MÔ HÌNH PHÂN LOẠI NGUY CƠ CHÁY RỪNG

Thái Tường An, Nguyễn Thúy Thùy Dương

Khoa Công nghệ thông tin, Đại học Nông Lâm TP.HCM

Tóm tắt: Nghiên cứu xây dựng mô hình phân loại nguy cơ cháy rừng dựa trên dữ liệu khí hậu thu thập từ hai khu vực tại Algeria. Các thuật toán Logistic Regression, Random Forest và Multi-Layer Perceptron (MLP) được áp dụng cùng các kỹ thuật tiền xử lý và tăng cường dữ liệu như thêm nhiễu Gaussian, Scaling, và sử dụng mô hình CTGAN. Kết quả cho thấy Random Forest đạt độ chính xác cao nhất với 99%, trong khi Logistic Regression và MLP đạt lần lượt 95% và 96%. Nghiên cứu nhấn mạnh vai trò của việc xử lý dữ liệu và lựa chọn mô hình trong việc phát triển hệ thống cảnh báo cháy rừng, đồng thời gợi mở hướng tích hợp dữ liệu thời gian thực và mô hình học sâu tiên tiến trong tương lai.

1. Giới Thiệu

Cháy rừng là một hiện tượng nguy hiểm, gây ra nhiều hậu quả nghiêm trọng đối với hệ sinh thái, các loài sinh vật và đời sống con người. Những tổn thất như thiệt hại môi trường, mất mát đa dạng sinh học gây ảnh hưởng tiêu cực đến sức khỏe con người và kinh tế xã hội. Trước tình hình này, việc phân loại nguy cơ cháy rừng đóng vai trò rất quan trọng, giúp xác định các khu vực dễ xảy ra cháy, từ đó hỗ trợ công tác phòng chống và giảm thiểu thiệt hại một cách hiệu quả. Đề tài áp dụng quy trình gồm ba giai đoạn chính để giải quyết bài toán. Đầu tiên là tăng cường và làm sạch dữ liệu nhằm đảm bảo độ chính xác và tính đại diện của tập dữ liệu. Tiếp đến, các đặc trưng quan trọng được trích xuất, chuẩn hóa, giúp mô hình học được các yếu tố tác động đến nguy cơ cháy. Cuối cùng, các thuật toán học máy, bao gồm Logistic Regression, Random Forest và Multi-Layer Perceptron (MLP) được sử dụng để huấn luyện và đánh giá hiệu quả mô hình.

Mô hình phân loại nguy cơ cháy rừng mang tính ứng dụng cao, hỗ trợ xây dựng các hệ thống cảnh báo sớm, giúp các cơ quan chức năng và cộng đồng có thể chuẩn bị và ứng phó kịp thời. Ngoài ra, kết quả nghiên cứu góp phần tối ưu hóa việc quản lý tài nguyên rừng, bảo vệ môi trường, và giảm thiểu tổn thất kinh tế do cháy rừng gây ra. Đây là một giải pháp có ý nghĩa thực tiễn, hướng đến phát triển bền vững và bảo tồn thiên nhiên.

2. Các Công Trình Liên Quan

Trong bài báo [1], nhóm tác giả đã tích hợp bộ phân loại Decision Tree (DT) vào kiến trúc của các nút cảm biến thông minh, sử dụng ba yếu tố khí tượng quan trọng ảnh hưởng đến nguy cơ cháy rừng gồm: nhiệt độ, độ ẩm tương đối (RH), và tốc độ gió. Từ đó, mô hình dự đoán khả năng cháy rừng được xây dựng nhằm tự động hóa và nâng cao tính thông minh trong việc dự đoán cháy rừng. Mô hình được phát triển và đánh giá trên nền tảng Weka, với việc phân loại các mẫu dữ liệu cháy rừng thành hai lớp, dựa trên hai thành phần quan trọng của hệ thống FWI (Fire Weather Index): FWI và FFMC (Fine Fuel Moisture Code) – đại diện cho độ ẩm của nhiên liệu mịn và khả năng bắt cháy. Nhóm tác giả đã so sánh hiệu năng của DT đơn giản với các phương pháp cải tiến của nó, bao gồm Boosting DT (AdaBoostM1), Bagging DT, và Random Forest. Kết quả cho thấy DT

đơn giản vượt trội hơn, đạt độ chính xác cao nhất so với các thuật toán còn lại. Đây là lý do mô hình này được lựa chọn để triển khai trong hệ thống. Với độ chính xác khoảng 82.92%, mô hình dự đoán cháy rừng dựa trên DT đơn giản không chỉ chứng minh tính hiệu quả trong việc dự đoán mà còn có khả năng triển khai hiệu quả trên phần cứng. Kết quả này mở ra tiềm năng xây dựng hệ thống dự báo cháy rừng tự động với độ chính xác cao, đáp ứng được yêu cầu thực tiễn.

