Đánh Giá Thành Phần Dinh Dưỡng Và Món Ăn Với Bộ Dữ Liệu Epicurious.

Nguyễn Thế Mạnh[1], Lê Quang Nhật[1], Đỗ Trọng Hợp[1]

Đại học Công nghệ Thông tin - Đại học Quốc gia Thành phố Hồ Chí Minh
{18521084, 18521190}@gm.uit.edu.vn,hopdt@uit.edu.vn

Tóm tắt nội dung Epicurious - Bộ dữ liệu được tạo ra từ trang web cùng tên được tạo ra để khám phá các yếu tố khác nhau ảnh hưởng đến việc thưởng thức thực phẩm và cách nấu ăn của mọi người! Việc đưa ra các thành phần và hướng dẫn cách nấu món ăn góp phần cho chúng ta có những món ăn đa dạng. VỚi tổng dữ liệu là 20057 điểm dữ liệu và 680 cột tương ứng với thành phần món ăn, địa điểm event, phương pháp nấu,.... Tuy nhiên trong bài toán này, Chúng tôi chỉ dừng ở việc phát triển bài toán này bằng xây dựng mô hình và đánh giá mô hình máy học bằng so sánh các thành phần năng lượng như fat, sodium, calories, protein. Ngoài ra chúng tôi còn đánh giá có phải là ăn chay tiêu chuẩn quốc tế kosher, từ đầu vào là món ăn, cách nấu, và nồng độ dinh dưỡng. và đầu ra là có phải là người ăn chay kosher hay không? Bộ dữ liệu này có dữ liệu thiếu chiếm 20% bộ dữ liệu nên chúng tôi muốn thực nghiệm bằng hai phương pháp. Thực nghiệm 1 là xóa dữ liệu Null.(EraserNULL) chúng tôi sẽ xóa toàn bộ dữ liệu NULL và thực hiện xử lý. Thực nghiệm 2 làthay thế dữ liệu Null bằng phương pháp mean.(ADDNULL) chúng tôi sẽ thay thế toàn bộ toàn bộ dữ liệu NULL và thực hiện xử lý các phương pháp xử lý và phân tích dữ liệu như trên.

Keywords: Đèn tín hiệu, deep learning.

1 Giới Thiệu

Ở mọi thời đại, thức ăn luôn đóng vai trò quan trọng trong cuộc sống của con người bởi nó cung cấp chất dinh dưỡng và năng lượng từ các thành phần của món ăn để chúng ta có thể đáp ứng nhu cầu hoạt động của mình. Trong xã hội hiện đại ngày nay, món ăn cũng ngày càng đa dạng, phong phú và bổ dưỡng mang nét đặc trưng và cả bản sắc văn hoá của từng vùng miền trên thế giới. Do đó, việc thiếu thức ăn là điều không thể bàn cãi. Song, với thế giới ngày càng phát triển hiện nay, việc thiếu hụt hay không cân bằng chất dinh dưỡng vẫn còn là vấn đề khá nhức nhối trên thế giới không chỉ. Theo thống kê ở Việt Nam, tỷ lệ suy dinh dưỡng thấp còi ở trẻ em tuổi học đường (5 - 19 tuổi) còn 14,8% (năm 2010 tỷ lệ này là 23,4%). Rất đáng lưu ý là tỷ lệ thừa cân, béo phì tăng từ 8,5% năm 2010 lên thành 19,0% năm 2020, trong đó tỷ lệ thừa cân béo phì khu vực thành thị là 26,8%, nông thôn là 18,3% và miền núi là 6,9%. Do đó, để

đảm bảo cho việc cân bằng chất dinh dưỡng cho con người, chúng ta cần thống kê dữ liệu món ăn, đưa ra được các thành phần món ăn và có thể đưa ra được con số cụ thể các món ăn đó cung cấp được hàm lương là bao nhiêu về các chất dinh dưỡng và năng lượng cho con người. Và từ đó, mọi người có thể đảm bảo được khẩu phần ăn của mình được tốt hơn và đầy đủ hơn.

