## **Lý do chọn đề tài**

1. Bổ sung thêm các công trình liên quan
2. Thay thế các gạch đầu dòng thành đoạn văn

Trong môi trường sản xuất công nghiệp, các sự cố máy móc không chỉ ảnh hưởng đến hiệu suất mà còn gây tổn thất nghiêm trọng về chi phí và thời gian. Theo báo cáo của McKinsey (2023), trung bình mỗi giờ dừng máy có thể gây thiệt hại lên đến 260.000 USD. Các phương pháp bảo trì truyền thống như bảo trì định kỳ thường không hiệu quả, dẫn đến 30% bảo trì dư thừa và vẫn bỏ sót tới 15% lỗi tiềm ẩn (IEEE, 2021).

Trong bối cảnh **Cách mạng Công nghiệp 4.0**, việc kết hợp giữa trí tuệ nhân tạo (AI), học máy (Machine Learning - ML) và Internet vạn vật (IoT) để phát hiện sớm các bất thường đang trở thành xu thế toàn cầu, đặc biệt quan trọng đối với Việt Nam trong mục tiêu nâng cao năng suất lao động đến năm 2030 (Bộ Công Thương).

Đề tài này nhằm phát triển hệ thống **phát hiện bất thường trong máy móc công nghiệp** thông qua các thuật toán máy học, với mục tiêu hỗ trợ bảo trì dự đoán, giảm thiểu chi phí và tối ưu hóa vận hành nhà máy.

## **1.2. Mục tiêu nghiên cứu**

luận văn đặt ra ba mục tiêu chính, định hướng cho toàn bộ quá trình nghiên cứu:

1. **Phát triển các mô hình nhận diện bất thường**: Mục tiêu trọng tâm là nghiên cứu, thiết kế và phát triển ba loại mô hình học máy cụ thể: **Isolation Forest, Variational Autoencoder (VAE) và Generative Adversarial Networks (GAN)**. Các mô hình này sẽ được tinh chỉnh để có khả năng nhận diện các dấu hiệu bất thường một cách chính xác từ dữ liệu cảm biến đa dạng như áp suất, rung động, nhiệt độ, dòng điện, v.v., thu thập từ các thiết bị công nghiệp.
2. **So sánh và đánh giá hiệu quả**: Sau khi các mô hình được phát triển, một phần quan trọng của luận văn sẽ là tiến hành các thử nghiệm thực nghiệm để **so sánh và đánh giá hiệu quả hoạt động của từng mô hình**. Việc so sánh này không chỉ dừng lại ở các chỉ số kỹ thuật mà còn xem xét khả năng ứng dụng thực tế trong việc cảnh báo sớm. Từ đó, mục tiêu là xác định được giải pháp tối ưu nhất, phù hợp với đặc thù của dữ liệu máy móc công nghiệp.
3. **Tích hợp hệ thống cảnh báo vào môi trường IoT**: Cuối cùng, để biến nghiên cứu lý thuyết thành một giải pháp có giá trị thực tiễn, luận văn sẽ tập trung vào việc **tích hợp hệ thống cảnh báo bất thường vào một môi trường Internet of Things (IoT) hoàn chỉnh**. Điều này bao gồm việc xây dựng một luồng dữ liệu liên tục từ cảm biến đến hệ thống phân tích và sau đó là cơ chế cảnh báo tự động, hỗ trợ đắc lực cho hoạt động bảo trì dự đoán và tự động hóa trong các nhà máy, giúp giảm thiểu thời gian ngừng hoạt động và chi phí bảo trì.

## **1.3. Đối tượng và phạm vi nghiên cứu**

Để đảm bảo tính khả thi và tập trung của luận văn, các đối tượng và phạm vi nghiên cứu được xác định rõ ràng:

* **Đối tượng nghiên cứu**: Luận văn sẽ tập trung phân tích **dữ liệu cảm biến thu thập từ các thiết bị công nghiệp điển hình**, cụ thể là máy bơm và máy nén khí. Các loại máy này thường xuyên hoạt động trong điều kiện khắc nghiệt và việc phát hiện sớm các bất thường có ý nghĩa quan trọng trong việc phòng tránh hỏng hóc lớn.
* **Phạm vi nghiên cứu**: Nghiên cứu sẽ **chỉ tập trung vào bài toán phát hiện bất thường (anomaly detection)**, tức là nhận diện khi có một sự sai lệch so với trạng thái hoạt động bình thường của máy. Luận văn sẽ không mở rộng sang các bài toán phức tạp hơn như dự đoán chính xác thời điểm hỏng hóc hay chẩn đoán nguyên nhân gốc rễ của sự cố. Về mặt đánh giá, trọng tâm chính là khả năng cảnh báo bất thường của hệ thống. Các chỉ số hiệu suất truyền thống như F1-score hay sai số trung bình tuyệt đối (MAE) sẽ không phải là thước đo chính, thay vào đó là tính kịp thời và độ tin cậy của cảnh báo.

## **1.4. Phương pháp tiếp cận**

Luận văn áp dụng một phương pháp tiếp cận kết hợp giữa nghiên cứu lý thuyết chuyên sâu và thực nghiệm ứng dụng:

* **Nghiên cứu lý thuyết**: Giai đoạn này bao gồm việc **tổng quan toàn diện về lý thuyết phát hiện bất thường**, từ các khái niệm cơ bản đến các phương pháp nâng cao. Đồng thời, nghiên cứu sâu về đặc điểm của dữ liệu công nghiệp, vốn thường có tính chu kỳ, nhiễu và đôi khi là mất cân bằng. Quan trọng hơn, luận văn sẽ đi sâu vào cơ chế hoạt động, ưu nhược điểm của từng mô hình học máy được lựa chọn:
  + **Isolation Forest**: Phương pháp dựa trên cây quyết định này được biết đến với khả năng phát hiện các điểm dữ liệu bất thường một cách hiệu quả bằng cách "cô lập" chúng thông qua phân chia ngẫu nhiên.
  + **Variational Autoencoder (VAE)**: Một mô hình học sâu có khả năng học biểu diễn nén (latent representation) của dữ liệu bình thường. Bất thường được phát hiện dựa trên lỗi tái tạo cao khi dữ liệu bất thường được đưa vào mô hình.
  + **Generative Adversarial Networks (GAN)**: Mặc dù chủ yếu được biết đến với khả năng tạo dữ liệu, GAN cũng có thể được biến đổi để phát hiện bất thường bằng cách đánh giá mức độ "giống" của dữ liệu đầu vào với dữ liệu bình thường mà mô hình đã học.
* **Thực nghiệm**: Giai đoạn này là trái tim của luận văn, nơi các lý thuyết được đưa vào thực tiễn:
  + **Thu thập dữ liệu**: Dữ liệu sẽ được thu thập từ các thiết bị công nghiệp thực tế nếu có điều kiện, hoặc sử dụng các bộ dữ liệu công khai có sẵn về hoạt động máy móc. Đây là bước then chốt để đảm bảo tính thực tiễn của nghiên cứu.
  + **Huấn luyện mô hình**: Các mô hình Isolation Forest, VAE và GAN sẽ được huấn luyện trên tập dữ liệu đã thu thập. Quá trình này bao gồm việc tiền xử lý dữ liệu, lựa chọn siêu tham số (hyperparameter tuning) và tối ưu hóa mô hình để đạt được hiệu suất tốt nhất.
  + **Đánh giá kết quả**: Kết quả của từng mô hình sẽ được phân tích kỹ lưỡng. Sự đánh giá sẽ tập trung vào khả năng nhận diện các sự kiện bất thường một cách chính xác và kịp thời, phù hợp với mục tiêu cảnh báo.
* **Triển khai hệ thống**: Để minh họa cho khả năng ứng dụng, luận văn sẽ bao gồm việc xây dựng một hệ thống phát hiện bất thường mẫu:
  + **Giao thức MQTT**: Dữ liệu cảm biến sẽ được truyền tải theo thời gian thực sử dụng giao thức MQTT, một chuẩn nhẹ và hiệu quả cho IoT.
  + **Cơ sở dữ liệu time-series (InfluxDB)**: Dữ liệu sẽ được lưu trữ trong InfluxDB, một cơ sở dữ liệu chuyên biệt cho chuỗi thời gian, giúp việc truy vấn và phân tích dữ liệu hiệu quả.
  + **Hiển thị trực quan qua giao diện web**: Một giao diện web sẽ được phát triển để trực quan hóa dữ liệu cảm biến, hiển thị các cảnh báo bất thường một cách rõ ràng, giúp người vận hành dễ dàng theo dõi tình trạng máy móc.

Luận văn này kỳ vọng sẽ đóng góp vào việc nâng cao khả năng giám sát và bảo trì trong ngành công nghiệp, đặc biệt là thông qua việc ứng dụng hiệu quả các mô hình học máy tiên tiến trong bài toán phát hiện bất thường.

Các Công Trình và Sản Phẩm Tương Tự Hiện Có

#### 1. Các Nghiên cứu Khoa học và Luận văn trước đây

Nhiều nghiên cứu đã ứng dụng các thuật toán học máy, đặc biệt là các phương pháp phát hiện bất thường, vào dữ liệu cảm biến công nghiệp.

