ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
KHOA KHOA HỌC VÀ KĨ THUẬT MÁY TÍNH

BÁO CÁO KHAI PHÁ DỮ LIỆU

CO3029

GVHD Lê Hồng Trang
Student Phan Khánh Thịnh - 1814182
Võ Công Thành - 1814038
Nguyễn Duy Thìn - 1814149

Hồ Chí Minh, 2021

Contents

1 Giới thiệu

Hiện nay, tính toán dữ liệu theo chuỗi thời gian phổ biến trong các ứng dụng thực tế về việc giám sát các hành vi phức tạp của hệ thống theo chuỗi thời gian như : quản lý các hoạt động công nghệ thông tin, công nghệ sản xuất hay trong lĩnh vực an ninh mạng. Dữ liệu theo chuỗi thời gian có thể đại cho độ trễ, các chỉ số kinh doanh như doanh thu hoặc số lượng user, phản hồi dành cho chiến lược marketing.

1.1 Vấn đề

Trong các ngành công nghiệp phần mềm, việc phát hiện bất thường (phát hiện những hành vi sai lệch bất ngờ so với các hành vi bình thường) cần phải nhanh chóng và thông báo cho người vận hành kịp lúc để giải quyết các vấn đề cơ bản. Việc này cần những kỹ thuật học máy quan trọng để tự động hóa việc xác định các sự cố, bất thường.

Hiện nay đã có nhiều ứng dụng tiềm năng cho phân tích dữ liệu theo chuỗi thời gian. Nhưng nhìn chung cẫn còn nhiều điểm nhức nhối trong quy trình công việc để phân tích dữ liệu theo chuỗi thời gian, gây khó khăn trong việc đánh giá các mô hình đa dạng trên nhiều tập dữ liệu và cài đặt. Chúng bao gồm các giao diện không nhất quán trên mô hình và bộ dữ liệu, các thước đó đánh giá không nhất quán giữa các bài báo và ứng dụng thực tế ngoài công nghiệp và sự thiếu hỗ trợ tương đối cho các tính năng thực tế như xử lý về sau, tự động hóa và kết hợp mô hình.

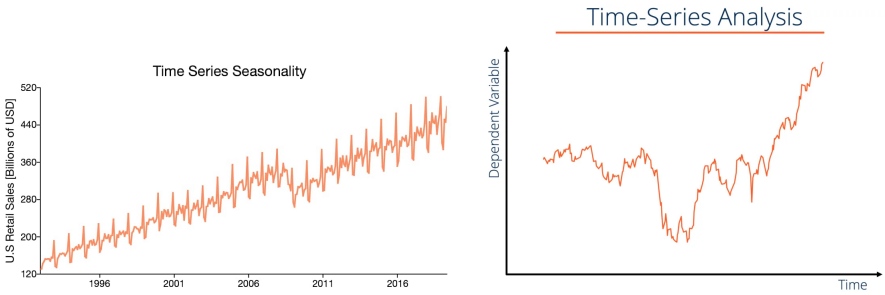

Figure 1: Mô tả dữ liệu theo chuỗi thời gian

Data time series : dữ liệu biến động theo thời gian, dữ liệu lịch sử, dữ liệu thu thập theo thời gian của các đối tượng nghiên cứu, sự vật, sự việc, hiện tượng được quan tâm. Ví dụ dữ liệu bán hàng/ doanh thu theo tháng của một sản phẩm, dữ liệu tiêu thụ năng lượng theo ngày của một nhà máy, dữ liệu thu chi tài chính của một tổ chức công,...

Data time series có các tính chất đặc trưng riêng như sau :

- **Tính xu hướng** : Tính xu hướng là yếu tố thể hiện xu hướng thay đổi của dữ liệu theo thời gian. Đây là đặc trưng thường thấy của rất nhiều dữ liệu chuỗi thời gian. Đặc biệt là các chuỗi trong kinh tế lượng như: giá cả thị trường chị ảnh hưởng của lạm phát, dân số thế giới tăng qua các năm, nhiệt độ trung bình trái đất tăng theo thời gian do hiệu ứng nhà kính,... Tính xu hướng cũng ảnh hưởng không nhỏ tới việc đưa ra nhận định về mối quan hệ tương quan giữa các chuỗi số. Tức là về bản chất các chuỗi không tương quan nhưng do chúng cùng có chung xu hướng theo thời gian nên chúng ta nhận định chúng là tương quan.

- **Tính chu kỳ** : Là qui luật có tính chất lặp lại của dữ liệu theo thời gian. Sự thay đổi thời tiết, sự phát triển của các loài động vật cho tới hành vi mua sắm, tiêu dùng của con người đều bị ảnh hưởng của chu kỳ và lặp lại theo thời gian. Chính vì thế tìm ra được yếu tố chu kỳ sẽ giúp ích cho việc dự báo chính xác hơn.
- **Tính nhiễu** : Là dữ liệu có chứa các thành phần nhiễu, những thành phần còn lại sau khi trích xuất các dữ liệu theo chu kỳ và xu hướng. Nó chỉ ra sự bất thường của các điểm dữ liệu hay gọi là thành phần ngoại biên.

1.2 Merlion

1.2.1 Khái niệm Merlion

Merlion là một thư viện học máy, mã nguồn mở, dùng để xử lý dữ liệu theo thời gian, thường được sử dụng để phát hiện bất thường và dự báo trên chuỗi thời gian đơn biến và đa biến (theo tiêu chuẩn tiền xử lý và hậu xử lý).

Merlion có nhiều modul dễ sử dụng bao gồm trực quan hóa, hiệu chỉnh điểm bất thường để cải thiện khả năng xuyên suốt theo thời gian, autoML tự điểu chỉnh thông số và lựa chọn mô hình và tổ hợp mô hình.

Đồng thời nó có tính năng một giao diện thống nhất cho nhiều mô hình và bộ dữ liệu và cung cấp 1 khung đánh giá độc đáo mô phỏng vệc triển khai trực tiếp và đào tạo lại một mô hình trong sản xuất.

Mục đích : Thư viện Merlion cung cấp cho các kỹ sư và nhà phát triển một giải pháp để phát triển nhanh chóng các mô hình phục vụ nhu cầu theo thời gian và tiêu chuẩn trên nhiều bộ dữ liệu.