Epicurious là một thương hiệu kỹ thuật số của Mỹ tập trung vào các chủ đề liên quan đến thực phẩm và nấu ăn. Thương hiệu này đã đóng góp rất nhiều trong việc cung cấp dữ liệu để giải quyết các vấn đề nêu trên. Với bộ dữ liệu được tạo ra từ thương hiệu này được tạo ra để khám phá các yếu tố khác nhau ảnh hưởng đến việc thưởng thức thực phẩm và cách nấu ăn của mọi người! Việc đưa ra các thành phần và hướng dẫn cách nấu món ăn góp phần đưa ra cho chúng ta rất nhiều bài toán để phân tích. Trong đó, bài toán có thể kể đến đó là phân tích bộ dữ liệu để dự đoán lượng calories trong món ăn với những thành phần dinh dưỡng của món ăn như protein, fat, sodium với các mô hình như Linear Regression, Random Forest Regressor và bài toán phân loại món ăn Kosher bằng việc áp dụng mô hình máy học phân loại Logistic Regression và Random Forest Regressor để phân biệt. Việc ứng dụng với hai bài toán đưa ra là vô cùng thực tiễn. .

Với các mô hình máy học hiện có, để phù hợp với việc phân tích dữ liệu trên, thì các mô hình như Linear Regression, Random Forest Regressor hay Logistic Regression cũng khá phù hợp với hai bài toán đã đặt ra. Kết quả thực nghiệm được phân tích và đánh giá để chọn ra mô hình tốt nhất.

2 Cơ Sở Lí Thuyết

2.1 Pyspark

Apache Spark là một trong những khung được sử dụng rộng rãi nhất khi xử lý và làm việc với Big Data và Python là một trong những ngôn ngữ lập trình được sử dụng rộng rãi nhất cho Phân tích dữ liệu, Học máy và hơn thế nữa.Apache Spark là một khung công tác điện toán cụm nguồn mở để xử lý thời gian thực được phát triển bởi Quỹ phần mềm Apache. Spark cung cấp một giao diện để lập trình toàn bộ các cụm với sự song song dữ liệu ngầm và khả năng chịu lỗi. Hình 1 mô tả những lợi thế của Apache Spark

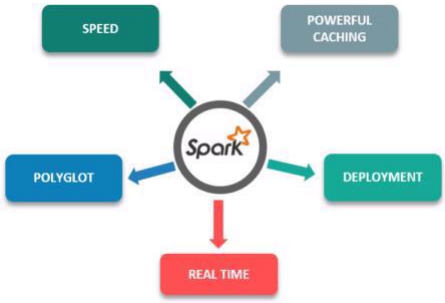
Tốc độ: Nó nhanh hơn 100 lần so với các khung xử lý dữ liệu quy mô lớn truyền thống. **Bộ nhớ đệm mạnh mẽ**: Lớp lập trình đơn giản cung cấp khả năng lưu trữ bộ nhớ cache và ổ đĩa mạnh mẽ.

Triển khai: Có thể được triển khai thông qua Mesos, Hadoop thông qua Sợi hoặc trình quản lý cụm riêng của Spark.

Thời gian thực: Tính toán thời gian thực và độ trễ thấp vì tính toán trong bộ nhớ.

Polyglot: Đây là một trong những tính năng quan trọng nhất của khung này vì nó có thể được lập trình bằng Scala, Java, Python và R.

Nói về Spark với Python, thư viện Py4j có thể làm việc với RDDs. PySpark Shell liên kết API Python với Spark Core và khởi chạy Bối cảnh Spark. Spark Context là trung tâm của bất kỳ ứng dụng Spark nào.



Hình 1: Lợi thế của Apache Spark

Spark Context thiết lập các dịch vụ nội bộ và thiết lập kết nối với môi trường thực thi Spark.

Đối tượng Spark Context trong chương trình trình điều khiển phối hợp tất cả các quy trình phân tán và cho phép phân bổ tài nguyên.

Các trình quản lý cụm cung cấp các Executor, đó là các quy trình JVM có logic.

Các đối tượng Spark Context gửi ứng dụng cho người thi hành.

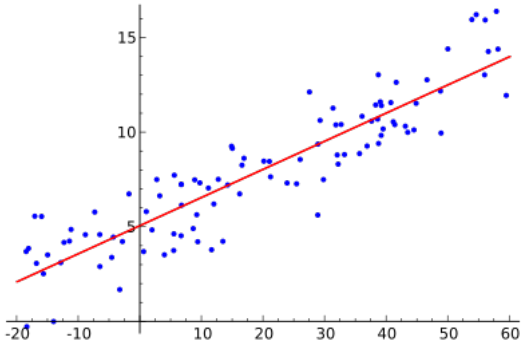
Spark Context thực thi các nhiệm vụ trong mỗi người thực thi.