* **Nghiên cứu sử dụng Isolation Forest**: Nhiều bài báo đã chứng minh hiệu quả của Isolation Forest trong việc phát hiện lỗi máy móc do tính toán nhanh và khả năng xử lý dữ liệu đa chiều. Ví dụ, các nghiên cứu về phát hiện bất thường trong dữ liệu máy nén khí hoặc động cơ điện thường sử dụng Isolation Forest để xác định các điểm dữ liệu bất thường.
* **Ứng dụng VAE trong giám sát tình trạng**: VAE được đánh giá cao trong việc học biểu diễn dữ liệu bình thường, đặc biệt là với dữ liệu phức tạp như tín hiệu rung động. Các công trình liên quan đến **giám sát sức khỏe kết cấu (Structural Health Monitoring - SHM)** hoặc **phát hiện lỗi trong hệ thống thủy lực** thường sử dụng VAE để tái tạo dữ liệu và phát hiện bất thường dựa trên lỗi tái tạo.
* **GAN cho phát hiện bất thường**: Mặc dù phức tạp hơn, GAN đã được ứng dụng trong việc phát hiện bất thường, đặc biệt là khi dữ liệu bất thường rất hiếm. Một số nghiên cứu đã dùng GAN để tạo ra các mẫu dữ liệu bình thường giả, sau đó sử dụng discriminator để đánh giá "độ bất thường" của dữ liệu mới. Điều này hữu ích trong các hệ thống nơi việc thu thập dữ liệu bất thường là rất khó.
* **So sánh đa mô hình**: Một số luận văn và bài báo khoa học đã tiến hành so sánh giữa các thuật toán khác nhau (ví dụ: SVM, One-Class SVM, Autoencoder, Isolation Forest) trong bài toán phát hiện bất thường trên cùng một tập dữ liệu công nghiệp. Điều này giúp bạn có cái nhìn tổng quan về ưu và nhược điểm của từng phương pháp.

#### 2. Nền tảng và Giải pháp Giám sát Tình trạng (Condition Monitoring - CM) thương mại

Hiện có nhiều công ty cung cấp các giải pháp phần mềm và phần cứng tích hợp để giám sát tình trạng máy móc, thường bao gồm khả năng phát hiện bất thường.

* **Siemens Mindsphere**: Một nền tảng IoT công nghiệp mở, cho phép thu thập dữ liệu từ thiết bị, phân tích dữ liệu bằng các ứng dụng và cung cấp các công cụ cho bảo trì dự đoán và phát hiện bất thường. Họ sử dụng nhiều thuật toán AI/ML khác nhau cho các mục đích này.
* **GE Predix**: Tương tự như Mindsphere, Predix là một nền tảng chuyên biệt cho công nghiệp, tích hợp dữ liệu từ các tài sản công nghiệp và sử dụng phân tích nâng cao để dự đoán và phát hiện các vấn đề tiềm ẩn.
* **PTC ThingWorx**: Cung cấp các công cụ để xây dựng ứng dụng IoT, bao gồm các module về phân tích dữ liệu và machine learning, giúp các doanh nghiệp phát triển giải pháp giám sát và cảnh báo.
* **AVEVA (trước đây là OSIsoft PI System)**: Hệ thống này là một nền tảng thu thập và lưu trữ dữ liệu thời gian thực mạnh mẽ từ hàng nghìn cảm biến công nghiệp. Mặc dù ban đầu không tập trung vào ML, nhưng giờ đây nó tích hợp các công cụ phân tích và API để kết nối với các mô hình phát hiện bất thường.
* **Azure IoT Central/AWS IoT Analytics**: Các dịch vụ đám mây từ Microsoft và Amazon cung cấp các công cụ và dịch vụ để thu thập, xử lý và phân tích dữ liệu IoT, bao gồm các module cho machine learning và phát hiện bất thường, giúp các doanh nghiệp dễ dàng triển khai các giải pháp của riêng họ.

#### 3. Các Dự án Mã nguồn Mở và Thư viện

Cũng có nhiều dự án và thư viện mã nguồn mở hỗ trợ phát hiện bất thường, cung cấp các khối xây dựng cho các giải pháp tùy chỉnh.

* **Scikit-learn**: Thư viện này chứa triển khai của Isolation Forest, cùng với nhiều thuật toán khác cho phát hiện bất thường như One-Class SVM, Local Outlier Factor (LOF).
* **TensorFlow/PyTorch**: Các framework học sâu này là nền tảng để xây dựng và triển khai các mô hình VAE và GAN. Nhiều dự án mã nguồn mở đã chia sẻ mã nguồn cho các biến thể của VAE và GAN ứng dụng trong phát hiện bất thường.
* **Thư viện chuyên biệt**: Một số thư viện Python chuyên biệt như pyod (Python Outlier Detection) cung cấp một bộ sưu tập phong phú các thuật toán phát hiện bất thường, bao gồm cả những thuật toán tiên tiến.

## **1.5. Bố cục luận văn**

Luận văn được chia thành 5 chương:

|  |  |
| --- | --- |
| **Chương** | **Nội dung chính** |
| **Chương 1** | Giới thiệu đề tài, mục tiêu và phương pháp tiếp cận |
| **Chương 2** | Trình bày cơ sở lý thuyết và mô hình thuật toán |
| **Chương 3** | Thiết kế và xây dựng hệ thống phát hiện bất thường |
| **Chương 4** | Triển khai thực nghiệm và đánh giá |
| **Chương 5** | Kết luận và đề xuất hướng nghiên cứu tiếp theo |

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## **2.1. Phát hiện bất thường trong công nghiệp**

### **2.1.1. Khái niệm**

**Phát hiện bất thường** (Anomaly Detection) là quá trình nhận diện những điểm dữ liệu không tuân theo mô hình hoặc xu hướng chung trong tập dữ liệu. Trong ngữ cảnh công nghiệp, điều này bao gồm việc phát hiện lỗi, sự cố vận hành, hoặc hành vi không mong đợi của máy móc.

Có ba loại bất thường chính:

* **Bất thường điểm** (*Point anomalies*): Một điểm dữ liệu đơn lẻ nằm ngoài quy luật (ví dụ: cảm biến đo nhiệt độ tăng đột ngột).
* **Bất thường theo chuỗi thời gian** (*Time-series anomalies*): Dữ liệu biến động theo thời gian và lệch khỏi mô hình thông thường.
* **Bất thường ngữ cảnh** (*Contextual anomalies*): Dữ liệu chỉ được xem là bất thường trong một điều kiện cụ thể (ví dụ: áp suất tăng cao trong điều kiện tải nhẹ).

### **2.1.2. Vai trò trong bảo trì công nghiệp**

Phát hiện sớm bất thường cho phép doanh nghiệp thực hiện **bảo trì dự đoán (Predictive Maintenance)** thay vì bảo trì định kỳ, giúp:

* Giảm thiểu thời gian chết (*downtime*).
* Tiết kiệm chi phí bảo trì.
* Nâng cao độ tin cậy và tuổi thọ thiết bị.

## **2.2. Dữ liệu cảm biến công nghiệp**

### **2.2.1. Đặc điểm**

* **Dữ liệu chuỗi thời gian (time-series):** liên tục theo thời gian, cần xử lý trôi, nhiễu, và xu hướng.
* **Không cân bằng (imbalanced):** các sự kiện bất thường hiếm gặp → mô hình khó học.
* **Nhiễu từ môi trường:** dễ bị ảnh hưởng bởi rung, nhiệt độ, nhiễu từ các thiết bị khác.
* **Đa dạng thiết bị:** Mỗi loại máy móc có tín hiệu khác nhau (máy nén khí, bơm dầu, motor...).

### **2.2.2. Ví dụ nguồn dữ liệu**

* Sensor áp suất, rung động, nhiệt độ (IoT devices).
* Dataset dữ liệu máy móc công nghiệp của công ty, nhà máy đối tác

## **2.3. Các thuật toán học máy trong phát hiện bất thường**

### **2.3.1. Isolation Forest (Liu et al., 2008)**



Thuật toán **Isolation Forest** là một phương pháp hiệu quả để phát hiện các điểm bất thường (anomalies) trong dữ liệu. Điểm đặc biệt của thuật toán này là thay vì cố gắng mô hình hóa các điểm dữ liệu "bình thường", nó tập trung vào việc **cô lập** các điểm bất thường.

## Cách hoạt động của Isolation Forest

Isolation Forest hoạt động dựa trên ý tưởng rằng các điểm bất thường dễ bị cô lập hơn nhiều so với các điểm dữ liệu bình thường. Dưới đây là các bước chính của thuật toán:

### 1. Xây dựng cây cô lập (Isolation Trees - iTrees)

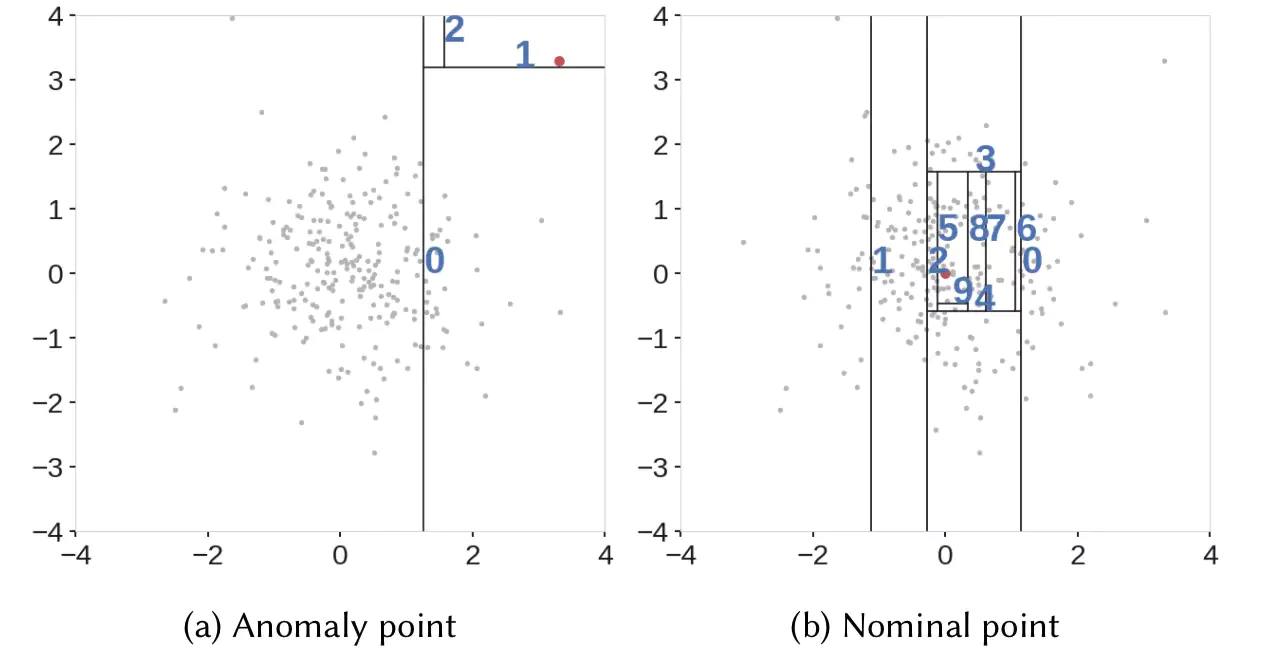
* **Lấy mẫu ngẫu nhiên**: Thuật toán bắt đầu bằng cách tạo ra nhiều cây cô lập (iTrees) từ các tập con (subsets) ngẫu nhiên của dữ liệu. Việc sử dụng các tập con giúp thuật toán hiệu quả về mặt tính toán, đặc biệt với các tập dữ liệu lớn.
* **Phân chia ngẫu nhiên**: Đối với mỗi cây, quá trình xây dựng bắt đầu bằng cách chọn ngẫu nhiên một đặc trưng (feature) từ dữ liệu, sau đó chọn một giá trị phân tách (split value) ngẫu nhiên trong phạm vi của đặc trưng đó. Giá trị này chia dữ liệu thành hai phần.
* **Phân vùng đệ quy**: Quá trình chọn đặc trưng và phân tách ngẫu nhiên này tiếp tục một cách đệ quy. Cây sẽ phát triển cho đến khi mỗi điểm dữ liệu được cô lập trong một nút lá (leaf node) riêng của nó, hoặc đạt đến độ sâu tối đa của cây. Sự ngẫu nhiên trong việc lựa chọn đặc trưng và phân tách đảm bảo rằng các cây không bị thiên vị bởi bất kỳ cấu trúc dữ liệu cụ thể nào.