1.2.2 Tính năng của Merlion

Merlion là một thư viện Python xử lý thông minh dữ liệu theo chuỗi thời gian. Nó cung cấp một khung học máy end-to-end bao gồm tải và chuyển đổi dữ liệu, xây dựng và đào tạo các mô hình, kết quả đầu ra của mô hình sau xử lý và đánh giá hiệu suất mô hình, bao gồm dự báo và phát hiện bất thường cho cả chuỗi thời gian đơn biến và đa biến với các tính năng chính:

- Framework được tiêu chuẩn hóa và dễ dàng mở rộng để tải dữ liệu, xử lý và đo điểm chuẩn cho một loạt các nhiệm vụ dự báo theo thời gian và phát hiện bất thường.
- Thư viện mô hình đa dạng cho cả việc phát hiện và dự báo bất thường, thống nhất dưới một giao diện. Mô hình bao gồm các phương pháp thống kê cổ điển, cây trừu tượng và học sâu.
- Mô hình trừu tượng DefaultDetector and DefaultForecaster hiệu quả, mạnh mẽ, đạt được hiệu suất tốt và cung cấp một điểm khởi đầu cho người dùng mới.
- Có autoML để điều chỉnh tham số và lựa chọn mô hình thích hợp.
- Có các quy tắc hậu xử lý chung, giảm tỉ lệ dương tính giả, tạo ra điểm dị thường dễ hiểu hơn.
- Output sinh ra dễ sử dụng đồng thời kết hợp nhiều mô hình giúp hiệu suất mạnh mẽ.
- Các pipeline linh hoạt mô phỏng và triển khai trực tiếp và đào tạo lại một mô hình trong sản xuất, đánh giá hiệu suất trên cả dự báo và phát hiện bất thường.

- Hỗ trợ hiện thị mô hình dự đoán.

MERLION: A MACHINE LEARNING LIBRARY FOR TIME SERIES

	Forecast		Anomaly		AutoML	Ensembles	Benchmarks	Visualization
	Uni	Multi	Uni	Multi				
alibi-detect	–	–	✓	✓	–	–	–	–
Kats	✓	✓	✓	✓	✓	–	–	✓
statsmodels	✓	✓	–	–	–	–	–	–
gluon-ts	✓	✓	–	–	–	–	✓	–
RRCF	–	–	✓	✓	–	✓	–	–
STUMPY	–	–	✓	✓	–	–	–	–
Greykite	✓	–	✓	–	✓	–	–	✓
Prophet	✓	–	✓	–	–	–	–	✓
pmdarima	✓	–	–	–	✓	–	–	–
Merlion	✓	✓	✓	✓	✓	✓	✓	✓

Figure 2: Các tính năng của Merlion

1.3 Kiến trúc và nguyên lý

Ở cấp độ cao, kiến trúc của mô-đun Merlion được chia thành năm lớp: lớp Data Layer, chuyển đổi nó thành cấu trúc dữ liệu Merlion's TimeSeries và thực hiện bất kỳ quá trình xử lý trước mong muốn nào; lớp Model hỗ trợ một loạt các mô hình cho cả dự báo và phát hiện bất thường, bao gồm autoML để điều chỉnh siêu thông số tự động; lớp Post Processing cung cấp các giải pháp thiết thực để cải thiện khả năng diễn giải và giảm tỷ lệ dương tính giả của các mô hình phát hiện dị thường; lớp Ensembles and Model Selection, và lớp đánh giá cuối cùng Evaluation Pipeline thực hiện các chỉ số đánh giá có liên quan và các đường ống mô phỏng việc triển khai trực tiếp một mô hình trong sản xuất. Hình dưới cung cấp một cái nhìn tổng quan trực quan về mối quan hệ giữa các mô-đun này.

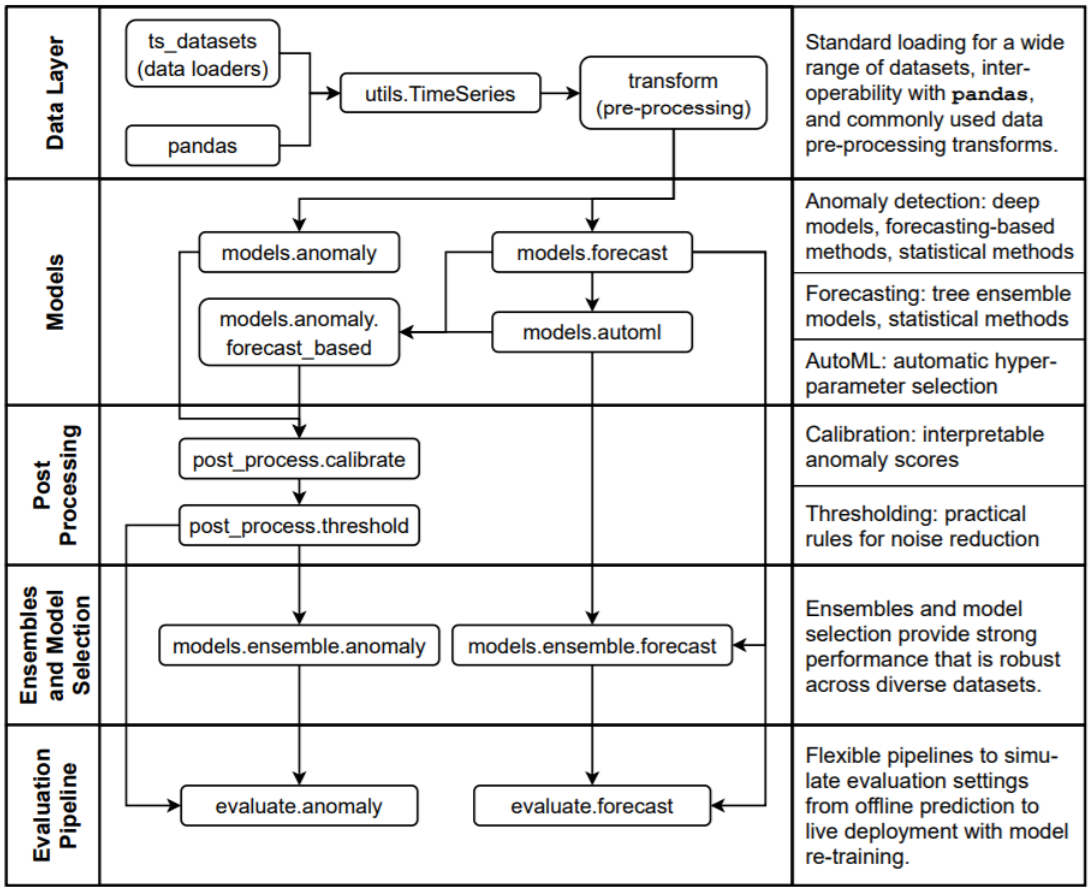

Figure 3: Kiến trúc và nguyên lý của AutoML Merlion

[2]