Với mục tiêu xem xét mức độ thiệt hại của cháy rừng theo địa điểm, bài báo [2] tiến hành phân loại và nhận dạng cháy bề mặt và cháy tán cây. Ở bài báo này, tác giả phát triển một mô hình phân loại và phát hiện cháy rừng dựa trên YOLOv5, sử dụng dữ liệu hình ảnh. Hàm CIoU Loss được thay thế bằng SIoU Loss giúp mô hình hội tụ nhanh hơn trong quá trình huấn luyện. Cơ chế chú ý CBAM được thêm vào để cải thiện độ chính xác nhận diện nhờ vào sự kết hợp giữa chú ý kênh và không gian. Cuối cùng, lớp PANet được cải tiến thành BiFPN để tăng cường khả năng kết hợp và lọc đặc điểm, giúp phát hiện cháy rừng ở nhiều chiều khác nhau và giảm thiểu mất mát đặc điểm. Tuy nhiên, tác giả cũng chỉ ra một số hạn chế trong bài báo. Các mô hình phát hiện và cơ chế chú ý khác nhau mang lại hiệu quả phát hiện khác nhau cho các loại cháy rừng có phân bố và hình dạng mục tiêu khác nhau. Cơ chế CBAM cải thiện chung cho tất cả các loại cháy, trong khi GAM hiệu quả hơn đối với cháy tán cây. Để tối ưu, cần tìm kiếm và tích hợp các mô hình phát hiện phù hợp cho từng loại cháy. Bên cạnh đó, cháy mặt đất thường chuyển thành cháy tán cây khi cường độ tăng, nên việc thu thập thêm hình ảnh ở giai đoạn đầu của đám cháy sẽ giúp mở rộng và cải thiện độ tin cậy của bộ dữ liệu. Kết quả của nghiên cứu đã chỉ ra rằng mô hình phân loại và nhận diện cháy rừng đề xuất có triển vọng ứng dụng tốt trong thực tế.

Nghiên cứu [3] tập trung vào phương pháp phát hiện cháy rừng dựa trên học sâu nhằm phát hiện sớm và ngăn ngừa các thảm họa lớn. Nghiên cứu này đề xuất một mô hình học sâu mới có tên là FFireNet, được thiết kế để giải quyết bài toán phát hiện và phân loại hình ảnh cháy rừng. Mô hình FFireNet sử dụng nền tảng tích chập được huấn luyện trước của MobileNetV2, đồng thời bổ sung các lớp kết nối đầy đủ để giải quyết bài toán phân loại cháy rừng. Phương pháp này cho phép FFireNet tận dụng khả năng của mô hình tiền huấn luyện trong việc trích xuất đặc trưng, đồng thời linh hoạt trong việc học các mẫu dữ liệu mới để phân loại hình ảnh cháy và không cháy. Trong nghiên cứu, các tác giả không chỉ tập trung vào việc phân loại các khu vực hoang dã như cây bụi hay nông trại như ở các nghiên cứu trước, mà còn chú trọng đến môi trường rừng. Kết quả đạt được cho thấy các chỉ số đánh giá của phương pháp đều vượt 97%, đồng thời cao hơn so với nhiều mô hình CNN khác như InceptionV3, Xception, NASNetMobile và ResNet152V2. Những kết quả này góp phần xây dựng các hệ thống phát hiện cháy rừng thông minh và hiệu quả trong bối cảnh biến đổi khí hậu.

3. Phát Biểu Bài Toán

3.1. Bài toán

Bài toán phân loại nguy cơ cháy rừng nhằm mục đích đưa ra kết luận một khu vực có xảy ra cháy rừng hay không dựa trên các yếu tố môi trường. Đầu vào của mô hình bao gồm các dữ liệu về các đặc trưng môi trường, như nhiệt độ, độ ẩm không khí, và các yếu tố khác ảnh hưởng đến cháy

rừng. Kết quả đầu ra của mô hình sẽ là một phân loại "cháy" hoặc "không cháy", giúp đưa ra cảnh báo sớm về cháy rừng trong khu vực.

3.2. Thuật toán

3.2.1. Logistic Regression

Logistic Regression là một thuật toán học máy có giám sát được sử dụng trong các bài toán phân loại, đặc biệt là phân loại nhị phân. Mục tiêu của thuật toán là xác định xác suất một đối tượng thuộc về một trong hai lớp phân loại. Thuật toán sử dụng một mô hình tuyến tính với dữ liệu đầu vào để ước tính xác suất. Mô hình có thể được biểu diễn như sau:

$$z = w_1x_1 + w_2x_2 + \ldots + w_nx_n + b$$

Trong đó:

- w_1, w_n là các trọng số cần học
- x_1, x_n là các đặc trưng đầu vào
- b là bias

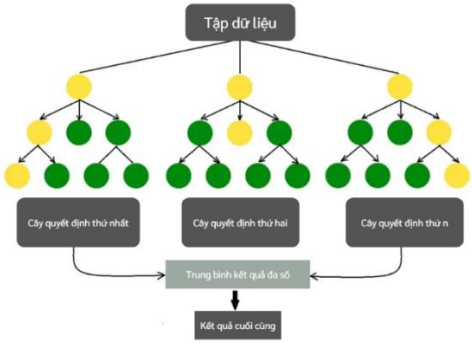
Kết quả z từ mô hình tuyến tính được đưa qua hàm sigmoid để chuyển đổi giá trị này thành xác suất. Hàm sigmoid có công thức:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Kết quả từ hàm sigmoid có giá trị từ [0,1] giúp mô hình dự đoán xác suất của một đầu vào. Nếu xác suất nhỏ hơn 0.5, mô hình sẽ phân loại đối tượng vào lớp 0. Nếu xác suất lớn hơn hoặc bằng 0.5, mô hình phân loại vào lớp 1. Sau khi đưa ra kết quả, để tối ưu hóa mô hình, thuật toán sử dụng hàm mất mát (hàm loss) để tính toán sự khác biệt giữa giá trị dự đoán và giá trị thực tế. Mục tiêu là tối thiểu hóa hàm mất mát để cải thiện độ chính xác của mô hình trong việc phân loại.

