất trong số tất cả các mô hình, với điểm R2 là 0.6833 và điểm Cross – validated R2 là 0.6958, cho thấy rằng nó giải thích được khoảng 69% phương sai trong dữ liệu. RMSE cũng thấp nhất ở mức 122066.3784.

Kết quả mô hình chưa được cao lắm. Có thể do vẫn còn dữ liệu nhiễu làm giảm độ chính xác của mô hình, bên cạnh đó có lẽ các phương pháp Feature Engineering chưa phù hợp.

5.3 McDonald's Store Review

5.3.1 Kết quả

a) Dự đoán Sentiment chia bởi rating

- Multinomial Naive Bayes với độ chính xác 79.7%.
- Support Vector Classifier (SVC) với độ chính xác 83.1%.
- Random Forest Classifier với độ chính xác 83.1%.

b) Dự đoán Sentiment được phân tích bởi TextBlob

- MultinomialNB với độ chính xác 81.4%.

- SVC với độ chính xác 0.95%.
- RandomForestClassifier với độ chính xác 93%.
- DecisionTreeClassifier: với độ chính xác 94.5%

5.3.2 Nhận xét

Mô hình SVC cho ra độ chính xác cao nhất với 95% nhưng thời gian chạy rất lâu. Trong khi đó DecisionTreeClassifier chạy chưa đến 5 giây nhưng lại cho ra độ chính xác đáng kinh ngạc 94.5%.

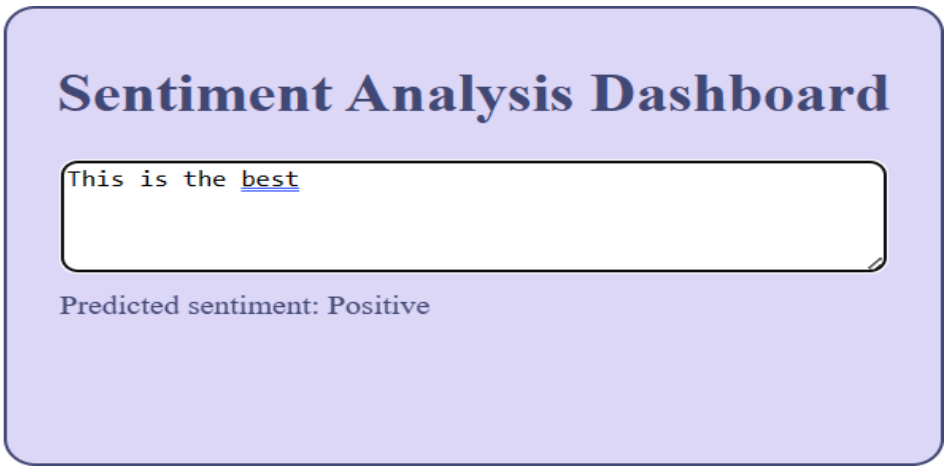
5.3.3 App Demo

```
def map_sentiment(sentiment_label):
    sentiment_mapping = {
        -1: "Negative",
        0: "Neutral",
        1: "Positive"
    }
    return sentiment_mapping.get(sentiment_label)
def predict_sentiment(review):
    review_tfidf = vectorizer.transform([review])
    sentiment_label = classifier.predict(review_tfidf)[0]
    sentiment_word = map_sentiment(sentiment_label)
    print("Predicted sentiment:", sentiment_word)
```

```
predict_sentiment("This is awesome")
```

```
Predicted sentiment: Positive
```

Hình 6: Hàm dự đoán sentiment của văn bản



Hình 7: App demo dự đoán sentiment văn bản

5.4 Dự đoán bán chéo về bảo hiểm

5.4.1 Kết quả

Base Model (RandomForestClassifier, random_state = 42): Mô hình RandomForest ban đầu với n_estimators=10 đã đạt được điểm ROC AUC là 0,795.