1. **Data Layer:** Cấu trúc dữ liệu chính của Merlion là TimeSeries. Merlion cho phép người dùng khởi tạo đối tượng TimeSeries trực tiếp từ pandas dataframes, và Merlion đã được hiện thực bộ tải dữ liệu tiêu chuẩn cho một tập dữ liệu rộng trong tập $\mathbf{ts}_{datasets}$.

2. **Models:** Không có một mô hình đơn lẻ nào có thể hoạt động tốt trong mọi tập dữ liệu TimeSeries và tất cả các trường hợp. Do đó, điều quan trọng là phải cung cấp được cho người dùng sự linh hoạt khi chọn nhiều các mô hình không đồng nhất. Merlion heein thực nhiều mô hình khác nhau dành cho dự đoán và phát hiện bất thường. Bao gồm phương pháp thống kê, mô hình dựa vào dạng cây, và phương pháp học sâu và nhiều mô hình khác nữa.
   Lớp này dùng để huấn luyện mô hình, đưa ra các dự đoán và phát hiện các bất thường trong tập dữ liệu.

3. **Post Processing:** Sau khi chạy model xong thì đến bước hậu xử lý, cải thiện hiệu chuẩn về những điểm số bất thường có thể giải thích được. Xử lý để giảm độ nhiễu của kết quả.

4. **Ensembles and Model Selection:** Ensembles được cấu trúc như một mô hình đại diện

cho sự kết hợp của nhiều mô hình cơ bản.Những kết hợp này bao gồm các tổng thể trung bình truyền thống, cũng như lựa chọn mô hình dựa trên các chỉ số đánh giá như sMAPE.

5. **Evaluation Pipeline:** Cung cấp một phạm vi rộng các độ đo ước lượng cho cả dự đoán và phát hiện bất thường, để hỗ trợ việc đánh giá trước khi triển khai mô hình lên môi trường sản phẩm.

2 Mô hình dự báo trong dữ liệu Timeseries

Trong phần này, Merlion giới thiệu các mô hình dự báo đơn biến và đa biến cụ thể của Merlion, cung cấp chi tiết thuật toán trên autoML của Merlion và các mô-đun tập hợp để dự báo.
Merlion chứa một số mô hình để dự báo chuỗi thời gian đơn biến. Chúng bao gồm các phương pháp thống kê cổ điển như ARIMA (Trung bình động tích hợp tự động cải tiến), SARIMA (ARIMA theo mùa) và ETS (Lỗi, Xu hướng, Theo mùa), các thuật toán gần đây hơn như Prophet (Taylor và Letham, 2017), thuật toán sản xuất trước đây như MSES (Cassius et al., 2021)

2.1 Mô hình ARIMA

An autoregressive integrated moving average, or ARIMA là một mô hình phân tích thống kê sử dụng dữ liệu dòng thời gian để hiểu rõ hơn về bộ dữ liệu hoặc để dự đoán các xu hướng trong tương lai.

Một mô hình thống kê là tự động hồi quy nếu nó dự báo các giá trị trong tương lai dựa trên các giá trị trong quá khứ. Ví dụ, một mô hình Arima có thể tìm cách dự đoán giá trong tương lai của cổ phiếu dựa trên hiệu suất trong quá khứ hoặc dự báo thu nhập của công ty dựa trên các giai đoạn quá khứ

Mô hình Arima có thể được hiểu bằng cách phác thảo từng thành phần của nó như sau:

- Autoregression (AR): Đề cập đến một mô hình hiển thị một biến thay đổi chiếm các giá trị chậm, hoặc trước đó.
- Integrated (I): thể hiện sự khác biệt của các quan sát thô để cho phép chuỗi thời gian trở thành stationary (tức là, các giá trị dữ liệu được thay thế bằng sự khác biệt giữa các giá trị dữ liệu và các giá trị trước đó).
- Moving average (MA): Đẳng định sự khác biệt của các quan sát thô để cho phép chuỗi thời gian trở thành văn phòng phẩm (tức là, các giá trị dữ liệu được thay thế bằng sự khác biệt giữa các giá trị dữ liệu và các giá trị trước đó).

Mỗi thành phần trong Arima có chức năng như một tham số có ký hiệu tiêu chuẩn. Đối với các mô hình Arima, một ký hiệu tiêu chuẩn sẽ là Arima với P, D và Q, trong đó các giá trị nguyên thay thế cho các tham số để chỉ ra loại mô hình Arima được sử dụng. Các tham số có thể được định nghĩa là:

- P: Số lượng quan sát độ trễ trong mô hình; Còn được gọi là thứ tự độ trễ.
- D: Số lần quan sát thô là khác nhau; Còn được gọi là mức độ khác nhau.
- Q: Kích thước của cửa sổ trung bình di chuyển; Còn được gọi là thứ tự của trung bình di chuyển.

2.2 Mô hình SARIMA

Seasonal Autoregressive Integrated Moving Average, SARIMA or Seasonal ARIMA, là một phần mở rộng của Arima hỗ trợ rõ ràng dữ liệu chuỗi thời gian đơn với một thành phần theo mùa.

Nó thêm ba hyperparameter mới để chỉ định autoregression (AR), differencing (I) and moving average (MA) cho thành phần theo mùa của chuỗi thời gian, cũng như một tham số bổ sung trong khoảng thời gian của thời vụ.

Cấu hình Sarima yêu cầu chọn HyperParameter cho cả các yếu tố xu hướng (giống như ARIMA) và theo mùa vụ của phần tử chuỗi. Và với SARIMA có thêm phần tử mùa vụ với các thành phần như sau:

- P: Thứ tự autoregressive theo mùa vụ.
- D: Thứ tự khác biệt của mùa vụ.
- Q: Thứ tự trung binh di chuyển của mùa vụ.
- m: Số bước thời gian cho một mùa vụ đơn.