### 2. Chiều dài đường đi (Path Length)

* **Định nghĩa**: Chiều dài đường đi của một điểm dữ liệu là khái niệm quan trọng trong Isolation Forest. Nó đề cập đến số lượng các cạnh (phân tách) cần thiết để cô lập điểm đó trong một cây cô lập.
* **Chiều dài đường đi ngắn hơn cho điểm bất thường**: Vì các điểm bất thường là độc đáo và nằm rải rác, chúng được cô lập nhanh chóng, dẫn đến chiều dài đường đi ngắn hơn. Những điểm này thường lệch đáng kể so với dữ liệu bình thường, giúp việc phân tách chúng dễ dàng hơn với ít lần phân tách hơn.
* **Chiều dài đường đi dài hơn cho điểm bình thường**: Các điểm dữ liệu bình thường, vốn tập trung và tương tự nhau hơn, yêu cầu nhiều lần phân tách hơn để được cô lập. Do đó, chúng có xu hướng có chiều dài đường đi dài hơn trong các cây cô lập.

### 3. Tính điểm bất thường (Anomaly Score)

Sau khi tạo ra nhiều cây cô lập, thuật toán tính toán điểm bất thường cho mỗi điểm dữ liệu. Điểm này được suy ra từ chiều dài đường đi trung bình của điểm đó trên tất cả các cây. Các điểm có chiều dài đường đi trung bình ngắn hơn sẽ có điểm bất thường cao hơn, cho thấy chúng có nhiều khả năng là các điểm ngoại lai.

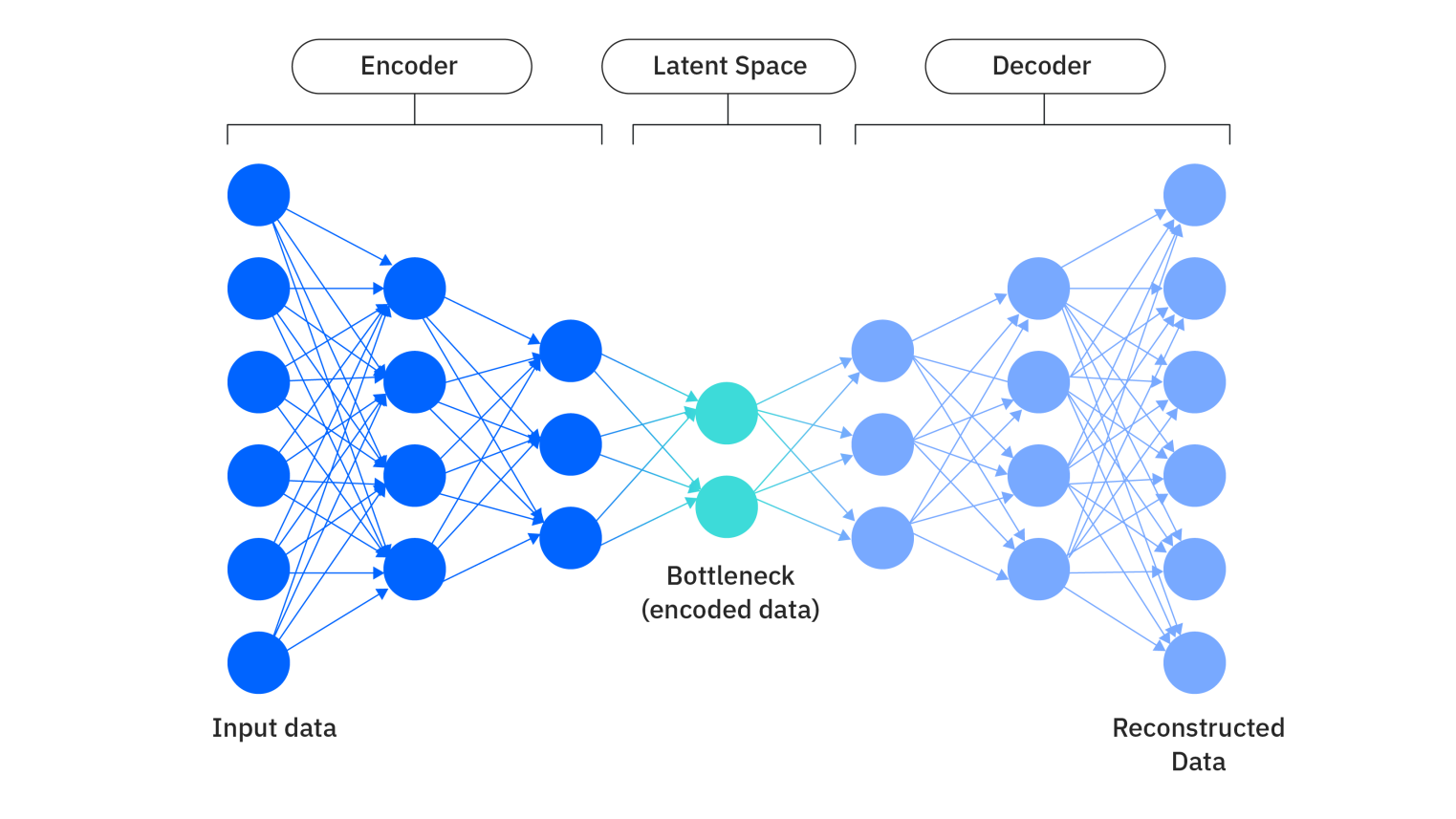


## Ưu điểm của Isolation Forest

* **Hiệu quả**: Hoạt động tốt với các tập dữ liệu lớn và có số chiều cao.
* **Không cần huấn luyện (unsupervised)**: Không yêu cầu dữ liệu được gán nhãn là "bình thường" hay "bất thường" để hoạt động.
* **Hiệu suất tốt**: Thường vượt trội so với các thuật toán phát hiện bất thường truyền thống trong nhiều trường hợp.

Isolation Forest là một công cụ mạnh mẽ và trực quan để phát hiện các điểm bất thường, được ứng dụng rộng rãi trong các lĩnh vực như phát hiện gian lận, phát hiện xâm nhập mạng, và giám sát sức khỏe thiết bị.

### **2.3.2. Variational Autoencoder (VAE) – Kingma & Welling, 2013**



## Variational Autoencoder (VAE) là gì?

**Variational Autoencoder (VAE)** là một mô hình học sâu tiên tiến thuộc họ **mạng nơ-ron sinh tạo (generative neural networks)**. Khác với các mô hình tự mã hóa (autoencoder) truyền thống chỉ học cách nén và giải nén dữ liệu một cách xác định, VAE có khả năng học một **phân phối xác suất** của dữ liệu đầu vào trong một không gian tiềm ẩn (latent space) có cấu trúc. Điều này không chỉ cho phép VAE tái tạo dữ liệu mà còn sinh ra dữ liệu mới có đặc điểm tương tự với dữ liệu gốc, biến nó thành một công cụ mạnh mẽ trong nhiều lĩnh vực, bao gồm phát hiện bất thường.

### Cấu trúc của VAE

Một VAE được cấu tạo từ hai thành phần chính, hoạt động phối hợp với nhau:

1. **Bộ mã hóa (Encoder)**: Bộ mã hóa nhận dữ liệu đầu vào X (ví dụ: một hình ảnh, một chuỗi số liệu) và ánh xạ nó tới các tham số của một phân phối xác suất trong **không gian tiềm ẩn**. Thay vì chỉ tạo ra một vector mã hóa duy nhất, bộ mã hóa của VAE sẽ tạo ra:

* **Vector trung bình (**μ**)**: Đại diện cho vị trí trung tâm của phân phối.
* **Vector log phương sai (**logσ2**)**: Đại diện cho độ phân tán của phân phối. Chúng ta thường giả định phân phối này là **Gaussian (chuẩn)**. Để cho phép quá trình lan truyền ngược gradient (gradient flow) qua việc lấy mẫu từ phân phối này, VAE sử dụng một kỹ thuật thông minh gọi là **"reparameterization trick"**. Thay vì lấy mẫu trực tiếp z∼N(μ,σ2), chúng ta tính toán z=μ+ϵ⋅σ, trong đó ϵ là một biến ngẫu nhiên được lấy mẫu từ phân phối chuẩn tắc N(0,1). Kỹ thuật này giúp VAE có thể được huấn luyện hiệu quả bằng các phương pháp tối ưu hóa dựa trên gradient.