Feature Binning: Việc áp dụng tính năng gộp tính năng cho "Age" và "Annual_Premium" không cải thiện đáng kể hiệu suất của mô hình, với điểm ROC AUC là 0,7919 cho n_estimators=10 và nhưng cải thiện đáng kể với điểm số lên đến 0,8363 cho n_estimators=100.

Feature Aggreation: Việc tổng hợp các tính năng, tính toán phản hồi trung bình và phí bảo hiểm cho từng khu vực, dẫn đến điểm ROC AUC là 0,8358 cho n_estimators=100.

Kết hợp Feature Interaction và Feature Aggreation: Kết hợp tương tác Features với tổng hợp Features dẫn đến kết quả tốt hơn, với điểm ROC AUC là 0,8372 cho n_estimators=100 và 0,8388 cho n_estimators=200.

Feature Selection: Việc loại bỏ Feature "Driving_Lilence" sau khi tương tác và tổng hợp tính năng có tác động tối thiểu đến điểm ROC AUC, dẫn đến điểm 0,8383 cho n_estimators=200.

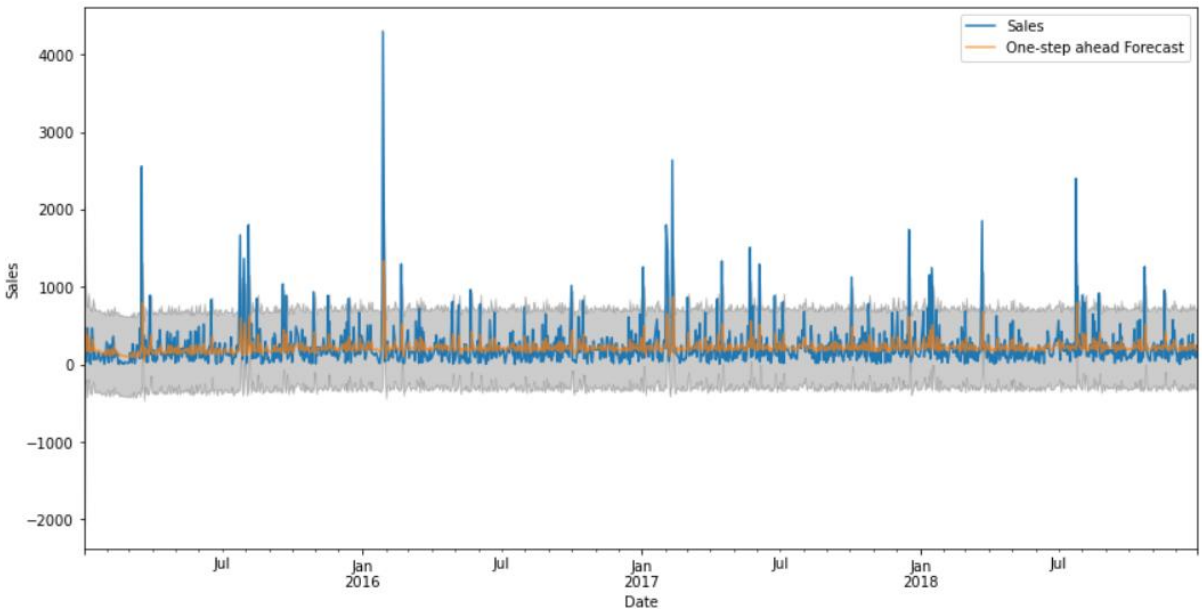
Mô hình tốt nhất với điều chỉnh siêu tham số: Mô hình hoạt động tốt nhất đã đạt được điểm ROC AUC là 0,8579 và có được bằng cách sử dụng RandomizedSearchCV để tìm siêu tham số tối ưu trong khi bao gồm tương tác và tổng hợp Feature.

5.4.2 Nhận xét

Với sự kết hợp của các cách trong Feature Engineering và tinh chỉnh tham số. Độ chính xác của mô hình được cải thiện lên rất nhiều (6.3%).