2.2 Random forest

Random Forests là thuật toán học có giám sát (supervised learning). Nó có thể được sử dụng cho cả phân lớp và hồi quy. Nó cũng là thuật toán linh hoạt và dễ sử dụng nhất. Được hiểu như là một khu rừng bao gồm cây cối. Người ta nói rằng càng có nhiều cây thì rừng càng mạnh. Random forests tạo ra cây quyết định trên các mẫu dữ liệu được chọn ngẫu nhiên, được dự đoán từ mỗi cây và chọn giải pháp tốt nhất bằng cách bỏ phiếu. Nó cũng cung cấp một chỉ báo khá tốt về tầm quan trọng của tính năng. Random forests có nhiều ứng dụng, chẳng hạn như công cụ đề xuất, phân loại hình ảnh và lựa chọn tính năng. Tuy nhiên đến nay nó vẫn là mô hình hộp đen. Chúng tôi nghĩ nó sẽ phù hợp 2 bài toán Regression và Classification mà chúng tôi xây dựng.

2.3 Linear Regression

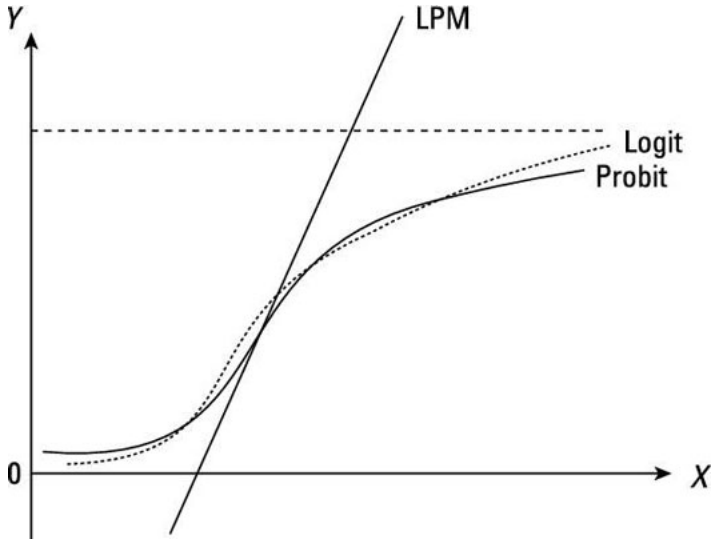
Là phương pháp hồi quy, là cơ sở cho nhiều mô hình học sâu sau này.sau khi qua bước hồi quy tuyến tính. Với việc đi tìm công thức Y=Ax+B với A và B là kết quả dự đoán của mô hình sau khi trainning. Mô hình dự đoán tham số Y từ Xtest có sẵn. Mô hình này được sử dụng rộng rãi vì độ thông dụng và có sự thông dụng của nó. Pyspark hỗ trợ mô hình trên ở dạng thư viên ml. Vì vậy, chúng tôi đã áp dụng Mô hình hồi quy logistic trên bộ dữ liệu trên. Hình 2



Hình 2: Mô hình Linrear Regression

2.4 Logistic Regression

Là phương pháp phân loại nhị phân, là cơ sở cho nhiều mô hình học sâu sau này.sau khi qua bước hồi quy tuyến tính. sẽ xác định bằng hàm Sigmoid. Mô hình này được sử dụng rộng rãi trong nhiều nghiên cứu về phân loại dữ liệu như phân loại chứng khoán, hình ảnh, NLP,... Pyspark hỗ trợ mô hình trên ở dạng thư viên ml. Vì vậy, chúng tôi đã áp dụng Mô hình hồi quy logistic trên bộ dữ liệu trên.Hình **??**:



Hình 3: Mô hình Linrear Regression

2.5 RMSE

Lỗi trung bình bình phương (RMSE) dùng để đánh giá mô hình hồi quy là độ lệch chuẩn của phần dư (lỗi dự đoán). Phần dư là thước đo khoảng cách từ các điểm dữ liệu đường hồi quy; RMSE là thước đo mức độ lan truyền của những phần dư này. Nói cách khác, nó cho bạn biết mức độ tập trung của dữ liệu xung quanh dòng phù hợp nhất . Lỗi bình phương trung bình thường được sử dụng trong khí hậu học, dự báo và phân tích hồi quy để xác minh kết quả thí nghiệm.Lỗi trung bình bình phương gốc (RMSE) là thước đo mức độ hiệu quả của mô hình của bạn. Nó thực hiện điều này bằng cách đo sự khác biệt giữa các giá trị dự đoán và giá trị thực tế . R-MSE càng nhỏ tức là sai số càng bé thì mức độ ước lượng cho thấy độ tin cậy của mô hình có thể đạt cao nhất.