3.2.2. Random Forest

Random Forest là một thuật toán học máy có giám sát, chủ yếu được sử dụng cho bài toán phân loại. Mô hình này bao gồm nhiều cây quyết định, và kết quả cuối cùng được tổng hợp từ các dự đoán của các cây quyết định đó.



Hình 3.1. Xây dựng thuật toán Random Forest

Đầu tiên, Random Forest sử dụng kỹ thuật Bootstrapping để lấy ngẫu nhiên một tập con n mẫu từ bộ dữ liệu gốc. Điều này có nghĩa là mỗi mẫu có thể được lấy nhiều lần, và do đó, tập con này có thể chứa các mẫu dữ liệu bị trùng lặp. Khi xây dựng mỗi cây quyết định, thay vì xét tất cả các thuộc tính có sẵn, Random Forest chỉ chọn ngẫu nhiên một tập con các thuộc tính. Điều này giúp giảm sự phụ thuộc vào các thuộc tính mạnh mẽ và tạo ra các cây quyết định độc lập hơn, từ đó làm giảm sự tương quan giữa các cây trong rừng. Với mỗi tập con dữ liệu và các thuộc tính đã được chọn ngẫu nhiên, thuật toán Decision Tree sẽ được áp dụng để xây dựng các cây quyết định. Mỗi cây quyết định sẽ được huấn luyện độc lập và có thể có độ chính xác cũng như cấu trúc khác nhau, nhờ vào sự ngẫu nhiên trong quá trình xây dựng.

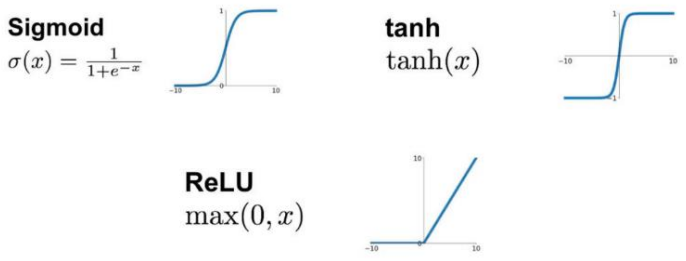
Việc kết hợp ngẫu nhiên của nhiều cây quyết định giúp Random Forest tránh hiện tượng overfitting, đồng thời tối ưu hóa khả năng dự đoán trong các bài toán thực tế, đặc biệt là khi dữ liệu có tính không đồng nhất và phức tạp.

3.2.3. Multi-Layer Perceptron

MLP là một mạng nơ-ron nhân tạo được thiết kế để ánh xạ dữ liệu đầu vào thành dữ liệu đầu ra phù hợp. Cấu trúc của MLP gồm ba lớp chính: lớp đầu vào, lớp ẩn và lớp đầu ra, trong đó các lớp được kết nối đầy đủ và thực hiện tính toán thông qua các hàm kích hoạt.

- **Lớp đầu vào:** Là lớp gồm các nơ-ron tiếp nhận dữ liệu đầu vào ban đầu. Mỗi nơ-ron biểu diễn một tính năng hoặc chiều của dữ liệu đầu vào. Số lượng nơ-ron trong lớp đầu vào được xác định bởi chiều của dữ liệu đầu vào.
- **Lớp ẩn:** Gồm một hoặc nhiều lớp, nhận đầu vào từ lớp trước, tính toán kết quả qua hàm kích hoạt và truyền sang lớp tiếp theo.
- **Lớp đầu ra:** Là lớp cung cấp kết quả cuối cùng của mạng. Số lượng nơ-ron ở lớp này tương ứng với nhiệm vụ cần giải quyết, ví dụ phân loại nhị phân hoặc đa lớp.

Hàm kích hoạt đóng vai trò chuyển đổi dữ liệu đầu ra của các nơ-ron, giúp mạng học được các mối quan hệ phi tuyến. Hàm ReLU (Rectified Linear Unit) đưa các giá trị âm về 0, giữ nguyên giá trị dương, thường được dùng ở các lớp ẩn để tăng hiệu quả tính toán. Hàm tanh (Hyperbolic Tangent) là một hàm kích hoạt phi tuyến, đưa giá trị đầu ra vào khoảng [-1,1], thường được sử dụng khi dữ liệu cần được chuẩn hóa với giá trị trung bình quanh 0, giúp tăng tốc độ học và cải thiện khả năng hội tụ của mô hình. Hàm sigmoid thường được dùng cho lớp đầu ra để dự đoán các giá trị nhị phân.



Hình 3.2. Các hàm kích hoạt

Quá trình học của MLP được thực hiện qua hai giai đoạn chính. Đầu tiên, MLP áp dụng lan truyền xuôi (Forward Propagation) để tính toán kết quả của dữ liệu được truyền qua các lớp dựa trên trọng số và hàm kích hoạt. Sau khi có kết quả đầu ra, hàm mất mát được tính toán để đo lường sự khác biệt giữa đầu ra dự đoán của mô hình và giá trị thực tế, từ đó giúp đánh giá hiệu quả của mô hình. Dựa trên sự chênh lệch này, lan truyền ngược (Backpropagation) được áp dụng để cập nhật lại trọng số, điều này giúp mô hình học tốt hơn và cải thiện kết quả sau mỗi vòng lặp huấn luyện.