2.3 Mô hình Prophet

Mô hình Prophet được giới thiệu bởi Facebook(Taylor Letham, 2018), Ban đầu dự báo dữ liệu hàng ngày với thời vụ hàng tuần và hàng năm, cộng với hiệu ứng ngày lễ. Sau đó, nó đã được mở rộng để bao gồm nhiều loại dữ liệu theo mùa. Nó hoạt động tốt nhất với loạt thời gian có tính thời vụ mạnh mẽ và một vài mùa của dữ liệu lịch sử.

Mô hình Prophet có thể được coi là một mô hình hồi quy phi tuyến:

$$y_t = g(t) + s(t) + h(t) + \varepsilon_t$$

Trong đó, g(t) miêu tả như là xu hướng thời hạn tăng trưởng, s(t) mẫu mùa vụ khác biệt, h(t) theo dõi những ảnh hưởng khác biệt của kì nghỉ, ε_t là một thông số độ nhiễu.

Những điểm thay đổi cho xu hướng tuyến tính được tự động chọn nếu không được chỉ định rõ ràng. Tùy chọn, một hàm logistic có thể được sử dụng để đặt giới hạn trên về xu hướng.

- Thành phần theo mùa bao gồm các điều khoản Fourier của các giai đoạn có liên quan. Theo mặc định, thứ tự 10 được sử dụng cho tính thời vụ hàng năm và thứ tự 3 được sử dụng cho tính thời vụ hàng tuần.
- Hiệu ứng kỳ nghỉ được thêm vào như các biến giả đơn giản.
- Mô hình được ước tính bằng cách sử dụng cách tiếp cận Bayes để cho phép lựa chọn tự động của Changepoint và các đặc điểm mô hình khác.

2.4 Mô hình MSES

The Multi-Scale Exponential Smoothing model là một gia đình của các mô hình dự báo. Họ sử dụng trung bình có trọng số của các quan sát trong quá khứ để dự báo các giá trị mới. Ở đây, ý tưởng là để mang lại tầm quan trọng hơn cho các giá trị gần đây trong sê-ri. Do đó, vì những

quan sát già đi (theo thời gian), tầm quan trọng của các giá trị này sẽ nhỏ hơn theo cấp số nhân.

Mô hình MSES Kết hợp lỗi, xu hướng và các thành phần theo mùa trong một tính toán làm mịn. Mỗi thuật ngữ có thể được kết hợp một cách bổ sung, nhân lên, hoặc bị bỏ lại khỏi mô hình. Ba thuật ngữ này (lỗi, xu hướng và mùa) được gọi là ets.

3 AutoML

3.1 AutoML thông thường

AutoML có thể được định nghĩa là một tập hợp các công cụ có thể tự động hóa quá trình giải quyết vấn đề với Học máy. Quá trình như vậy bao gồm một số bước yêu cầu chuyên môn cụ thể trong lĩnh vực này, chẳng hạn như xử lý trước dữ liệu, kỹ thuật tính năng, trích xuất và lựa chọn. Không chỉ vậy, các chuyên gia Machine Learning còn phải chọn thuật toán phù hợp và thực hiện các tác vụ tối ưu hóa trong siêu tham số để tối đa hóa độ chính xác của nó.

Khi được kết hợp với các phương pháp và khuôn khổ MLOps để phát triển và triển khai trên quy mô lớn các mô hình Học máy, AutoML có thể trở thành một công cụ thú vị để dân chủ hóa AI cho các tổ chức kinh doanh.

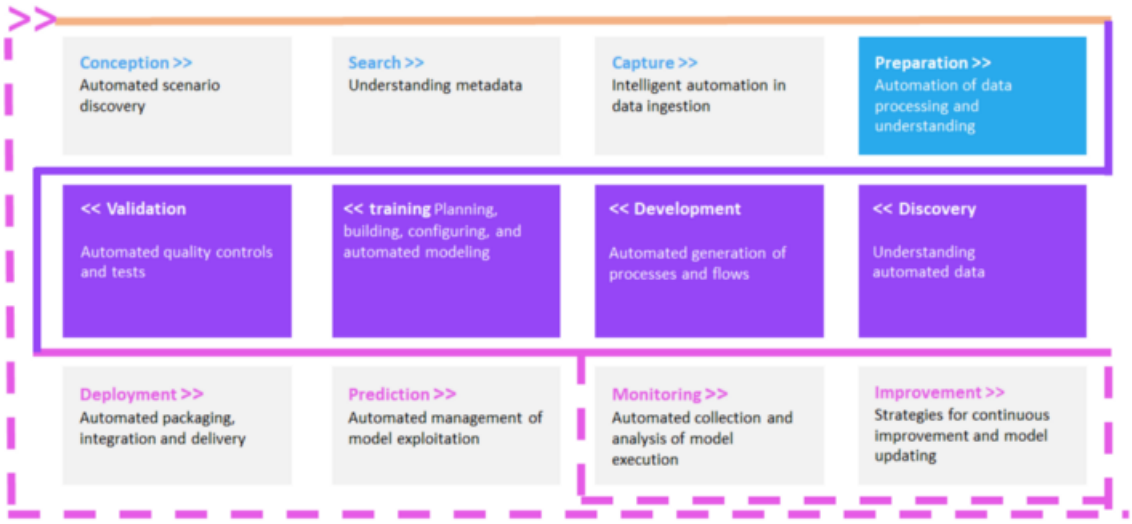

Figure 4: Quy trình Học máy điển hình trong đó AutoML có thể giúp tự động hóa các bước

Cải tiến cốt lõi được sử dụng trong AutoML là tìm kiếm siêu tham số, được sử dụng để xử lý trước các thành phần và lựa chọn kiểu mô hình và để tối ưu hóa siêu tham số của chúng. Có rất nhiều loại thuật toán tối ưu hóa đi từ tìm kiếm ngẫu nhiên đến các thuật toán di truyền học và Bayesian. Các khuôn khổ autoML hiện tại cũng sử dụng trải nghiệm của chúng để cải thiện hiệu suất của chúng. AutoML không thể thay thế chuyên môn của nhà khoa học dữ liệu và định nghĩa đảm nhận, tuy nhiên nó có thể giúp chúng ta tiết kiệm thời gian nếu sử dụng hợp lí.
Tất cả các thuật toán học máy đều có các tham số hoặc trọng số cho từng biến hoặc tính năng

trong mô hình. Một tham số được sinh ra từ quá tình đào tạo. Trong khi siêu tham số là một giá trị có thể điều chỉnh được, dùng để kiểm soát quá trình học máy. Tối ưu hóa siêu tham số để cải thiện hiệu suất mô hình. Các công cụ AutoML có thể tự động đánh giá các siêu tham số khác nhau để xác định tập hợp dẫn đến mô hình hoạt động tốt nhất.