1. **Bộ giải mã (Decoder)**: Bộ giải mã nhận một vector z được lấy mẫu từ không gian tiềm ẩn (từ phân phối mà bộ mã hóa đã học) và cố gắng tái tạo lại dữ liệu đầu vào ban đầu, X^. Mục tiêu của bộ giải mã là đảm bảo X^ phải càng giống X càng tốt, thể hiện khả năng của VAE trong việc học được các đặc trưng quan trọng của dữ liệu.

### Không gian tiềm ẩn (Latent Space) - Trái tim của VAE

**Không gian tiềm ẩn** là một không gian có số chiều thấp hơn so với dữ liệu gốc, nơi mà bộ mã hóa biểu diễn dữ liệu. Tuy nhiên, trong VAE, đây không chỉ là một không gian nén thông thường mà là một không gian **liên tục và có cấu trúc xác suất**. Mỗi điểm trong không gian này đại diện cho một phân phối xác suất của các đặc trưng chính của dữ liệu.

* **Học biểu diễn ngữ nghĩa**: VAE ép buộc bộ mã hóa học một biểu diễn mà ở đó các điểm dữ liệu tương tự nhau sẽ nằm gần nhau trong không gian tiềm ẩn. Điều này tạo ra một "bản đồ" ngữ nghĩa, nơi các biến thể nhỏ trong dữ liệu đầu vào tương ứng với những dịch chuyển nhỏ và mượt mà trong không gian tiềm ẩn.
* **Học phân phối, không phải điểm cố định**: VAE học được sự biến thiên tự nhiên của dữ liệu bằng cách mã hóa mỗi đầu vào thành các tham số của một phân phối xác suất, thay vì một điểm cố định. Điều này giúp mô hình không bị "học thuộc lòng" các điểm dữ liệu cụ thể.
* **Khả năng sinh dữ liệu**: Vì không gian tiềm ẩn của VAE là liên tục và có cấu trúc tốt, chúng ta có thể lấy mẫu ngẫu nhiên từ phân phối chuẩn tắc N(0,1) trong không gian này và truyền chúng qua bộ giải mã để tạo ra các mẫu dữ liệu mới. Các mẫu này sẽ có các đặc điểm tương tự với dữ liệu huấn luyện, minh chứng rằng VAE đã học được "bản chất" của dữ liệu.

### Hàm mất mát (Loss Function) của VAE

Hàm mất mát của VAE là sự kết hợp của hai thành phần, cân bằng giữa khả năng tái tạo dữ liệu và việc học một không gian tiềm ẩn có cấu trúc tốt:

1. **Mất mát tái tạo (Reconstruction Loss)**: Thành phần này đo lường mức độ tương đồng giữa dữ liệu đầu vào và dữ liệu được tái tạo . Mục tiêu là giảm thiểu lỗi tái tạo, đảm bảo bộ giải mã có thể tạo ra dữ liệu chất lượng cao từ không gian tiềm ẩn. Các hàm phổ biến bao gồm **Mean Squared Error (MSE)** cho dữ liệu liên tục hoặc **Binary Cross-Entropy (BCE)**. Lỗi tái tạo cao cho thấy VAE không thể tái tạo lại dữ liệu gốc một cách chính xác.
2. **Mất mát Kullback-Leibler (KL Divergence Loss)**: Thành phần này đo lường sự khác biệt giữa phân phối mà bộ mã hóa học được - thường là Gaussian) và một phân phối tiên nghiệm (prior distribution) đã định trước trong không gian tiềm ẩn (P(z) - thường là Gaussian chuẩn tắc N(0,1)). Mục tiêu là khuyến khích bộ mã hóa tạo ra các phân phối tiềm ẩn gần với phân phối chuẩn tắc, giúp không gian tiềm ẩn trở nên "liền mạch" và dễ dàng lấy mẫu để sinh dữ liệu mới. Điều này ngăn chặn mô hình học các không gian tiềm ẩn rời rạc và không có ý nghĩa.

**Tổng hàm mất mát của VAE** là:

​

Tham số β là một siêu tham số quan trọng, cho phép chúng ta điều chỉnh sự cân bằng giữa hai thành phần của hàm mất mát.

### Ứng dụng VAE trong Phát hiện Bất thường

VAE là một công cụ đặc biệt mạnh mẽ trong việc phát hiện bất thường (anomaly detection), hoạt động dựa trên nguyên lý rằng mô hình được huấn luyện để hiểu và tái tạo dữ liệu "bình thường". Khi một điểm dữ liệu bất thường được đưa vào mô hình, nó sẽ gặp khó khăn trong việc xử lý.

1. **Huấn luyện trên dữ liệu bình thường**: VAE được huấn luyện chủ yếu trên một tập dữ liệu được giả định là **"sạch"** (chỉ chứa dữ liệu bình thường). Điều này rất quan trọng vì nó cho phép VAE học được "mô hình" hoặc "ngôn ngữ" của dữ liệu bình thường trong không gian tiềm ẩn.
2. **Lỗi tái tạo làm chỉ dấu bất thường**: Khi một điểm dữ liệu mới Xtest​ được đưa vào VAE đã huấn luyện, nếu đó là một **điểm bất thường**, bộ mã hóa sẽ gặp khó khăn trong việc ánh xạ nó vào một vùng "hợp lý" trong không gian tiềm ẩn đã được học từ dữ liệu bình thường. Do đó, khi bộ giải mã cố gắng tái tạo lại điểm này từ một biểu diễn tiềm ẩn không "chuẩn", kết quả là **lỗi tái tạo (reconstruction error)** sẽ rất cao. Đây là chỉ số chính để xác định bất thường.
3. **KL Divergence làm chỉ dấu bổ trợ**: Mặc dù ít được sử dụng trực tiếp để xác định bất thường, giá trị KL Divergence cũng có thể cung cấp thông tin bổ sung. Một điểm bất thường có thể tạo ra một phân phối tiềm ẩn có độ lệch lớn hơn so với phân phối tiên nghiệm, dẫn đến giá trị KL Divergence cao hơn. Tuy nhiên, lỗi tái tạo thường là chỉ số mạnh mẽ và trực quan hơn.
4. **Đặt ngưỡng và phát hiện**: Bằng cách đặt một ngưỡng (threshold) cho lỗi tái tạo (hoặc kết hợp lỗi tái tạo và KL Divergence), bất kỳ điểm dữ liệu nào có giá trị vượt quá ngưỡng này đều có thể được gắn cờ là bất thường.

**Ưu điểm chính khi sử dụng VAE để phát hiện bất thường:**

* **Không cần giám sát (Unsupervised)**: VAE có thể được huấn luyện chỉ với dữ liệu "bình thường" hoặc không cần nhãn, điều này rất phù hợp vì dữ liệu bất thường thường hiếm và khó gán nhãn.
* **Học biểu diễn ngữ nghĩa**: VAE học được một biểu diễn nén, liên tục và có ý nghĩa thống kê của dữ liệu bình thường. Điều này giúp nó hiểu được "bản chất" của dữ liệu bình thường.
* **Khả năng sinh dữ liệu**: Mặc dù không trực tiếp liên quan đến phát hiện bất thường, khả năng sinh dữ liệu của VAE có thể hữu ích trong việc tạo ra các mẫu dữ liệu bình thường để mở rộng tập huấn luyện hoặc để kiểm tra mô hình.
* **Phát hiện bất thường "mới lạ" (Novelty Detection)**: VAE hiệu quả trong việc phát hiện các loại bất thường chưa từng thấy trong quá trình huấn luyện, vì chúng không khớp với mô hình dữ liệu bình thường đã học.
* **Biểu diễn dữ liệu học được**: VAE học được một biểu diễn nén, có ý nghĩa của dữ liệu, giúp hiểu sâu hơn về cấu trúc của dữ liệu bình thường.

## Thách thức và cân nhắc

* **Lựa chọn siêu tham số**: Việc chọn kích thước không gian tiềm ẩn (d), tham số β, và ngưỡng cho lỗi tái tạo là rất quan trọng và thường đòi hỏi thử nghiệm.
* **Dữ liệu huấn luyện**: VAE cần một lượng lớn dữ liệu bình thường để học một biểu diễn tốt. Nếu dữ liệu huấn luyện chứa nhiều bất thường, mô hình có thể học cả các đặc điểm của bất thường, làm giảm hiệu quả phát hiện.
* **Tính toán**: Huấn luyện VAE có thể tốn kém về mặt tính toán, đặc biệt với các tập dữ liệu lớn và kiến trúc mạng phức tạp.

**2.3.3. Generative Adversarial Networks (GANs) – [Ian J. Goodfellow](https://arxiv.org/search/stat?searchtype=author&query=Goodfellow,+I+J) , 2014**

## Generative Adversarial Networks (GANs) là gì?

**Generative Adversarial Networks (GANs)**, được giới thiệu bởi Ian Goodfellow và cộng sự vào năm 2014, là một khuôn khổ mạng nơ-ron sinh tạo bao gồm hai mạng đối thủ cạnh tranh với nhau trong một "trò chơi" tổng bằng không (zero-sum game). Hai mạng này được huấn luyện đồng thời:

1. **Mạng sinh (Generator - G)**: Mạng Generator có nhiệm vụ tạo ra dữ liệu tổng hợp (synthetic data) từ một vector nhiễu ngẫu nhiên (random noise vector, thường được lấy mẫu từ phân phối Gaussian). Mục tiêu của Generator là tạo ra dữ liệu tổng hợp chân thực đến mức có thể lừa được mạng Discriminator rằng đó là dữ liệu thật.
2. **Mạng phân biệt (Discriminator - D)**: Mạng Discriminator là một mạng phân loại (classifier) có nhiệm vụ phân biệt giữa dữ liệu thật (từ tập huấn luyện) và dữ liệu giả (được tạo ra bởi Generator). Mục tiêu của Discriminator là phân loại chính xác nguồn gốc của dữ liệu.