3.2.4. Biện pháp tăng cường dữ liệu

Gaussian Noise

Phương pháp thêm nhiễu Gaussian được sử dụng để tăng cường tính đa dạng của dữ liệu bằng cách thêm các biến thể nhỏ vào các đặc trưng số. Nhiễu Gaussian được lấy từ phân phối chuẩn, sau đó được cộng vào dữ liệu gốc nhằm tạo ra các phiên bản dữ liệu mới. Phương pháp này giúp mô hình học máy không chỉ tập trung vào các mẫu dữ liệu cụ thể mà còn mở rộng khả năng khái quát hóa, đồng thời giảm thiểu nguy cơ overfitting. Việc thêm nhiễu Gaussian không làm thay đổi các đặc điểm cốt lõi của dữ liệu, đảm bảo rằng dữ liệu sau tăng cường vẫn phản ánh đúng bản chất của dữ liệu gốc.

Scaling

Scaling là phương pháp điều chỉnh giá trị của các đặc trưng bằng cách nhân hoặc chia chúng với các hệ số ngẫu nhiên nhỏ. Điều này tạo ra các phiên bản dữ liệu mới gần với phân phối gốc nhưng có sự biến thiên nhỏ, từ đó tăng cường tính đa dạng của tập dữ liệu. Phương pháp này thường được áp dụng khi các đặc trưng đầu vào có sự chênh lệch lớn về giá trị, giúp làm mượt các giá trị đầu vào và cải thiện hiệu quả huấn luyện của mô hình. Scaling không chỉ giúp tăng số lượng dữ liệu mà còn đảm bảo rằng dữ liệu mới được tạo ra vẫn duy trì được mối quan hệ và cấu trúc của dữ liệu ban đầu.

CTGAN (Conditional Tabular GAN)

CTGAN (Conditional Tabular GAN) là một mạng GAN được thiết kế để tạo dữ liệu bảng nhân tạo, đặc biệt hiệu quả trong việc xử lý các tập dữ liệu mất cân bằng. CTGAN bao gồm hai thành phần chính: Generator (G) và Discriminator (D). Generator nhận đầu vào là nhiễu ngẫu nhiên và điều kiện từ dữ liệu gốc để tạo ra dữ liệu nhân tạo, trong khi Discriminator phân biệt giữa dữ liệu thực và nhân tạo. Quá trình huấn luyện diễn ra theo cơ chế đối kháng, giúp Generator cải thiện khả năng tạo dữ liệu giống thực tế. Đặc biệt, việc tích hợp điều kiện trong quá trình tạo dữ liệu giúp CTGAN tái tạo các đặc điểm cụ thể của dữ liệu gốc, từ đó cân bằng dữ liệu giữa các lớp, tăng tính đa dạng và cải thiện hiệu năng của các mô hình học máy.

4. Thực Nghiệm

4.1. Dữ Liệu

Mô Tả Dữ Liệu

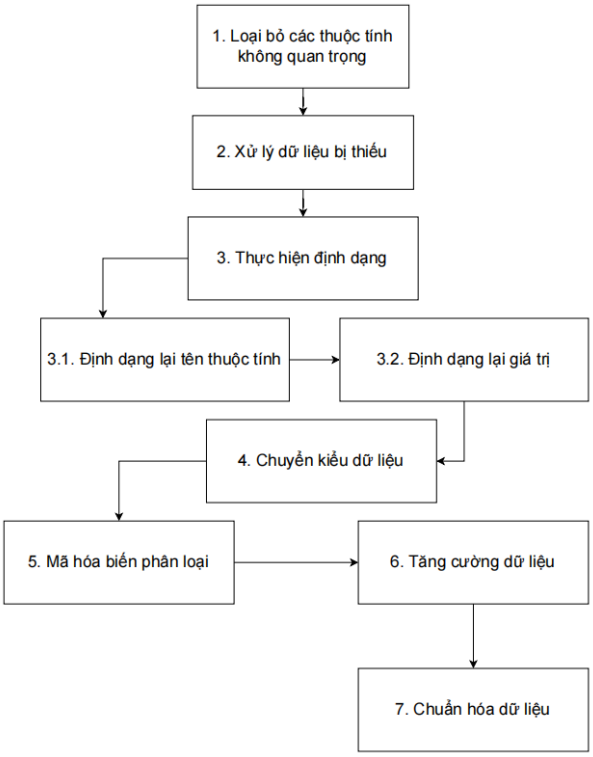
Algerian Forest Fires là một tập dữ liệu đa biến gồm 244 mẫu, được chia đều giữa hai khu vực: Bejaia (122 mẫu) ở đông bắc Algeria và Sidi Bel-Abbes (122 mẫu) ở tây bắc Algeria. Về kết quả phân loại, 244 mẫu được phân thành hai lớp: **'fire'** (có cháy) với 138 mẫu và **'not fire'** (không cháy) với 106 mẫu. Dataset này cung cấp một cơ sở quan trọng để nghiên cứu và phân loại nguy cơ cháy rừng dựa trên các yếu tố khí hậu và môi trường.