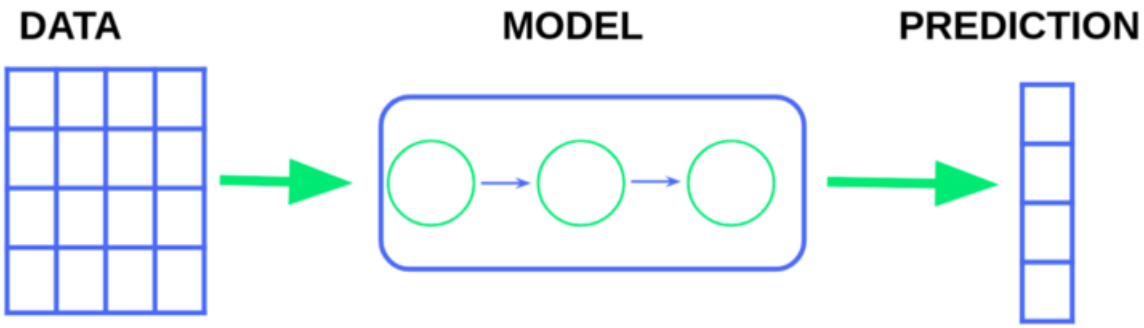

Figure 5: Trường hợp AutoML phổ biến

3.2 Trường hợp không có AutoML

Thông thường, các nhà khoa học dữ liệu phải thực hiện nhiều bước để có được giải pháp cho vấn đề trong thế giới thực bằng cách sử dụng kỹ thuật học máy (ML): làm sạch dữ liệu và chuẩn bị tập dữ liệu, lựa chọn các tính năng nhiều thông tin nhất, chuyển đổi không gian tính năng, lựa chọn Mô hình ML và điều chỉnh các siêu tham số của nó. Trình tự này thường được biểu diễn dưới dạng đường ống ML:

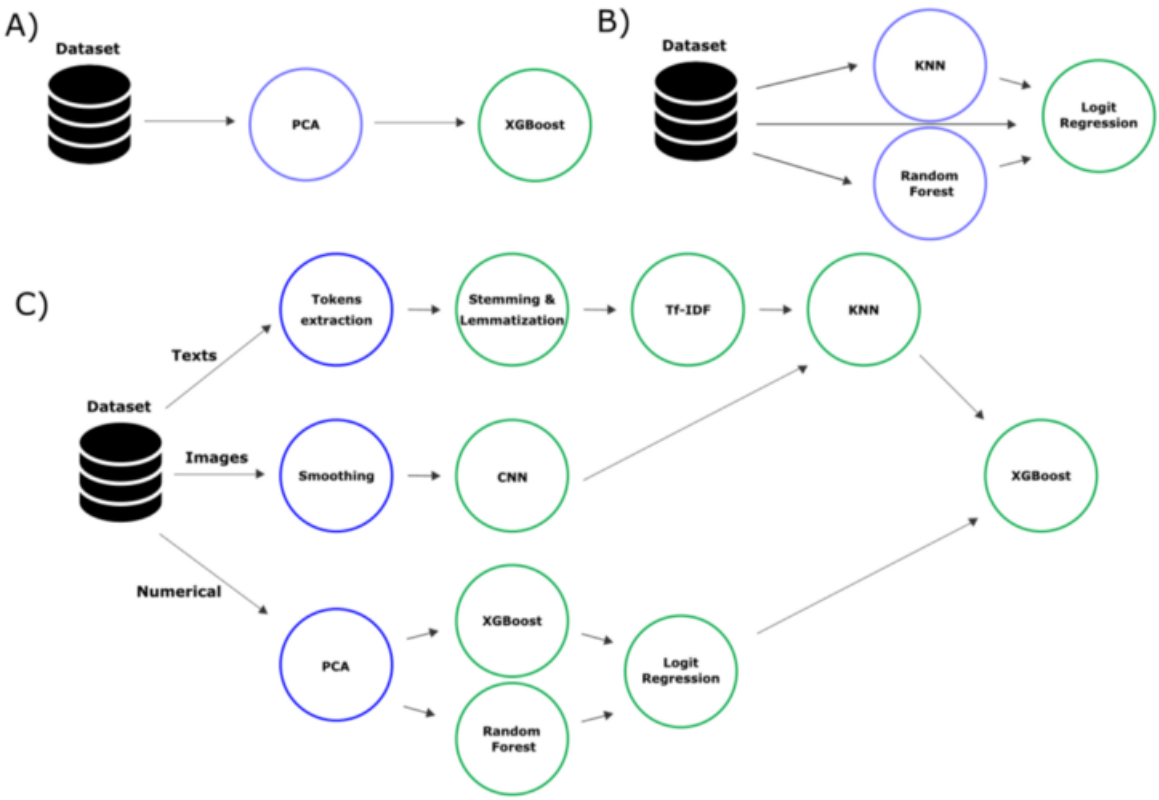

Figure 6: Các đường ống học máy khác nhau

Tuy nhiên, việc xử lý thủ công ngay cả các đường ống tuyến tính đơn giản (A, trong hình trên) và lựa chọn cấu trúc và thông số của chúng có thể mất vài ngày hoặc thậm chí vài tuần làm việc của nhà khoa học dữ liệu. Đối với các nhiệm vụ phức tạp, cấu trúc đường ống có thể trở nên phức hợp hơn - như thể hiện trong trường hợp B và C trong hình trên. Trường hợp B cho thấy đường ống phân nhánh với các phương pháp tổng hợp (xếp chồng) để kết hợp một số mô hình; case C hiển thị đường ống phân nhánh tham gia các phương pháp và mô hình tiền xử lý khác nhau cho các phần khác nhau của tập dữ liệu ban đầu.