**Quá trình huấn luyện đối nghịch (Adversarial Training)**: Trong quá trình huấn luyện, Generator và Discriminator "cạnh tranh" với nhau. Generator cố gắng tạo ra dữ liệu ngày càng chân thực để đánh lừa Discriminator. Ngược lại, Discriminator cố gắng trở nên ngày càng giỏi hơn trong việc phát hiện dữ liệu giả. Quá trình này tiếp diễn cho đến khi Generator tạo ra dữ liệu đủ thuyết phục để Discriminator không thể phân biệt được nó với dữ liệu thật một cách rõ ràng (ví dụ: Discriminator trả về xác suất 0.5 cho cả dữ liệu thật và giả).

## Ứng dụng GANs trong Phát hiện Bất thường

Ý tưởng cốt lõi khi sử dụng GANs để phát hiện bất thường là **huấn luyện GANs chỉ trên dữ liệu "bình thường"**. Sau khi huấn luyện thành công, Generator sẽ học được cách tạo ra dữ liệu giống hệt dữ liệu bình thường. Khi một điểm dữ liệu mới (có thể là bất thường) được đưa vào hệ thống, khả năng "phù hợp" của nó với mô hình dữ liệu bình thường đã học sẽ được đánh giá.

Một số phương pháp tiếp cận phổ biến để sử dụng GANs trong phát hiện bất thường:

### 1. Dựa trên lỗi tái tạo hoặc lỗi ánh xạ ngược (AnoGAN)

Đây là một trong những phương pháp tiên phong và phổ biến nhất (AnoGAN - Anomaly Detection with GANs).

* **Giai đoạn huấn luyện**:
  1. Huấn luyện một GAN tiêu chuẩn chỉ sử dụng **dữ liệu bình thường**. Sau khi huấn luyện, Generator (G) sẽ có khả năng tạo ra các mẫu dữ liệu bình thường rất chân thực từ không gian tiềm ẩn. Discriminator (D) sẽ rất giỏi trong việc nhận biết dữ liệu bình thường.
* **Giai đoạn kiểm tra/phát hiện bất thường**:
  1. Đối với một điểm dữ liệu mới , chúng ta muốn tìm một vector nhiễu z∗ trong không gian tiềm ẩn sao cho gần với nhất có thể. Điều này thường được thực hiện bằng cách tối ưu hóa thông qua lan truyền ngược (backpropagation) để giảm thiểu sự khác biệt giữa và
  2. **Tính điểm bất thường (Anomaly Score)**: Điểm bất thường của được tính dựa trên hai yếu tố chính:
     + **Lỗi tái tạo/Lỗi dư (Residual Error)**: Đo lường sự khác biệt giữa Xtest​ và . Nếu là bình thường, sẽ tái tạo nó tốt, dẫn đến lỗi thấp. Nếu là bất thường, Generator (được huấn luyện chỉ trên dữ liệu bình thường) sẽ không thể tái tạo nó một cách chính xác, dẫn đến lỗi cao.
     + **Điểm Discriminator (Discriminator Score/Feature Matching Error)**: Đo lường mức độ "thật" của trong mắt Discriminator. Một số biến thể còn so sánh các đặc trưng trung gian (intermediate features) của Xtest​ và trong mạng Discriminator. Nếu là bất thường, Discriminator sẽ nhận ra sự "bất thường" trong hoặc trong được tạo ra, dẫn đến một điểm Discriminator cao hoặc một lỗi khớp đặc trưng (feature matching error) lớn.
  3. **Ngưỡng**: Một ngưỡng được đặt ra cho điểm bất thường. Các điểm có điểm số vượt quá ngưỡng này được phân loại là bất thường.

### 2. Sử dụng Discriminator làm bộ phân loại bất thường

Trong một số biến thể, Discriminator tự nó có thể được sử dụng để phát hiện bất thường. Sau khi huấn luyện GAN trên dữ liệu bình thường:

* Discriminator sẽ có khả năng phân biệt dữ liệu thật (bình thường) với dữ liệu giả (được tạo bởi Generator).
* Khi một điểm dữ liệu mới được đưa vào Discriminator, nếu điểm đó là bất thường, Discriminator có xu hướng phân loại nó là "giả" (false) hoặc "không phải thật" với xác suất cao, ngay cả khi nó không được tạo ra bởi Generator. Lý do là điểm bất thường nằm ngoài phân phối dữ liệu bình thường mà Discriminator đã học.
* Điểm xác suất đầu ra của Discriminator có thể được sử dụng trực tiếp làm điểm bất thường.

### 3. GANs dựa trên Encoder (ví dụ: BiGAN, AnoGAN biến thể)

Một số kiến trúc phức tạp hơn như BiGAN (Bidirectional GAN) bao gồm thêm một bộ mã hóa (Encoder) để ánh xạ dữ liệu đầu vào trực tiếp vào không gian tiềm ẩn. Điều này giải quyết vấn đề tìm kiếmcho mỗi mẫu mới, giúp quá trình kiểm tra nhanh hơn nhiều.

* **Giai đoạn huấn luyện**: BiGAN huấn luyện đồng thời một Generator, một Discriminator và một Encoder. Encoder học cách ánh xạ dữ liệu thật vào không gian tiềm ẩn, trong khi Discriminator học cách phân biệt giữa các cặp (x,E(x)) (thật) và (G(z),z) (giả).
* **Giai đoạn kiểm tra**: Đối với một mẫu , chúng ta mã hóa nó thành ​=) và sau đó tính toán điểm bất thường dựa trên lỗi tái tạo và/hoặc điểm của Discriminator khi phân loại cặp (,).

## Ưu điểm của GANs trong Phát hiện Bất thường

* **Học biểu diễn dữ liệu phức tạp**: GANs có khả năng học các phân phối dữ liệu cực kỳ phức tạp và đa dạng, đặc biệt là với dữ liệu có cấu trúc cao như hình ảnh hoặc chuỗi thời gian.
* **Phát hiện bất thường "mới lạ" (Novelty Detection)**: Tương tự VAE, vì GANs được huấn luyện chỉ trên dữ liệu bình thường, chúng sẽ khó tái tạo hoặc nhận diện các mẫu bất thường, cho phép phát hiện các loại bất thường chưa từng thấy.
* **Không cần giám sát (Unsupervised/Semi-supervised)**: Các phương pháp dựa trên GAN thường yêu cầu ít hoặc không cần dữ liệu bất thường được gán nhãn để huấn luyện, điều này cực kỳ hữu ích trong các kịch bản thực tế.
* **Khả năng tạo dữ liệu tổng hợp**: Mặc dù không trực tiếp liên quan đến phát hiện bất thường, khả năng của GANs để tạo ra dữ liệu tổng hợp chất lượng cao có thể hữu ích trong việc tăng cường dữ liệu bình thường cho huấn luyện hoặc mô phỏng.

## Thách thức và cân nhắc

* **Khó khăn trong huấn luyện**: GANs nổi tiếng là khó huấn luyện và không ổn định. Chúng nhạy cảm với siêu tham số (hyperparameters), và có thể gặp phải các vấn đề như "mode collapse" (Generator chỉ tạo ra một số loại mẫu giới hạn) hoặc gradient vanishing/exploding.
* **Điểm bất thường**: Việc định nghĩa và tính toán điểm bất thường có thể phức tạp và cần điều chỉnh tùy thuộc vào kiến trúc GAN và ứng dụng cụ thể.
* **Hiệu suất thời gian thực**: Một số phương pháp (như AnoGAN gốc) yêu cầu tối ưu hóa lặp lại cho mỗi mẫu kiểm tra, điều này có thể làm chậm quá trình phát hiện trong các ứng dụng thời gian thực.
* **Đánh giá định lượng**: Việc đánh giá hiệu suất của GANs trong phát hiện bất thường có thể khó khăn do tính chất không giám sát và sự hiếm có của dữ liệu bất thường.