Dataset được lưu trữ trong tệp CSV, bao gồm 13 đặc trưng đầu vào và 1 đặc trưng đầu ra (Classes).

Bảng 4.1.1. Bảng mô tả dữ liệu cho tập dữ liệu Algerian Forest Fires

Tên thuộc tính	Mô tả	Kiểu dữ liệu
day	Ngày thu thập dữ liệu	int64
month	Tháng thu thập dữ liệu	int64
year	Năm thu thập dữ liệu	int64
Temperature	Nhiệt độ không khí	int64
RH	Độ ẩm tương đối: Độ ẩm càng thấp càng dễ bắt lửa (Relative Humidity)	int64
Ws	Tốc độ gió (Wind Speed)	int64
Rain	Lượng mưa: Lượng mưa càng thấp càng làm tăng nguy cơ cháy	float64
FFMC	Chỉ số độ ẩm nhiên liệu mịn trên bề mặt rừng: FFMC càng thấp càng dễ cháy (Fine Fuel Moisture Code)	float64
DMC	Chỉ số độ ẩm nhiên liệu ở bề mặt sâu hơn: DMC càng thấp càng dễ cháy và kéo dài thời gian cháy (Duff Moisture Code)	float64
DC	Chỉ số hạn hán (Drought Code)	object
ISI	Chỉ số tốc độ lan cháy (Initial Spread Index)	float64
BUI	Chỉ số độ cháy (Buildup Index)	float64
FWI	Chỉ số nguy cơ cháy rừng: Kết hợp từ FFMC, DMC, DC, ISI, BUI (Fire Weather Index)	object
Classes	Nhãn phân loại (fire hoặc not fire)	object

Tiền Xử Lý Dữ Liệu



Hình 4.1.2. Quá trình tiền xử lý

1. **Loại bỏ các thuộc tính không quan trọng**

   Đề tài tập trung vào các yếu tố môi trường để huấn luyện mô hình phân loại trường hợp có cháy hay không cháy. Vì vậy, các cột liên quan đến thời gian thu thập dữ liệu sẽ được loại bỏ khỏi tập dữ liệu.

2. **Xử lý dữ liệu bị thiếu**

   Thuộc tính FWI xuất hiện một dòng có giá trị không xác định, do chỉ có một dòng bị thiếu nên có thể xóa dòng dữ liệu này mà không gây ảnh hưởng lớn đến tập dữ liệu.

3. **Thực hiện định dạng**

   Tên của cột RH và Ws trong tập dữ liệu xuất hiện khoảng trắng không cần thiết. Loại bỏ khoảng trắng này giúp việc truy xuất theo tên thuộc tính trong quá trình phân tích sẽ được thực hiện dễ dàng hơn. Tương tự, giá trị ‘fire’ và ‘not fire’ ở cột **Classes** xuất hiện khoảng trắng, vì vậy việc loại bỏ chúng là điều cần thiết để các giá trị được đồng nhất.

4. **Chuyển kiểu dữ liệu**

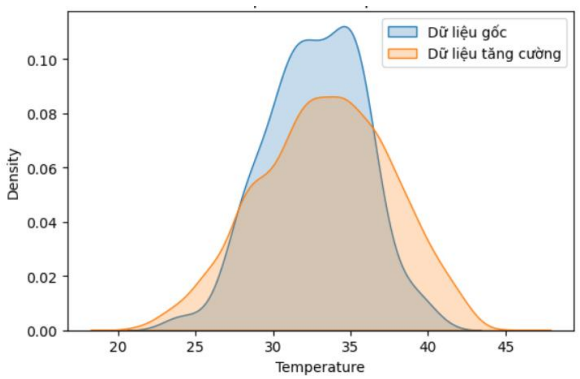
   Các giá trị ở cột DC và FWI được chuyển đổi thành kiểu dữ liệu số để có thể hỗ trợ quá trình phân tích ở các giai đoạn sau.

5. Mã hóa biến phân loại

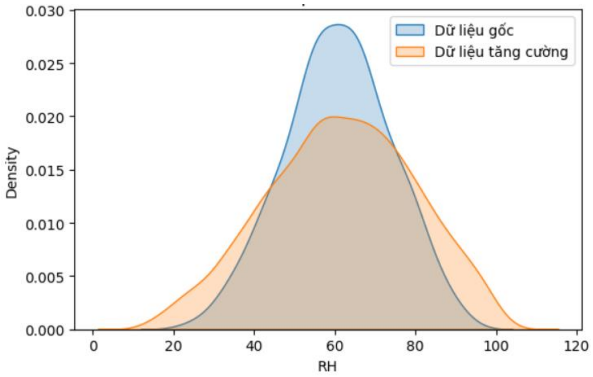
Hai giá trị ‘fire’ và ‘not fire’ được mã hóa thành 1 (fire) và 0 (not fire).