5.5 Dự đoán giá bán lẻ dựa trên chuỗi thời gian

5.5.1 Kết quả



Hình 8: Biểu đồ thể hiện giá cả bán lẻ được dự đoán so với giá bán lẻ thực

Chỉ số RMSE của mô hình là 262.96

2018-12-31	201
2019-01-01	214
2019-01-02	229
2019-01-03	228
2019-01-04	227
2019-01-05	226
2019-01-06	226

Hình 9: Giá bán lẻ được dự đoán trong 7 ngày tiếp theo của tập dữ liệu

5.5.2 Nhận xét

RMSE (Root Mean Square Error): Với giá trị RMSE đạt 262.96, mô hình có vẻ đưa ra dự đoán chấp nhận được.

Dự đoán 7 ngày tiếp theo: Dự đoán cho thấy một xu hướng giảm dần trong 7 ngày tiếp theo.

Tính ổn định: Các giá trị dự đoán duy trì ổn định, không có biến động lớn, điều này có thể chỉ ra sự ổn định trong dự đoán.

TỔNG KẾT

Trên hành trình nghiên cứu Sales Prediction với năm dự án nhỏ trên Kaggle,ta đã khám phá và ứng dụng một loạt các phương pháp để dự đoán và phân tích doanh số bán hàng. Mỗi chủ đề của dự án là một cuộc phiêu lưu sâu sắc vào các lĩnh vực cụ thể liên quan đến Sales Prediction, từ hệ thống đề xuất sản phẩm đến dự đoán giá nhà, phân tích đánh giá khách hàng, dự đoán ý định mua bảo hiểm xe cộ, và cuối cùng là dự đoán doanh số bán hàng theo chuỗi thời gian.

Qua việc sử dụng nhiều phương pháp như TF-IDF, cosine similarity, Linear Regression, Random Forest, Decision Tree, ARIMA, ta không chỉ đạt được những hiểu biết sâu sắc về từng khía cạnh của doanh số bán hàng mà còn mang lại giá trị thực tế cho doanh nghiệp. Những kết quả này không chỉ là nền tảng cho quá trình ra quyết định và chiến lược kinh doanh mà còn là sự chứng minh rằng sự đa dạng trong phương pháp là chìa khóa cho hiểu biết toàn diện và ứng dụng thực tế.

TÀI LIỆU THAM KHẢO

[1] AYUSH VERMA, BigBasket Product Recommendation Systems: BigBasket Product Recommendation System | Kaggle, truy cập cuối cùng ngày 12/11/2023.

[2] TF – IDF: tf–idf - Wikipedia, truy cập cuối cùng ngày 12/11/2023.

[3] ROBIN S, House Price Prediction Beginner's Notebook: House Price Prediction Beginner's Notebook | Kaggle, truy cập cuối cùng ngày 12/11/2023.

[4] ANTHIME VALIN, Talking Points & Geographical Analyses, Talking Points & Geographical Analyses | Kaggle, truy cập cuối cùng ngày 12/11/2023.

[5] SWAYAM PATIL, Reviews Sentiment Analysis || ML Predictive Model: Reviews Sentiment Analysis || ML Predictive Model | Kaggle, truy cập cuối cùng ngày 12/11/2023.

[6] NLTK (Nature Language Toolkit): Natural Language Processing With Python's NLTK Package – Real Python, truy cập cuối cùng ngày 12/11/2023.

[7] ROSHAN KUMAR G, #Rank 10 solution cross sell prediction hackathon: #Rank 10 solution cross sell prediction hackathon | Kaggle, truy cập cuối cùng ngày 12/11/2023.

[8] YASHVI PATEL, Vehicle Insurance EDA and boosting models: Vehicle Insurance EDA and boosting models | Kaggle, truy cập cuối cùng ngày 12/11/2023.

[9] SAMRUDDHI MHATRE, Exploratory Data Analysis and Time Series Analysis: Part 1: Exploratory Data Analysis | Kaggle, truy cập cuối cùng ngày 12/11/2023.