**2.3.4. Tổng kết so sánh các mô hình**

|  |  |  |  |
| --- | --- | --- | --- |
| Tiêu chí So sánh | Isolation Forest (iForest) | Variational Autoencoder (VAE) | Generative Adversarial Networks (GANs) |
| Cơ chế chính | Dựa trên việc cô lập (isolating) các điểm dữ liệu bằng cách phân vùng ngẫu nhiên. Bất thường dễ bị cô lập hơn. | Học một phân phối xác suất của dữ liệu bình thường trong không gian tiềm ẩn, sau đó tái tạo dữ liệu. | Huấn luyện hai mạng đối nghịch (Generator và Discriminator) để học phân phối dữ liệu bình thường. |
| Cách phát hiện bất thường | Đo chiều dài đường đi (path length) trung bình trong các cây cô lập. Điểm có chiều dài đường đi ngắn hơn là bất thường. | Lỗi tái tạo (reconstruction error) cao: Mô hình khó tái tạo các điểm bất thường vì chúng nằm ngoài phân phối dữ liệu bình thường đã học. | Lỗi tái tạo/ánh xạ ngược (từ Generator) và/hoặc điểm của Discriminator: GANs khó tái tạo hoặc Discriminator dễ dàng nhận diện các điểm bất thường. |
| Yêu cầu dữ liệu huấn luyện | Thường yêu cầu dữ liệu có sự trộn lẫn (bình thường và bất thường), nhưng nó không thực sự học một "mô hình bình thường" cụ thể. | Ưu tiên dữ liệu chỉ chứa điểm bình thường để học chính xác phân phối của dữ liệu bình thường. | Ưu tiên dữ liệu chỉ chứa điểm bình thường để Generator học cách tạo ra dữ liệu bình thường và Discriminator học cách phân biệt chúng. |
| Tính chất học | Không giám sát (Unsupervised): Không yêu cầu nhãn cho dữ liệu. | Không giám sát (Unsupervised): Học tự động các đặc trưng của dữ liệu. | Không giám sát (Unsupervised): Huấn luyện dựa trên sự đối nghịch, không cần nhãn bất thường. |
| Loại dữ liệu phù hợp | Hiệu quả với dữ liệu đa chiều, số liệu liên tục. | Tốt với dữ liệu có cấu trúc cao (hình ảnh, chuỗi thời gian) và dữ liệu có thể học được biểu diễn liên tục. | Rất tốt với dữ liệu có cấu trúc cao và phức tạp (hình ảnh, video, âm thanh) nhờ khả năng học phân phối dữ liệu phức tạp. |
| Độ phức tạp tính toán | Khá nhanh và hiệu quả, đặc biệt với các tập dữ liệu lớn nhờ kỹ thuật subsampling. | Trung bình đến cao: Huấn luyện VAE có thể tốn thời gian và tài nguyên, đặc biệt với kiến trúc mạng sâu. | Rất cao và khó khăn: Huấn luyện GANs nổi tiếng là không ổn định, dễ gặp vấn đề như mode collapse và cần điều chỉnh siêu tham số cẩn thận. |
| Khả năng giải thích (Interpretability) | Tương đối dễ giải thích: Chiều dài đường đi cung cấp một chỉ số trực quan. | Khó giải thích hơn: Biểu diễn không gian tiềm ẩn có thể khó hiểu. | Rất khó giải thích: Cơ chế đối nghịch làm cho quá trình học và các quyết định của mạng khó hiểu. |
| Khả năng phát hiện bất thường "mới lạ" (Novelty Detection) | Tốt: Tập trung vào sự khác biệt, không yêu cầu mô hình hóa phân bố. | Rất tốt: Dữ liệu lạ không phù hợp với phân phối bình thường đã học, gây lỗi tái tạo cao. | Rất tốt: Không thể tái tạo hoặc bị Discriminator nhận diện là "không thật" vì nằm ngoài phân phối đã học. |
| Ưu điểm nổi bật | Nhanh, nhẹ, hiệu quả với dữ liệu lớn, ít tham số cần điều chỉnh. | Tạo ra không gian tiềm ẩn mượt mà, có ý nghĩa, có thể sinh dữ liệu mới. Cung cấp điểm bất thường dựa trên lỗi tái tạo trực quan. | Khả năng học các phân phối dữ liệu cực kỳ phức tạp và sinh ra dữ liệu rất chân thực. |
| Nhược điểm/Thách thức | Có thể không hiệu quả với các điểm bất thường rất gần với phân phối bình thường. Khó với dữ liệu có chiều rất cao. | Huấn luyện có thể phức tạp và nhạy cảm với siêu tham số β. Đòi hỏi lượng lớn dữ liệu bình thường. | Huấn luyện rất khó và không ổn định (mode collapse, gradient vanishing). Thường yêu cầu kiến trúc phức tạp. |

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## **3.1. Yêu cầu hệ thống**

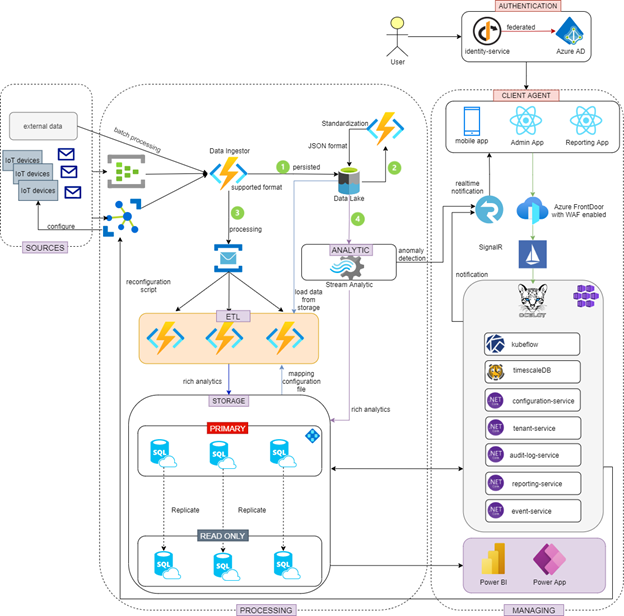
## Hệ thống phát hiện bất thường cần đáp ứng các yêu cầu sau:

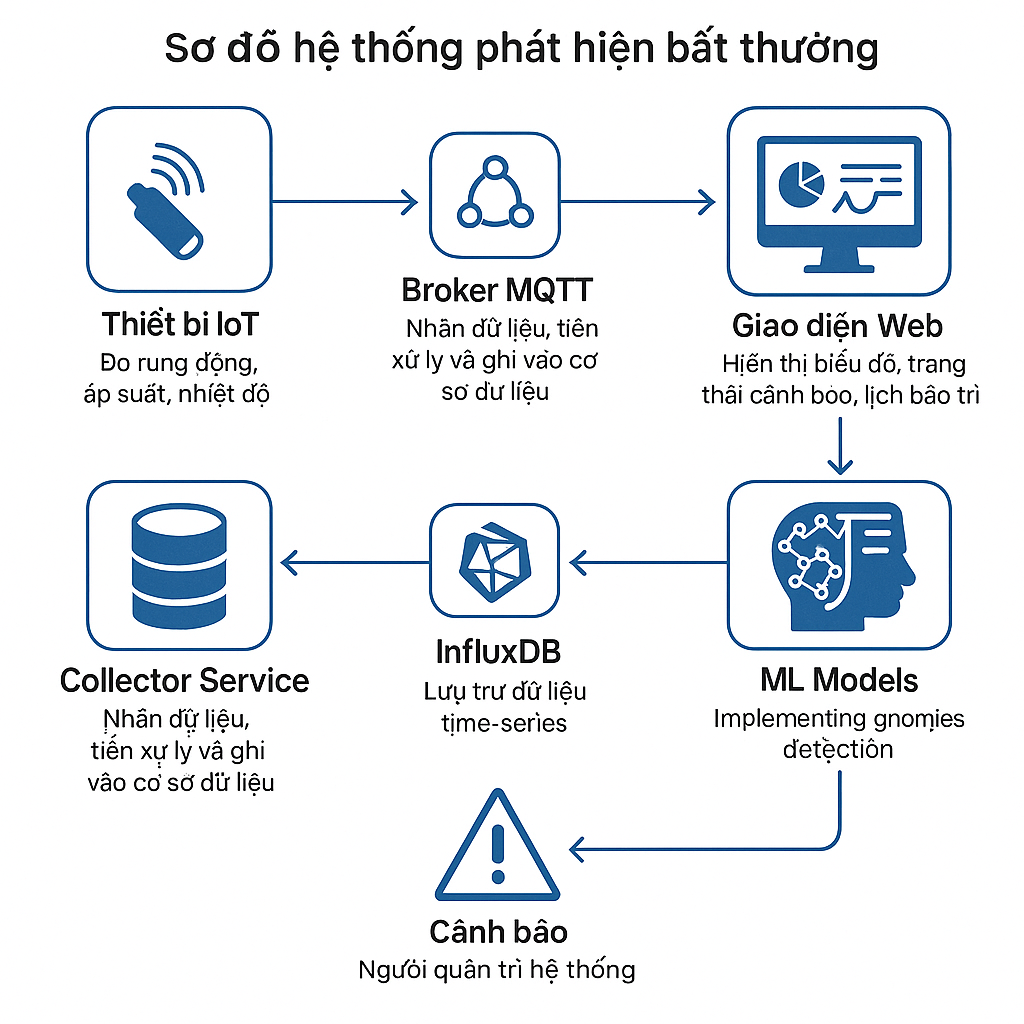
|  |  |
| --- | --- |
| **Nhóm yêu cầu** | **Nội dung chi tiết** |
| **Chức năng chính** | - Thu thập và xử lý dữ liệu từ cảm biến |
| - Phát hiện bất thường dựa trên mô hình học máy |
| - Cảnh báo qua giao diện web hoặc email/API |
| **Hiệu suất** | - Phát hiện bất thường trong thời gian thực |
| - Xử lý >100.000 điểm dữ liệu |
| - Độ trễ dưới 1 giây |
| **Khả năng mở rộng** | - Hỗ trợ nhiều thiết bị cảm biến |
| - Tùy biến mô hình phát hiện tùy loại máy móc |
| **Tính tích hợp** | - Sử dụng giao thức MQTT |
| - Kết nối dễ dàng với hệ thống IoT hiện tại |
| **Giao diện** | - Trực quan, dễ sử dụng |
| - Hiển thị thời gian thực, biểu đồ cảnh báo theo mức độ (xanh - vàng - đỏ) |