6. Tăng cường dữ liệu

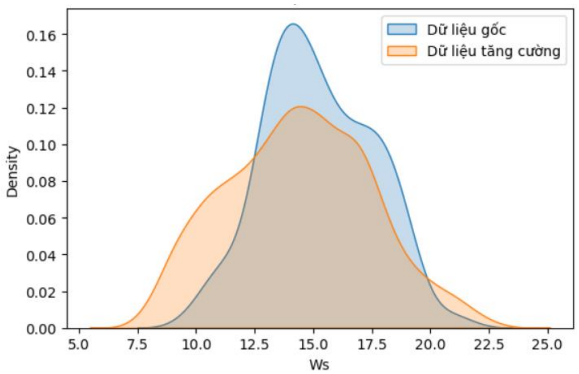
Quá trình cân bằng và tăng cường dữ liệu được thực hiện thông qua các bước hiệu quả nhằm đảm bảo sự đa dạng và cân bằng giữa các lớp trong tập dữ liệu. Đầu tiên, nhiễu **Gaussian** được thêm vào các đặc trưng số, giúp tạo ra các biến thể nhỏ từ dữ liệu gốc để tăng cường sự đa dạng mà vẫn giữ nguyên các đặc điểm quan trọng. Tiếp theo, phương pháp **Scaling** được áp dụng, điều chỉnh giá trị các đặc trưng bằng cách nhân hoặc chia với các hệ số ngẫu nhiên nhỏ, tạo ra các giá trị mới gần với phân phối ban đầu. Đồng thời, mô hình **CTGAN** (Conditional Tabular GAN) được sử dụng để tạo thêm dữ liệu nhân tạo riêng biệt cho từng lớp, đảm bảo dữ liệu nhân tạo vừa cân bằng vừa phản ánh đúng đặc điểm đặc trưng của mỗi lớp. Cuối cùng, các dữ liệu tăng cường này được kết hợp với dữ liệu gốc để hình thành một tập dữ liệu hoàn chỉnh, vừa cân bằng vừa đa dạng, tối ưu hóa cho quá trình huấn luyện mô hình.



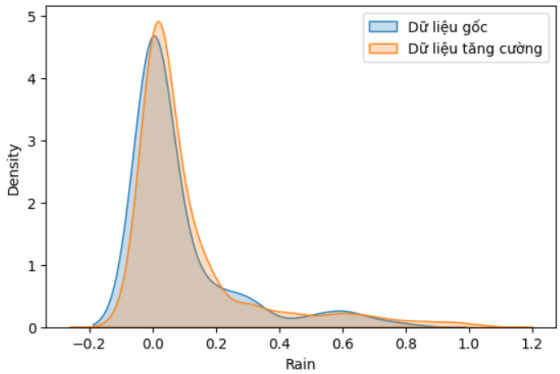
a. Temperature



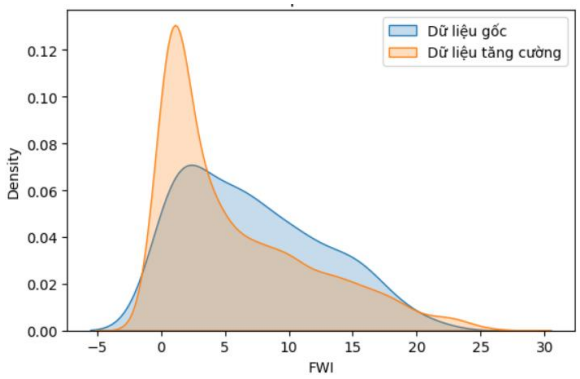
b. RH



c. Ws



d. Rain



e. FWI

Hình 4.1.3 Biểu đồ phân phối dữ liệu trước và sau khi tăng cường

7. Chuẩn hóa dữ liệu

Các giá trị thuộc tính trong tập dữ liệu huấn luyện có sự chênh lệch lớn, có thể ảnh hưởng đến hiệu quả của mô hình. Vì vậy, MinMaxScaler được sử dụng để chuẩn hóa giá trị của các thuộc tính này về khoảng [0,1], đảm bảo tất cả các thuộc tính có mức độ quan trọng là như nhau.

4.2. Kết quả

Kết quả thực nghiệm được tiến hành trên tập dữ liệu đã tăng cường với ba mô hình phân loại Logistic Regression, Random Forest, và Multi-Layer Perceptron (MLP). Những chỉ số hiệu năng bao gồm độ chính xác (Accuracy), độ chính xác của từng lớp (Precision), độ nhạy (Recall), và F1-Score. Kết quả chi tiết của từng mô hình như sau:

Bảng 4.2.1 Bảng so sánh kết quả thực nghiệm giữa các thuận toán.

THUẬT TOÁN	ACCURACY	PRECISION	RECALL	F1-SCORE
LOGISTIC REGRESSION	0.95	0.95	0.96	0.95
RANDOM FOREST	0.99	0.99	0.99	0.99
MLP	0.96	0.97	0.95	0.96

Kết quả thực nghiệm được thực hiện trên tập dữ liệu tăng cường với ba mô hình phân loại phổ biến là Logistic Regression, Random Forest, và Multi-Layer Perceptron (MLP). Những chỉ số đánh giá bao gồm Accuracy, Precision, Recall và F1-Score. Quá trình thử nghiệm đã làm rõ hiệu quả của từng mô hình trong bối cảnh dữ liệu đã được xử lý và tăng cường.

Trong ba mô hình, **Random Forest** đã đạt được kết quả tốt nhất với độ chính xác gần như tuyệt đối, đạt 99%. Điều này cho thấy khả năng kết hợp từ nhiều cây quyết định của Random Forest giúp tăng cường độ mạnh mẽ trong dự đoán và giảm thiểu sai lệch. Các chỉ số Precision, Recall của Random Forest đều duy trì ở mức rất cao (gần 99%), thể hiện sự cân bằng giữa khả năng phát hiện đúng các trường hợp "Fire" và "Not Fire" mà không gây ra quá nhiều sai sót. Ma trận nhầm

lẫn cho thấy chỉ có một số lượng rất nhỏ các mẫu bị dự đoán sai, chứng minh rằng Random Forest là một mô hình đáng tin cậy trong bài toán phân loại nguy cơ cháy rừng.