Trên thực tế, đường ống với việc sử dụng một số mô hình ML có thể được coi là mô hình tổng hợp toàn bộ vì giữa chúng không có quá nhiều khác biệt theo quan điểm tính toán. Vì vậy, cấu trúc của các đường ống trong (B) và () thực sự trở thành hỗn hợp bởi vì chúng kết hợp các thuật toán ML khác nhau. Ví dụ, một mô hình NLP và một mạng phức hợp có thể được kết hợp để có được dự đoán bằng cách sử dụng dữ liệu đa phương thức. Các mô hình tổng hợp và đường ống ML có thể được xử lý bằng các phương pháp và kỹ thuật AutoML.

Do vậy, nếu không có AutoML thì chúng ta phải tự làm sạch dữ liệu và chuẩn bị tập dữ liệu, lựa chọn các tính năng nhiều thông tin nhất, chuyển đổi không gian tính năng, và đặc biệt là **tự lựa chọn Mô hình ML và điều chỉnh các siêu tham số của nó**.

3.3 AutoML trong Merlion

Ứng dụng trong Merlion không những dùng để tối ưu hóa siêu tham số thông thường mà còn dùng để phát hiện những đặc trưng trong mô hình dự đoán chuỗi thời gian. Và đó là điểm khác biệt giữa AutoML thông thường và AutoML trong Merlion

Để tăng tốc hơn nữa thời gian đào tạo của model autoML, Merlion sử dụng chiến lược xấp xỉ: liệt kê ra một danh sách ban đầu các mô hình ứng viên đạt được hiệu suất tốt sau tương đối ít lần lặp lại tối ưu hóa; sau đó re-train lại từng ứng viên này cho đến khi hội tụ mô hình và cuối cùng là AIC chọn ra mô hình tốt nhất.
Mô hình tốt nhất sẽ được lựa chọn dựa vào điểm số AIC (tiêu chí AIC), mô hình tốt nhất là mô hình có điểm tiêu chí AIC nhỏ nhất.

3.3.1 Tiêu chí thông tin AIC

[1] Tiêu chí thông tin có lẽ là công cụ phổ biến nhất được sử dụng để lựa chọn mô hình, thông qua quá trình so sánh điểm định lượng cho từng mô hình và chọn mô hình có điểm tốt nhất. Điểm tiêu chí thông tin là sự cân bằng giữa tính phù hợp và độ phức tạp của mô hình: cụ thể là

$$-2\log\mathcal{L}(\hat{\theta}) + penalty$$

trong đó L (^) là khả năng tối đa của mô hình ứng viên. Có tính đến độ phức tạp của mô hình con.

Tiêu chí thông tin phổ biến nhất ngày nay là tiêu chí thông tin Akaike (AIC), trong đó penalty AIC chỉ đơn giản là 2k, gấp đôi số tham số trong mô hình. Ý tưởng về tiêu chí thông tin này bắt nguồn từ sự phân kỳ Kullback-Leibler (KL), một phép đo định lượng về khoảng cách có hướng giữa hai mô hình. AIC ước tính lượng thông tin bị mất khi làm gần đúng với mô hình thực sự chưa biết, thực sự tạo ra dữ liệu quan sát, với mô hình ước tính.

$$AIC = -2\log\mathcal{L}(\hat{\theta}) + 2k$$

Trong đó:

- k số tham số trong mô hình
- L(^) log-likelihood: thước đo sự phù hợp của mô hình. Con số càng cao thì càng phù hợp. Điều này thường thu được từ kết quả thống kê.

AIC xác định giá trị thông tin tương đối của mô hình bằng cách sử dụng ước tính khả năng xảy ra tối đa và số lượng các tham số (biến độc lập) trong mô hình.

K mặc định luôn là 2, vì vậy nếu mô hình sử dụng một biến độc lập thì K sẽ là 3, nếu nó sử dụng hai biến độc lập thì K sẽ là 4, v.v.

Để so sánh các mô hình sử dụng AIC, chúng ta cần tính AIC của từng mô hình. Nếu một mô hình thấp hơn 2 đơn vị AIC so với mô hình khác, thì nó được coi là tốt hơn đáng kể so với mô hình đó.

Chỉ số AIC càng nhỏ thì mô hình đó càng tốt, mô hình có chỉ số AIC nhỏ nhất sẽ được lựa chọn.

3.4 Mô hình tự động Sarima (AutoSarima)

Khi xem xét thứ tự mùa vụ phù hợp cho mô hình seasonal ARIMA (Sarima), hạn chế sự chú ý đến độ trễ của mùa vụ. Quy trình mô hình gần giống với dữ liệu phi thời theo mùa vụ, ngoại trừ chúng ta cần chọn các thuật ngữ AR và MA theo mùa cũng như các thành phần phi thời theo mùa vụ của mô hình. Qua trình tự động (Auto) với mô hình Sarima bằng cách lựa chọn những tham số mô hình (p, q, P, Q) được xác định bằng cách tối thiểu chỉ số AIC.

4 Miêu tả dữ liệu

4.1 Giới thiệu dữ liệu

Chúng ta chủ yếu đánh giá những mô hình thực nghiệm trên bộ dữ liệu hàng giờ của M4, nằm trong một cuộc thi dự báo chuỗi thời gian có nhiều ảnh hưởng. Bộ dữ liệu bao gồm khoảng 750 dòng tương ứng với chuỗi thời gian cụ thể từ lĩnh vực dự báo thời tiết như hình 7. Chúng có tầng số lấy mẫu hàng giờ.

	H1
1970-01-01 00:00:00	605.0
1970-01-01 01:00:00	586.0
1970-01-01 02:00:00	586.0
1970-01-01 03:00:00	559.0
1970-01-01 04:00:00	511.0
...	...
1970-01-31 23:00:00	785.0
1970-02-01 00:00:00	756.0
1970-02-01 01:00:00	719.0
1970-02-01 02:00:00	703.0
1970-02-01 03:00:00	659.0

748 rows × 1 columns

Figure 7: Bộ dữ liệu minh họa của M4

4.2 Đồ thị của dữ liệu

Chúng ta bắt đầu bằng cách nhập thư viện chuỗi thời gian của Merlion và sau đó tải dữ liệu lên cho bộ dữ liệu hàng giờ của M4. Sau đó chúng ta có thể chia chuỗi thời gian cụ thể từ bộ dữ liệu vào việc huấn luyện và kiểm thử chúng được thể hiện ở hình số 8. 95% dữ liệu đầu tiên sẽ được dùng cho việc huấn luyện và 5% dữ liệu cuối cùng (mới nhất) sẽ được dùng cho việc kiểm thử cho mục đích thay đổi tham số cho mô hình hiệu quả hơn.