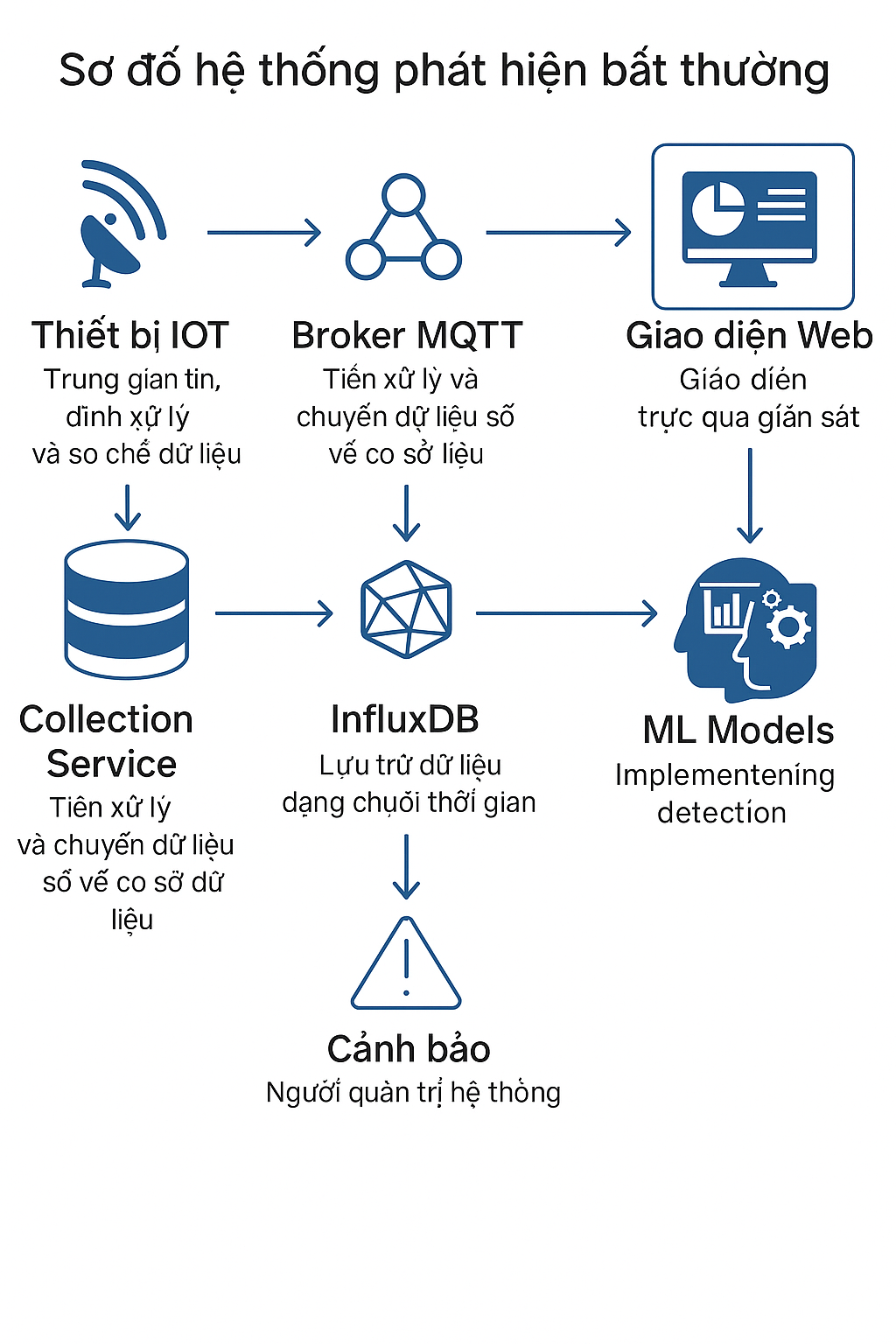
## **3.2. Thiết kế kiến trúc tổng thể**

Hệ thống được chia thành các thành phần chính như sau:

### **🔧 Kiến trúc hệ thống.**







### **✅ Mô tả các thành phần:**

* **Cảm biến:** Đo rung động, áp suất, nhiệt độ.
* **Broker MQTT:** Trung gian truyền dữ liệu cảm biến.
* **Collector Service:** Nhận dữ liệu, tiền xử lý và ghi vào cơ sở dữ liệu.
* **InfluxDB, DataLake:** Lưu trữ dữ liệu time-series.
* **ML Models:** Thực hiện phân tích bất thường với Isolation Forest, VAE, GAN.
* **Giao diện Web:** Hiển thị biểu đồ, trạng thái cảnh báo, lịch bảo trì.

## **3.3. Quy trình phát hiện bất thường**

Quy trình phát hiện bất thường được thiết kế để liên tục giám sát và nhận diện các hành vi không mong muốn hoặc bất thường trong dữ liệu thu thập từ các thiết bị IoT tại nhà máy hoặc của khách hàng. Quy trình này bao gồm năm bước chính:

### 1. Thu thập dữ liệu

Bước đầu tiên và quan trọng nhất là **thu thập dữ liệu** từ các nguồn IoT. Dữ liệu này có thể đến từ nhiều loại cảm biến và thiết bị khác nhau trong môi trường nhà máy, hoặc từ các máy móc, thiết bị của khách hàng. Việc đảm bảo nguồn dữ liệu ổn định và đầy đủ là nền tảng cho toàn bộ quy trình phát hiện bất thường.

### 2. Tiền xử lý dữ liệu

Sau khi thu thập, dữ liệu thô thường chứa nhiễu, giá trị thiếu hoặc định dạng không đồng nhất. Bước tiền xử lý nhằm mục đích làm sạch và chuẩn bị dữ liệu cho quá trình huấn luyện mô hình:

* **Loại bỏ hoặc xử lý dữ liệu thiếu/sai lệch**: Các giá trị trống (NaN), trùng lặp, hoặc sai lệch rõ ràng (ví dụ: giá trị cảm biến vượt quá giới hạn vật lý) cần được xác định và xử lý (ví dụ: loại bỏ, điền giá trị trung bình, nội suy).
* **Chuẩn hóa dữ liệu**: Để đảm bảo các đặc trưng có ảnh hưởng đồng đều đến mô hình, dữ liệu được chuẩn hóa về một phạm vi nhất định. Hai phương pháp phổ biến là:
  + **Min-Max Scaling**: Biến đổi dữ liệu về phạm vi [0,1].
  + **Z-score Standardization**: Biến đổi dữ liệu để có trung bình bằng 0 và độ lệch chuẩn bằng 1.
* **Trích xuất đặc trưng thống kê**: Từ dữ liệu chuỗi thời gian, có thể trích xuất các đặc trưng thống kê hữu ích để cung cấp thêm thông tin cho mô hình. Các đặc trưng này bao gồm:
  + **Mean (Giá trị trung bình)**: Xu hướng trung tâm của dữ liệu.
  + **Standard Deviation (Độ lệch chuẩn - std)**: Mức độ phân tán của dữ liệu.
  + **Skewness (Độ xiên)**: Mức độ bất đối xứng của phân phối dữ liệu.
  + Các đặc trưng khác như median, variance, min, max, v.v.

### 3. Huấn luyện mô hình phát hiện bất thường

Sau khi dữ liệu đã được tiền xử lý, chúng ta tiến hành huấn luyện các mô hình phát hiện bất thường. Việc lựa chọn và cấu hình mô hình phụ thuộc vào đặc điểm của dữ liệu và yêu cầu cụ thể của bài toán:

* **Isolation Forest (iForest)**:
  + Mô hình này có thể được huấn luyện trực tiếp trên **dữ liệu thô** hoặc trên **dữ liệu đã trích xuất đặc trưng thống kê**. iForest hoạt động hiệu quả bằng cách cô lập các điểm dữ liệu, với giả định rằng các điểm bất thường dễ bị cô lập hơn trong một số ít bước phân chia ngẫu nhiên.
* **Variational Autoencoder (VAE) và Generative Adversarial Networks (GANs)**:
  + Các mô hình học sâu này thường được huấn luyện trên **dữ liệu chuỗi thời gian**. Để chuẩn bị dữ liệu cho VAE/GAN, kỹ thuật **Sliding Window** (cửa sổ trượt) được áp dụng. Kỹ thuật này cắt chuỗi dữ liệu dài thành các mẫu chuỗi con có độ dài cố định, tạo ra tập dữ liệu lớn hơn và phù hợp với đầu vào của mạng nơ-ron.
  + Cụ thể, VAE sẽ học một biểu diễn nén của dữ liệu bình thường, trong khi GANs sẽ học cách tạo ra dữ liệu giống hệt dữ liệu bình thường thông qua sự cạnh tranh giữa Generator và Discriminator.

### 4. Phát hiện và đánh dấu bất thường

Sau khi mô hình được huấn luyện, nó sẵn sàng để giám sát dữ liệu mới và phát hiện các điểm bất thường:

* **Tính toán điểm bất thường (Anomaly Score)**: Với mỗi điểm dữ liệu mới (hoặc cửa sổ dữ liệu mới), mô hình sẽ tính toán một "điểm bất thường".
  + **Isolation Forest**: Điểm bất thường tỷ lệ nghịch với chiều dài đường đi trung bình cần thiết để cô lập điểm đó.
  + **VAE**: Điểm bất thường thường dựa trên **lỗi tái tạo (reconstruction error)** của dữ liệu; lỗi càng cao, điểm bất thường càng lớn.
  + **GANs**: Điểm bất thường có thể là sự kết hợp của lỗi tái tạo từ Generator (trong các kiến trúc như AnoGAN) và/hoặc khả năng Discriminator phân loại điểm đó là "không thật".
* **Xác định ngưỡng và đánh dấu**: Một **ngưỡng (threshold)** được thiết lập cho điểm bất thường. Nếu điểm bất thường của một điểm dữ liệu mới vượt quá ngưỡng này, nó sẽ được **đánh dấu là bất thường**.

### 5. Giao diện cảnh báo và đề xuất hành động

Bước cuối cùng là trình bày kết quả phát hiện bất thường cho người dùng và đưa ra các đề xuất hữu ích:

* **Hiển thị cảnh báo trực quan**: Một giao diện người dùng sẽ hiển thị trạng thái của thiết bị hoặc hệ thống bằng màu sắc để biểu thị mức độ bất thường:
  + **Xanh**: Bình thường (Normal)
  + **Vàng**: Nghi ngờ (Suspicious/Warning)
  + **Đỏ**: Bất thường (Anomalous/Critical)
* **Đề xuất lịch bảo trì**: Dựa trên tần suất và mức độ nghiêm trọng của các sự kiện bất thường được phát hiện, hệ thống có thể **đề xuất một lịch trình bảo trì dự đoán** (predictive maintenance). Ví dụ, nếu một thiết bị thường xuyên báo cáo các điểm bất thường nhỏ, điều đó có thể chỉ ra sự hao mòn và cần được kiểm tra hoặc bảo trì sớm để tránh hỏng hóc lớn.