Mặc dù không đạt hiệu năng cao như Random Forest, **Logistic Regression** và **MLP** cũng thể hiện khả năng phân loại chính xác tốt. Logistic Regression, với độ chính xác 95%, cung cấp một phương pháp dự đoán hiệu quả, giữ được sự đơn giản và tốc độ xử lý nhanh. Precision và Recall của Logistic Regression cũng duy trì ở mức 95%, cho thấy nó có thể là một lựa chọn tối ưu khi tài nguyên tính toán hạn chế.

MLP đạt độ chính xác 96%, nhờ khả năng mô hình hóa các mối quan hệ phi tuyến tính trong dữ liệu. Tuy nhiên, MLP yêu cầu nhiều tài nguyên tính toán hơn và thời gian huấn luyện lâu hơn, điều này có thể là một hạn chế trong các tình huống cần tính toán nhanh chóng. Các chỉ số Precision, Recall của MLP cũng đạt mức khá cao (trên 95%), nhưng sự chênh lệch nhỏ giữa Precision và Recall có thể chỉ ra rằng mô hình này còn gặp khó khăn trong việc cân bằng giữa hai lớp.

Kết quả thực nghiệm cũng cho thấy rằng quá trình tiền xử lý và tăng cường dữ liệu đóng vai trò quan trọng trong việc nâng cao hiệu năng của các mô hình. Dữ liệu ban đầu với chỉ 244 mẫu đã được mở rộng thông qua các kỹ thuật như Gaussian Noise, Scaling, và CTGAN. Điều này không chỉ giúp cân bằng dữ liệu giữa các lớp mà còn tăng cường tính đa dạng, giúp cải thiện khả năng học của các thuật toán. Nhờ vậy, cả ba mô hình đều đạt được hiệu năng cao hơn đáng kể, với độ chính xác vượt mức 95%.

Mặc dù Random Forest vượt trội về mặt hiệu năng, nhưng nó có thể không phải là lựa chọn tối ưu trong các tình huống yêu cầu mô hình hóa trực tuyến hoặc yêu cầu giải thích kết quả chi tiết. Logistic Regression, với cấu trúc tuyến tính đơn giản, có lợi thế trong việc giải thích mối quan hệ giữa các đặc trưng và nguy cơ cháy rừng. Ngược lại, MLP với khả năng học các mối quan hệ phi tuyến có thể phù hợp hơn trong các bài toán phức tạp, đặc biệt khi dữ liệu đầu vào không tuyến tính.

Sinh luật từ các mô hình

Để giải thích và phân tích cách các mô hình dự đoán nguy cơ cháy rừng, các luật được sinh ra từ Logistic Regression và Random Forest cung cấp một cái nhìn sâu sắc về mối quan hệ giữa các đặc trưng đầu vào và nhãn phân loại. Quá trình sinh luật không chỉ minh họa cách thức hoạt động của mô hình mà còn mở ra khả năng ứng dụng trực tiếp vào hệ thống cảnh báo cháy rừng.

- **Logistic Regression**

  Logistic Regression là một mô hình tuyến tính sử dụng các trọng số để biểu diễn mức độ ảnh hưởng của từng đặc trưng đến kết quả phân loại, kết hợp với hệ số tự do (Intercept) để xác định ranh giới phân loại. Trong mô hình này, hệ số tự do **Intercept = -7.1866** đóng vai trò điều chỉnh ngưỡng ban đầu cho các đặc trưng. Giá trị âm của Intercept cho thấy nguy cơ cháy rừng ban đầu được ước tính ở mức thấp, trước khi các đặc trưng tác động đến kết quả dự đoán.

  Các đặc trưng có trọng số dương cao như **FWI** (6.0991), **ISI** (5.9180) và **FFMC** (4.8565) là những yếu tố quan trọng nhất. Điều này cho thấy khi các chỉ số tốc độ lan cháy (ISI), nguy cơ cháy tổng hợp (FWI) và độ ẩm nhiên liệu mịn (FFMC) tăng, khả năng xảy ra cháy rừng cũng

tăng theo. Ngoài ra, BUI (4.5576) và **DC** (3.0045) cũng đóng vai trò quan trọng trong việc xác định nguy cơ cháy, phản ánh tác động của độ cháy tích lũy và hạn hán đến sự bùng phát lửa.

Ngược lại, các đặc trưng có trọng số âm như **Rain** (-3.0159) và **RH** (-0.3924) cho thấy lượng mưa lớn và độ ẩm tương đối cao có tác dụng làm giảm nguy cơ cháy rừng. Tốc độ gió (**Ws** = 1.7261) có trọng số dương nhưng nhỏ hơn, cho thấy nó có thể hỗ trợ lửa lan nhanh nhưng không phải là yếu tố quyết định chính trong bài toán phân loại nguy cơ cháy rừng.