2.6 R Square

Công thức tính hệ số R bình phương dùng để đánh giá mô hình hồi quy xuất phát từ ý tưởng: toàn bộ sự biến thiên của biến phụ thuộc được chia làm hai phần: phần biến thiên do hồi quy và phần biến thiên không do hồi quy(còn gọi là phần dư). Giá trị R bình phương dao động từ 0 đến 1. R bình phương càng gần 1 thì mô hình đã xây dựng càng phù hợp với bộ dữ liệu dùng chạy hồi quy. R bình phương càng gần 0 thì mô hình đã xây dựng càng kém phù hợp với bộ dữ liệu dùng chạy hồi quy. Trường hợp đặt biệt, phương trình hồi quy đơn biến (chỉ có 1 biến độc lập) thì R2 chính là bình phương của hệ số tương quan r giữa hai biến đó.

2.7 Accuracy

Khi xây dựng mô hình phân loại chúng ta sẽ muốn biết một cách khái quát tỷ lệ các trường hợp được dự báo đúng trên tổng số các trường hợp là bao nhiêu. Tỷ lệ đó được gọi là độ chính xác. Độ chính xác giúp ta đánh giá hiệu quả dự báo của mô hình trên một bộ dữ liệu. Độ chính xác càng cao thì mô hình của chúng ta càng chuẩn xác. Với công thức là (True Positive+True Negative)/ Total sample.

3 Hướng Tiếp Cận

3.1 Bộ Dữ Liệu

Như đã nói, Epicurious là bộ dữ liệu được thu thập bởi trang web cùng tên đã thu thập các dữ liệu về món ăn và các dữ liệu liên quan đến món ăn đó. Bộ dữ liệu gồm có 20057 điểm dữ liệu và trong đó có tổng cộng 680 nhãn bao gồm: Title (Tên món ăn) – string, Rating (Đánh giá món ăn) – float64, calories (lượng calori) – float64, protein (lượng protein) – float64, fat (lượng chất béo) – float64, sodium (lượng kiềm) – float64, và hơn 670 nhãn còn gồm thành phần thức ăn (beef, been, v.v), loại món ăn (dessert, v.v), địa điểm (London, v.v), loại bữa ăn(breakfast, v.v), cách chế biến, sử dụng lò nướng,.ect.. – bolean(1,0).

Hình 4: Bộ dữ liệu Epicurious về món ăn

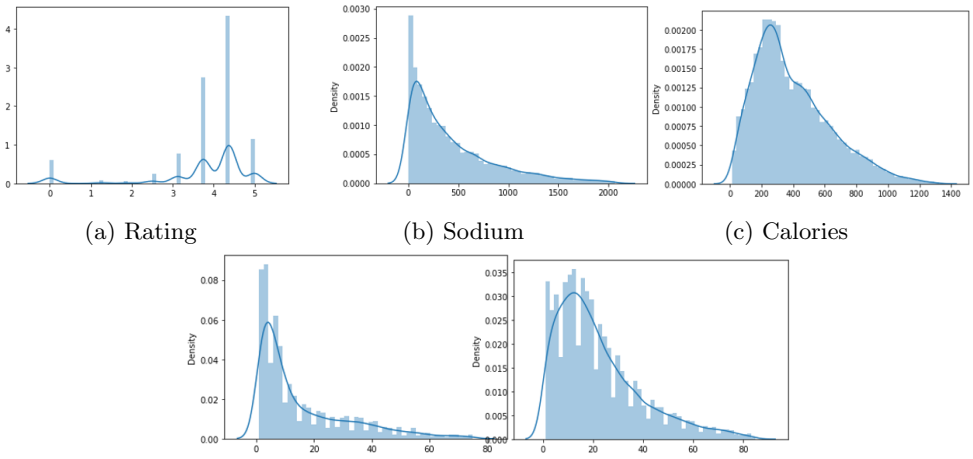
3.2 Tiền xử lý

Null	4193
Impossible values	1426
Duplicate	24
Outlier (> 95%)	2727
Data	11687
Total	20057