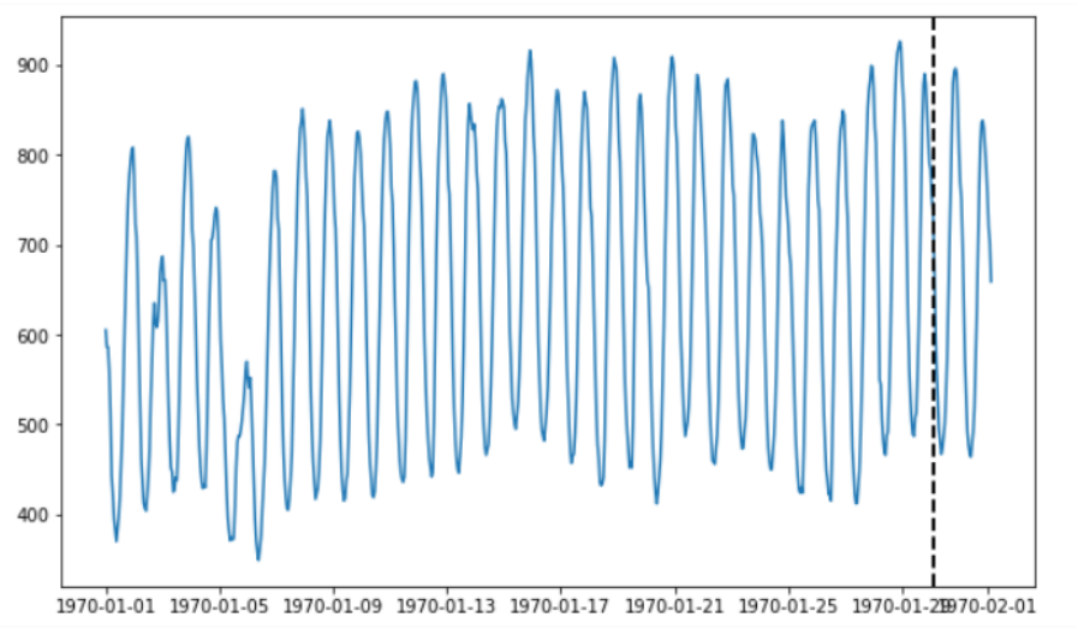

Figure 8: Đồ thị đường thể hiện dòng dữ liệu

5 Trực quan hóa

Liên quan đến việc trực quan hóa dữ liệu, Có hai mô hình được đề xuất ở đây là mô hình Sarima và AutoSarima:

- Trục tung thể hiện giá trị cần được dự báo của dữ liệu và trục hoành thể hiện theo chuỗi thời gian tương ứng với giá trị của dữ liệu.
- Đường màu đen thể hiện giá trị thực (giá trị quan sát) của dữ liệu cần dự báo và đường màu xanh thể hiện giá trị dự báo của dữ liệu dựa trên mô hình đã thực hiện.
- Hai mô hình chỉ ra giá trị dự báo đã so sánh với giá trị thực (giá trị quan sát) tương ứng dựa trên xu hướng, mùa vụ và chu kì của dữ liệu theo chuỗi thời gian như được thấy trong hình 6 và hình 7.