- **Random Forest**

  Random Forest, một tập hợp của nhiều cây quyết định, cung cấp các luật phân nhánh chi tiết dựa trên các đặc trưng và giá trị ngưỡng. Các luật này minh họa mối quan hệ phi tuyến tính phức tạp giữa các yếu tố môi trường và nguy cơ cháy rừng. Ví dụ, nếu **ISI** <= 0.16, **DC** <= 0.28 và **FWI** <= 0.10, nguy cơ cháy rừng là "Không cháy". Ngược lại, nếu **DMC** > 0.25 và **FWI** > 0.26, nguy cơ cháy rừng là "Cháy".

  Các luật phức tạp hơn cũng được ghi nhận, ví dụ trong điều kiện **Rain** > 0.21 và **BUI** <= 0.32, chỉ số ISI trở thành yếu tố quyết định. Điều này chứng tỏ Random Forest có khả năng phát hiện các mối quan hệ đa chiều trong dữ liệu, điều mà Logistic Regression không thể làm được.

Logistic Regression, với trọng số tuyến tính, cung cấp một cách giải thích rõ ràng và dễ hiểu về vai trò của từng đặc trưng trong bài toán phân loại. Đây là lợi thế lớn của Logistic Regression khi áp dụng vào các hệ thống cần tính đơn giản, rõ ràng. Tuy nhiên, Random Forest vượt trội hơn trong việc mô hình hóa các mối quan hệ phi tuyến phức tạp, nhờ vào các luật phân nhánh chi tiết. Điều này giúp Random Forest đạt độ chính xác cao nhất (99%) trong số các mô hình thử nghiệm, mặc dù việc giải thích các luật này có thể khó khăn hơn.

Những luật được sinh ra từ các mô hình có thể được tích hợp vào các hệ thống cảnh báo cháy rừng tự động, giúp cải thiện hiệu quả dự đoán và phản ứng kịp thời. Ví dụ, trong điều kiện thời tiết khô hanh với chỉ số ISI cao và độ ẩm nhiên liệu thấp, hệ thống có thể tự động đưa ra cảnh báo nguy cơ cháy rừng. Ngược lại, khi ghi nhận lượng mưa lớn và độ ẩm tương đối cao, hệ thống có thể giảm mức độ cảnh báo, từ đó tối ưu hóa nguồn lực phòng cháy.

5. Kết Luận

Kết quả nghiên cứu đã chứng minh hiệu quả của các mô hình học máy trong việc phân loại nguy cơ cháy rừng, với Random Forest đạt độ chính xác cao nhất (99%) nhờ khả năng xử lý tốt các đặc trưng phi tuyến và giảm thiểu sai lệch. Logistic Regression và MLP cũng cho thấy hiệu năng tốt, lần lượt đạt độ chính xác 95% và 96%, phù hợp với các ứng dụng yêu cầu tính đơn giản hoặc xử lý phi tuyến. Việc tiền xử lý và tăng cường dữ liệu đóng vai trò quan trọng, giúp cân bằng và mở rộng tập dữ liệu từ 244 lên hơn 2000 mẫu, cải thiện đáng kể hiệu năng của các mô hình. Trong tương lai, nghiên cứu có thể mở rộng với dữ liệu thời gian thực, tích hợp các đặc trưng địa lý, hoặc áp dụng thêm các kỹ thuật học sâu hiện đại để xây dựng hệ thống cảnh báo cháy rừng tự động và hiệu quả hơn.

Tài Liệu Tham Khảo

[1] Abid, F., & Izeboudjen, N. (2020). Predicting forest fire in algeria using data mining techniques: Case study of the decision tree algorithm. Springer International Publishing.

[2] Xue, Q., Lin, H., & Wang, F. (2022). Fcdm: an improved forest fire classification and detection model based on yolov5. Forests.

[3] Khan, S., & Khan, A. (2022). Ffirenet: Deep learning based forest fire classification and detection in smart cities. Symmetry.

[4] Pham, B. T., Jaafari, A., Avand, M., Al-Ansari, N., Dinh Du, T., Yen, H. P. H., ... & Tuyen, T. T. (2020). Performance evaluation of machine learning methods for forest fire modeling and prediction. Symmetry, 12(6), 1022.

[5] MFuchs. (2021). NN - Multi-layer Perceptron Classifier (MLPClassifier).

Link source code: Colab_Project_Mining

PHÂN CÔNG NHÓM

Tên thành viên	Nhiệm vụ
Nguyễn Thúy Thùy Dương	– Tìm kiếm và mô tả dữ liệu. – Cơ sở lý thuyết các thuật toán Logistic Regression, Random Forest và Multi-Layer Perceptron (MLP). – Viết phần giới thiệu, các công trình liên quan, và trình bày thuật toán. – Train mô hình Logistic Regression, Random Forest và Multi-Layer Perceptron (MLP).
Thái Tường An	– Tiền xử lý dữ liệu, bao gồm loại bỏ các thuộc tính không quan trọng, xử lý giá trị bị thiếu và chuyển đổi định dạng dữ liệu. – Tăng cường dữ liệu, bao gồm thêm nhiễu Gaussian, Scaling, và sử dụng mô hình CTGAN. – Sinh luật từ các mô hình Logistic Regression và Random Forest, phân tích mối quan hệ giữa các đặc trưng và nhãn phân loại. – Đánh giá kết quả của các mô hình và viết phần kết quả thực nghiệm. – Tổng hợp và viết phần kết luận.