Hình 5: Thống kê Bộ dữ liệu Epicurious

Theo thống kê như hình 5, bộ dữ liệu của chúng ta có tổng cộng là 20057 với 680 cột là thành phần dinh dưỡng, tên món ăn, thành phần món ăn địa điểm,.... nhưng sau xử lý thực tế dữ liệu, chúng tôi nhận được 11657(Eraser-Null),11465(ADDvalues) điểm dữ liệu qua các quy trình thành phần như Null, impossible values, duplicate và outlier sau khi phân tích ra làm hai hướng, đó là Erase values và Add values. Dữ liệu được biểu diễn như hình (6)

Erase values: Với thực nghiệm này này, về căn bản đó là xoá tất cả những yếu tố bị rỗng, dư thừa vì những cột dữ liệu có null là những dữ liệu cần thiết thuận tiện cho chúng tôi phân tích và xây dựng mô hình và lặp. Impossible values (Giá trị bất thường) cũng là một vấn đề mà nhóm khá cân nhắc, khá nhiều bài báo và trang web đề không thể có một số thành phần dinh dưỡng nào trong nhóm



(a) Rating (b) Sodium (c) Calories

(d) Protein (e) Fat

Hình 6: Biểu diễn các thuộc tính sau khi erase values

4 loại dinh dưỡng chính có giá trị bằng 0 cho tất cả thực phẩm nên xóa các trường hợp như calories = 0, protein = 0, sodium = 0, fat = 0. Việc tồn tại các duplicate trong bộ dữ liệu cũng sẽ được nhóm loại bỏ hoàn toàn bởi nó sẽ làm cho dữ liệu không nhất quán.Trong bộ dữ liệu có vấn đề khá là khá quan tâm là Outlier, dữ liệu có nhiều điểm bất thường khi trực quan dữ liệu ngoài khoảng 95%. quyết định bỏ các giá trị dữ liệu có các thuộc tính calories $>$ 1331, protein $>$ 76, sodium $>$ 2063, fat $>$ 86. Sau các quy trình trên còn 11687 điểm dữ liệu. Sau đó, nhóm tiến hành EDA như hình (7) để xét sự tương quan giữa các thuộc tính có ảnh hưởng trong bài:

```
coreclation cua calories va fat la: 0.8489857878710382
coreclation cua calories va protein la: 0.6565678268551042
coreclation cua calories va sodium la: 0.42867463506314335
coreclation cua fat va sodium la: 0.34782700637822705
coreclation cua fat va protein la: 0.5489835043792513
coreclation cua protein va sodium la: 0.5253039018674166
coreclation cua rating va sodium la: 0.0774868963391573
coreclation cua rating va calories la: 0.10530933569401878
coreclation cua rating va fat la: 0.10880205265189782
coreclation cua rating va protein la: 0.09487155738563739
```

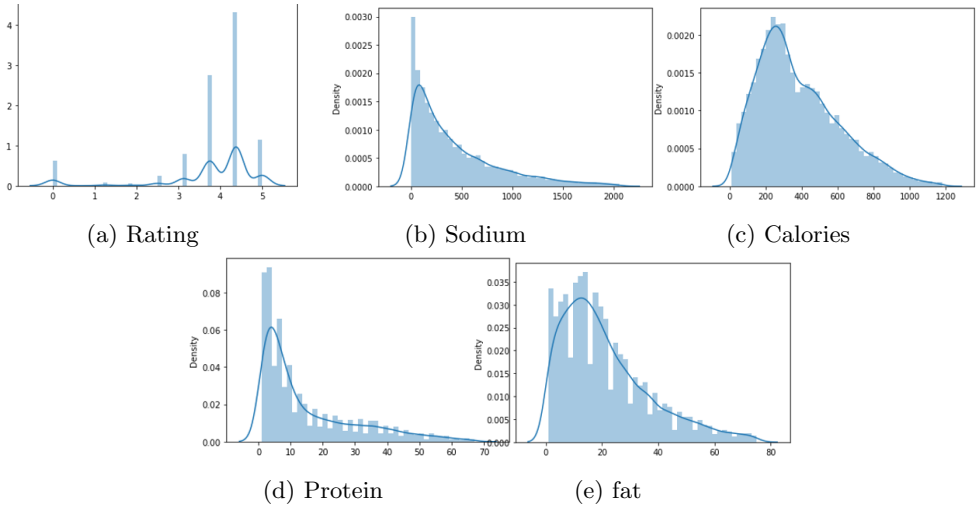
Hình 7: Sự tương quan của các thuộc tính của Erase values