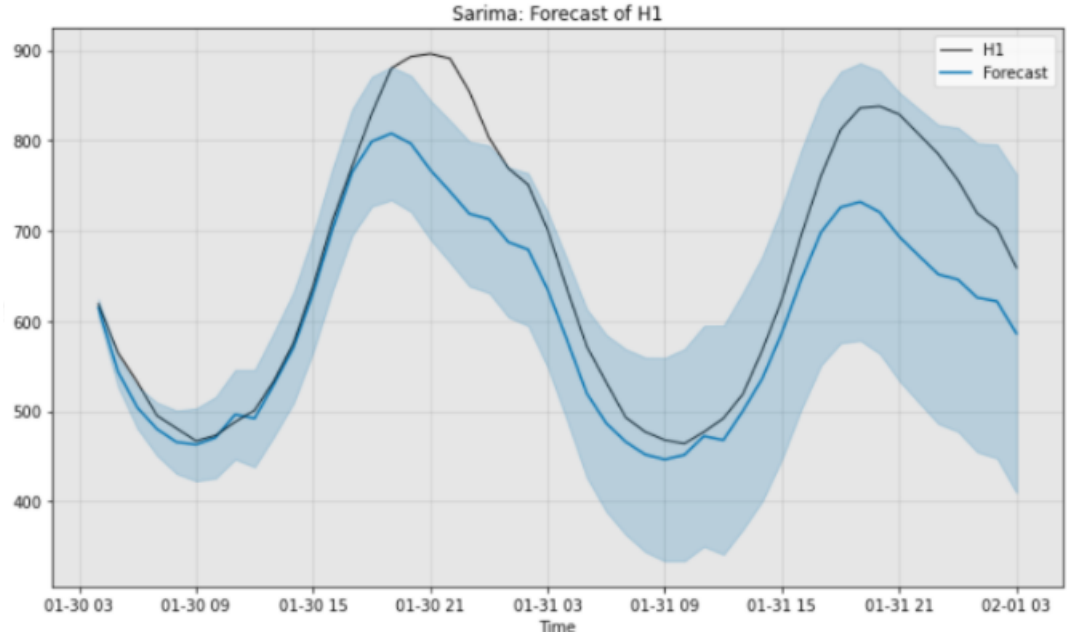

Figure 9: Đồ thị của mô hình Sarima

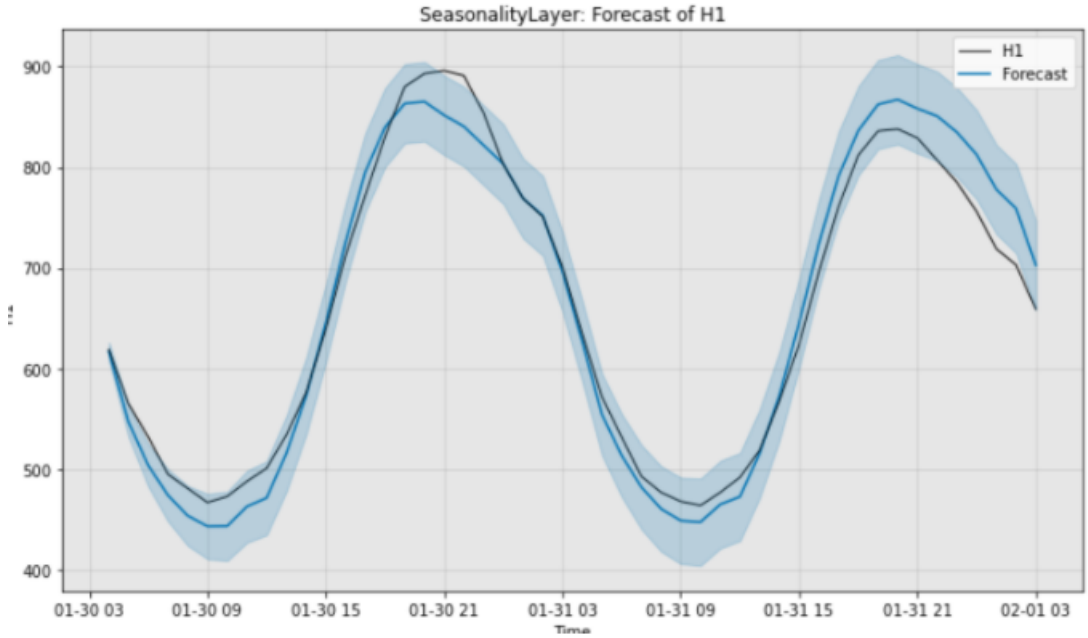

Figure 10: Đồ thị của mô hình AutoSarima

6 Phép đo đánh giá định lượng

6.1 Giới thiệu phép đo

Có nhiều cách để đánh giá độ chính xác của một mô hình dự báo. Một dữ liệu chuỗi thời gian được đưa ra với n quan sát, với y_t được biểu diễn là giá trị thực (giá trị quan sát) ở thời điểm t và $\hat{y}_t$ được biểu diễn là giá trị dự đoán tương ứng. Sau đó lỗi dự báo e_t ở thời điểm t là y_t - $\hat{y}_t$. Những Phép đo lỗi dựa trên lỗi tuyệt đối hoặc bình phương được sử dụng khá phổ biến là:

- Root-mean-square error (RMSE) là một phép đo được sử dụng phổ biến về sự khác biệt giữa giá trị thực với giá trị dự đoán bởi một mô hình hoặc bộ ước tính và giá trị đã quan sát.

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^{n} e_t^2}{n}},$$

- Symmetric mean absolute percentage error (SMAPE or sMAPE) là một độ đo chính xác dựa trên lỗi phần trăm (tương đối). Nó thường xuyên được xác định như sau: Nơi y_t là giá trị thực và $\hat{y}_t$ là giá trị dự đoán.

$$\text{sMAPE} = \frac{1}{n} \sum_{t=1}^{n} \frac{|e_t|}{|y_t| + |\hat{y}_t|} * 200(\%),$$

6.2 Đánh giá định lượng dựa trên các phép đo

Dựa trên hai phép đo đã được giải thích ở trên, Mô hình AutoSarima là một mô hình tốt nhất với các phép đo SMAPE và RMSE bởi vì nó tối ưu được các tham số dựa trên việc lựa chọn tham số tự động.

	sMAPE	**RMSE**
Sarima	7.81	70.29
Arima	5.00	36.67
Prophet	3.72	32.06
MSES	35.03	191.49
ForecasterEnsemble	7.76	56.30
Selector	3.72	32.06
AutoSarima	**3.50**	**27.61**

Bảng 1: Tóm tắt phép đo sMAPE và RMSE của các mô hình. Kết quả tốt nhất được in đậm.

7 Kết luận

Chúng ta đã giới thiệu về thư viện Merlion, một thư viện học máy mã nguồn mở cho chuỗi thời gian nơi được thiết kế để giải quyết nhiều điểm quan trọng trong quy trình công việc trong ngành ngày nay.

Chúng ta đã trình bày và giải thích về AutoML nói chung và AutoML về mô hình dự báo chuỗi thời gian nói riêng giúp cải tiếng được các mô hình dự báo. Từ đó cải thiện được độ chính xác và tốc độ chạy của mô hình qua việc trực quan dữ liệu và đánh giá từ các số đo đã nêu trên.

References

[1] Cohen, N.; Berchenko, Y. Normalized Information Criteria and Model Selection in the Presence of Missing Data. Mathematics 2021, 9, 2474. https://doi.org/10.3390/ math9192474

[2] Merlion: https://arxiv.org/abs/2109.09265

[3] Protphet: https://otexts.com/fpp3/prophet.html

[4] SARIMA: https://machinelearningmastery.com/sarima-for-time-series-forecasting-in-python/

[5] ARIMA: https://www.investopedia.com/terms/a/autoregressive-integrated-moving-average-arima.asp

[6] RMSE: https://en.wikipedia.org/wiki/Root-mean-square_deviation

[7] sMAPE: https://en.wikipedia.org/wiki/Symmetric_mean_absolute_percentage_error

[8] Dữ liệu M4: https://www.kaggle.com/yogesh94/m4-forecasting-competition-dataset

[9] Dữ liệu theo chuỗi thời gian: https://machinelearningcoban.com/tabml_book/ch_data_processing/timeseries_data.html?fbclid=IwAR3_X4TaTGCCV_jjp57wig9aqU19hl0d4LXpFdnScjKNy2iLDwNNYQ2Prks

[10] AIC: https://www.greelane.com/vi/khoa-h%E1%BB%8Dc-c%C3%B4ng-ngh%E1%BB%87-to%C3%A1n/khoa-h%E1%BB%8Dc-x%C3%A3-h%E1%BB%99i/introduction-to-akaikes-information-criterion-1145956/?fbclid=IwAR1WHoZD8Hmz1Hy28_sDTSTTD0l1j6JXr_z1fcO9GfGcMePpJ9EdB2thC7Y