Theo hình (7), ta thấy sự tương quan giữa calories với các chất dinh dưỡng là khá cao, đặc biệt là giữa calories với fat: 0.84, trong khi sự tương quan giữa rating với các chất dinh dưỡng lại khá thấp, đặc biệt là giữa rating với sodium:

0.77. Còn lại, sự tương quan giữa fat và protein với các chất dinh dưỡng khác là khá tốt.

Impossible values	1801
Duplicate	16
Outlier (> 95%)	6775
Data	11465
Total	20057

Hình 8: Thống kê bộ dữ liệu sau khi Add Valuess



(a) Rating (b) Sodium (c) Calories

(d) Protein (e) fat

Hình 9: Biểu diễn các thuộc tính sau khi add values

Add values: Với thực nghiệm này này, về căn bản đó là thêm giá trị bị rỗng, dư thừa vì những cột dữ liệu có null là những dữ liệu cần thiết cho chúng tôi xây dựng mô hình và lặp. Trước khi add values, lấy tập erase sau khi xoá khoảng 2000/10000 để test vì đây được xem là dữ liệu chuẩn như ban đầu. Rồi sau đó tiến hành add value với giá trị ở median=0.5. Sau đó, có khá nhiều điểm dữ liệu giống nhau nên chúng tôi đã loại bỏ nó bằng loại bỏ duplicate. Impossible values (Giá trị bất thường) cũng là vấn đề mà nhóm đã cân nhắc loại bỏ một số thành

phần dinh dưỡngtrong nhóm 4 loại dinh dưỡng chính có giá trị bằng 0 cho tất cả thực phẩm nên xóa các trường hợp như calories = 0, protein = 0, sodium = 0, fat = 0. Trong bộ dữ liệu có vấn đề khá là khá quan tâm là Outlier, dữ liệu có nhiều điểm bất thường khi trực quan dữ liệu ngoài khoảng 95%. quyết định bỏ các giá trị dữ liệu có các thuộc tính calories $>$ 1186, protein $>$ 68, sodium $>$ 2058, fat $>$ 76. Cuối cùng bài toán chúng tôi còn 11465 điểm dữ liệu như hình (9).Sau đó, nhóm tiến hành EDA như hình (10) để xét sự tương quan giữa các thuộc tính có ảnh hưởng trong bài:

```
coreclation cua calories va fat la: 0.837299285125589
coreclation cua calories va protein la: 0.6375134190911126
coreclation cua calories va sodium la: 0.4206939796890131
coreclation cua fat va sodium la: 0.3427457634732143
coreclation cua fat va protein la: 0.5274882673349054
coreclation cua protein va sodium la: 0.5297271893792865
coreclation cua rating va sodium la: 0.0786433397061752
coreclation cua rating va calories la: 0.10872941859073906
coreclation cua rating va fat la: 0.11068943146805965
coreclation cua rating va protein la: 0.09555357709507022
```

Hình 10: Sự tương quan của các thuộc tính của Add values

Theo hình (10), ta thấy sự tương quan giữa calories với các chất dinh dưỡng là khá cao, đặc biệt là giữa calories với fat: 0.83, trong khi sự tương quan giữa rating với các chất dinh dưỡng lại khá thấp, đặc biệt là giữa rating với sodium: 0.78. Còn lại, sự tương quan giữa fat và protein với các chất dinh dưỡng khác là khá tốt.

4 Kết Quả Thực Nghiệm Và Đánh giá

Sau khi sử dụng các thuật toán để giải quyết hai bài toán trên vào bộ dữ liệu Epicurious, chúng tôi thu được kết quả như sau:

4.1 Bài toán 1: Dự đoán Calories

Theo kết quả như hình (11), test lấy từ erasernull 2000 điểm dữ liệu để đánh giá cho 2 bộtại Erase Values ở Linear Regression cho kết quả thấp hơn ở RMSE Train = 114.27 và RMSE Test = 115.22 nhưng cho Rsquared Error tốt hơn so với Random Forest là Rsquared Error Train = 0.77 và Rsquared Error Test = 0.78. Trong khi ở Random Forest thì cho kết quả RMSE Train = 18034.5 và RMSE Test = 3601.77 và cho Rsquared Error Rsquared Error Train = 0.83 và Rsquared Error Test = 0.84. Và tại Add Values cũng như vậy, Linear Regression cho kết quả tốt hơn so với Random Forest RMSE Train = 114.15 và RMSE Test = 108.16 nhưng cho Rsquared Error tốt hơn so với Random Forest là Rsquared Error Train = 0.75 và Rsquared Error Test = 0.77. Trong khi ở Random Forest

	A. Erase values				B. Add values			
Mô hình	**Rsquared Error Train**	**Rsquared Error Test**	**RMSE Train**	**RMSE Test**	**Rsquared Error Train**	**Rsquared Error Test**	**RMSE Train**	**RMSE Test**
Linear Regression()	**0.77**	**0.78**	**114.27**	**115.22**	**0.75**	**0.77**	**114.15**	**108.16**
Random Forest(100 trees)	**0.83**	**0.84**	**18034.5**	**3601.77**	**0.84**	**0.852**	**15070**	**2300**

Hình 11: Kết quả dự đoán Calories

thì cho kết quả RMSE Train = 15070 và RMSE Test = 2300 và cho Rsquared Error Rsquared Error Train = 0.84 và Rsquared Error Test = 0.852.

4.2 Bài toán 2: Phân loại món ăn Kosher

Biến phụ thuộc		Mô hình	Accuracy test	Accuracy train
Kosher	A. Erase values	Logistic Regression()	0.802	0.90
	B. Add values		0.904	0.95
	A. Erase values	Random Forest (100tree)	0.76	0.93
	B. Add values		0.85	0.96

Hình 12: Kết quả phân loại món ăn Kosher

Theo kết quả như hình (12), với việc test lấy từ erasernull 2000 điểm dữ liệu để đánh giá cho 2 bộ tại mô hình Logistic Regression, Add Values cho kết quả tốt hơn là Accuracy test = 0.904 và Accuracy train = 0.95 trong khi Erase Values cho kết quả là Accuracy test = 0.802 và Accuracy train = 0.9. Và tại mô hình

Random Forest cũng như vậy, Add Values cho kết quả tốt hơn là Accuracy test = 0.85 và Accuracy train = 0.96 trong khi Erase Values cho kết quả là Accuracy test = 0.76 và Accuracy train = 0.93.

5 Kết Luận Và Hướng Phát Triển

Trong bài báo này, chúng tôi đã trình bày chi tiết quá trình nghiên cứu, thực nghiệm và kết quả của hai bài toán đó phân tích bộ dữ liệu để dự đoán lượng calories trong món ăn và bài toán phân loại món ăn Kosher. Với những quả đã được tương đối tốt ở mô hình được chọn, ta có thể thấy được bộ dữ liệu có tính thực tiễn rất cao qua hai bài toán. Và tương lai có tiềm năng để phân tích, phát triển và ứng dụng vào thực tế rất cao không chỉ từ hai bài toán trên mà còn cả những dữ liệu tiềm năng có trong bộ dữ liệu như việc có thể xây dựng phần mềm đánh giá chuẩn thức ăn kosher, healthy trong tương lai vào một ngày không xa.. Trong tương lai, chúng tôi sẽ tiếp tục nghiên cứu, thực nghiện trên nhiều mô hình khác để cải thiện kết quả bài toán. Một trong những hướng phát triển chúng tôi sẽ nghiên cứu sắp tới là xây dựng app ứng dụng để phân loại món ăn một cách đa dạng nhất để phục vụ cộng đồng thế giới.

Tài liệu

1. ick Pentreath, Machine Learning with Spark, Beijing, pp. 1-140, 2015.
2. pache Hive. http://hadoop.apache.org/hive
3. Hadoop Map/Reduce tutorial. http://hadoop.apache.org/ common/docs/r0.20.0/mapred tutorial.html.
4. Raswitha Bandi, J Amudhavel, R Karthik, “Machine Learning with PySpark - Review” in Indonesian Journal of Electrical Engineering and Computer Science, 2018. http://ijeecs.iaescore.com/index.php/IJEECS/article/view/11563
5. https://www.kaggle.com/hugodarwood/epirecipes/code
6. https://spark.apache.org/docs/latest/api/python/?fbclid=IwAR1ejP-gxVWMBlU8jKdkFqeHhao78IpAT-uhwozHeOoVtp1bfz0REaqEETY