## **3.4. Lựa chọn ngưỡng và xử lý bất thường**

Việc xác định một điểm dữ liệu là bất thường hay không phụ thuộc rất nhiều vào **ngưỡng (threshold)** được thiết lập cho điểm bất thường (anomaly score). Đối với các mô hình phát hiện bất thường không giám sát như Isolation Forest, VAE hay GANs, việc lựa chọn ngưỡng là một bước quan trọng và thường đòi hỏi sự cân nhắc kỹ lưỡng, vì chúng ta không có nhãn "bất thường" rõ ràng trong dữ liệu huấn luyện.

### Phương pháp lựa chọn ngưỡng

Có nhiều cách tiếp cận để lựa chọn ngưỡng, mỗi cách đều có ưu điểm và nhược điểm riêng:

1. **Dựa trên phân phối điểm bất thường**:

* **Z-score**: Phương pháp này tính toán Z-score của từng điểm bất thường dựa trên phân phối của tất cả các điểm bất thường. Những điểm có Z-score vượt quá một giá trị nhất định (ví dụ: 2 hoặc 3 độ lệch chuẩn so với trung bình) sẽ được coi là bất thường. Cách này giả định phân phối điểm bất thường là chuẩn.
* **Percentile (Phân vị)**: Đây là một cách tiếp cận đơn giản và phổ biến. Chúng ta có thể chọn một phân vị nhất định (ví dụ: top 1%, 5%, hoặc 10% các điểm có điểm bất thường cao nhất) làm ngưỡng. Những điểm rơi vào phân vị này sẽ được đánh dấu là bất thường. Phương pháp này đặc biệt hữu ích khi chúng ta biết hoặc ước tính được tỷ lệ phần trăm bất thường trong dữ liệu.
* **Các kỹ thuật thống kê khác**: Có thể sử dụng các phương pháp phân tích phân phối điểm bất thường như Box plot (IQR rule) hoặc thử nghiệm các thuật toán phân cụm (clustering) trên không gian điểm bất thường để tìm ra các nhóm điểm tách biệt.

1. **Kiểm tra thủ công hoặc kinh nghiệm thực tế**:

* Trong nhiều trường hợp ứng dụng thực tế, đặc biệt khi dữ liệu là từ các thiết bị vật lý hoặc quy trình công nghiệp, **kinh nghiệm của các chuyên gia** là vô cùng quý giá. Họ có thể xem xét các trường hợp được mô hình đánh dấu là bất thường và đánh giá xem đó có thực sự là bất thường trong ngữ cảnh thực tế hay không.
* Quá trình này thường là **lặp đi lặp lại**: huấn luyện mô hình, kiểm tra các bất thường được phát hiện, điều chỉnh ngưỡng, và lặp lại cho đến khi đạt được hiệu suất mong muốn (cân bằng giữa việc phát hiện đúng bất thường và giảm thiểu cảnh báo sai).

### Hệ thống hỗ trợ thiết lập ngưỡng động

Để tăng cường tính linh hoạt và độ chính xác, hệ thống nên được trang bị khả năng **thiết lập ngưỡng động (dynamic thresholding)**:

* **Ngưỡng động theo từng loại thiết bị**: Các loại thiết bị khác nhau (ví dụ: máy bơm, cảm biến nhiệt độ, van điều khiển) có thể có hành vi "bình thường" và phạm vi biến động khác nhau. Do đó, một ngưỡng cố định cho tất cả có thể không tối ưu. Hệ thống nên cho phép người dùng hoặc tự động điều chỉnh ngưỡng cho từng loại thiết bị hoặc thậm chí từng thiết bị cụ thể. Điều này giúp giảm thiểu cảnh báo sai (false positives) và tăng cường khả năng phát hiện các bất thường thực sự (true positives) phù hợp với ngữ cảnh riêng của từng thiết bị.
* **Điều chỉnh dựa trên phản hồi (Feedback-driven adjustment)**: Nếu có khả năng thu thập phản hồi từ người dùng về các cảnh báo (ví dụ: "đây là bất thường thực sự", "đây là cảnh báo sai"), hệ thống có thể sử dụng thông tin này để tự động tinh chỉnh ngưỡng theo thời gian, giúp mô hình ngày càng chính xác hơn.
* **Ngưỡng thích nghi theo thời gian**: Hành vi của thiết bị có thể thay đổi theo mùa, tải trọng hoạt động, hoặc quá trình hao mòn. Ngưỡng động có thể thích nghi với những thay đổi này bằng cách liên tục cập nhật dựa trên dữ liệu "bình thường" mới nhất, thay vì dựa vào một ngưỡng tĩnh được thiết lập ban đầu.

Việc lựa chọn và quản lý ngưỡng hiệu quả là chìa khóa để chuyển đổi các điểm bất thường do mô hình tính toán thành các cảnh báo hữu ích, giúp người vận hành đưa ra quyết định kịp thời và chính xác.

# **CHƯƠNG 4: THỰC NGHIỆM**

## **4.1. Tập dữ liệu và tiền xử lý**

### **4.1.1. Nguồn dữ liệu**

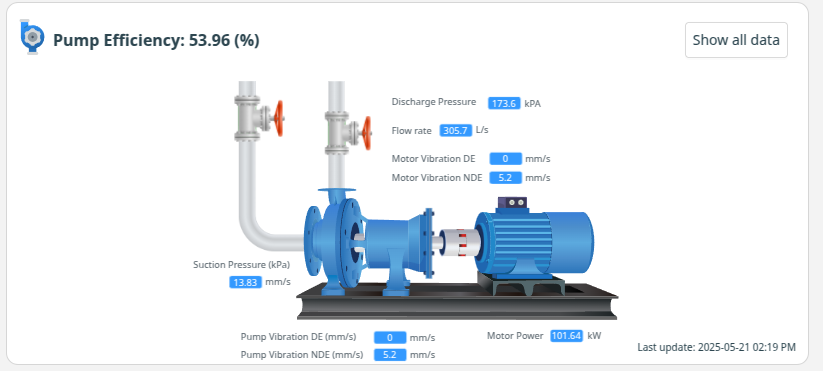
Hai loại dữ liệu được sử dụng trong quá trình thực nghiệm:

* **Dữ liệu thực tế từ nhà máy**: Lấy từ hệ thống cảm biến gắn trên máy bơm, máy nén khí. Gồm các thông số như:  
  + Áp suất (pressure)
  + Rung động (vibration)
  + Nhiệt độ (temperature)

**Mô tả dữ liệu máy bơm**

Thông tin tổng quát

* Số dòng dữ liệu: 6.000 dòng
* Số cột (thuộc tính): 67 cột
* Khoảng thời gian ghi nhận:
* Bắt đầu từ: 01/01/2019 00:00
* Kết thúc vào: 05/05/2019 23:30



Thông số kỹ thuật

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Thông số** | **Đơn vị** | **Giải thích** |
| 1 | Time/Date | - | Mốc thời gian của bản ghi dữ liệu, thường theo giây hoặc mili-giây. |
| 2 | Suction Pressure | kPa | Áp suất hút vào của máy bơm – thể hiện áp lực của chất lỏng trước khi được bơm. |
| 3 | Discharge Pressure | kPa | Áp suất xả ra – áp lực của chất lỏng sau khi được bơm ra. |
| 4 | Discharge FlowRate | L/s | Lưu lượng chất lỏng chảy ra tại đầu xả của máy bơm. |
| 5 | Motor Power | kW | Công suất tiêu thụ của động cơ – phản ánh mức độ tải của máy. |
| 6 | Motor Winding Temp R | °C | Nhiệt độ cuộn dây pha R của động cơ. |
| 7 | Motor Winding Temp S | °C | Nhiệt độ cuộn dây pha S. |
| 8 | Motor Winding Temp T | °C | Nhiệt độ cuộn dây pha T. |
| 9 | Motor Bearing Temp DE | °C | Nhiệt độ ổ trục phía đầu truyền động (Drive End - DE) của động cơ. |
| 10 | Motor Bearing Temp NDE | °C | Nhiệt độ ổ trục phía không truyền động (Non-Drive End - NDE). |
| 11 | Pump Bearing Temp DE | °C | Nhiệt độ ổ trục đầu truyền động của bơm. |
| 12 | Pump Bearing Temp NDE | °C | Nhiệt độ ổ trục không truyền động của bơm. |
| 13 | Motor Vibration DE | mm/s | Rung động tại đầu truyền động của động cơ. |
| 14 | Motor Vibration NDE | mm/s | Rung động tại đầu không truyền động của động cơ. |
| 15 | Pump Vibration DE | mm/s | Rung động đầu truyền động của bơm. |
| 16 | Pump Vibration NDE | mm/s | Rung động đầu không truyền động của bơm. |
| 17 | Fluid Inlet Temp | °C | Nhiệt độ chất lỏng tại đầu vào. |
| **Phân tích rung động theo tần số** | | | |
| 18 | Motor DE FFT for 1X | mm/s | Phổ tần số rung động 1X tại đầu truyền động động cơ – thường liên quan đến tốc độ quay. |
| 19 | Motor DE FFT for 2X | mm/s | Phổ rung động 2X – có thể phản ánh mất cân bằng hoặc lệch tâm. |
| 20 | Motor DE Synchronous | mm/s | Thành phần rung động đồng bộ tại DE. |
| 21 | Motor DE Non-Synchronous | mm/s | Thành phần rung không đồng bộ tại DE. |
| 22 | Motor NDE FFT for 1X | mm/s | Phổ rung 1X tại đầu không truyền động động cơ. |
| 23 | Motor NDE FFT for 2X | mm/s | Phổ rung 2X tại NDE. |
| 24 | Motor NDE Synchronous | mm/s | Thành phần rung đồng bộ tại NDE. |
| 25 | Motor NDE Non-Synchronous | mm/s | Thành phần rung không đồng bộ tại NDE. |
| 26 | Pump DE FFT for 1X | mm/s | Phổ rung 1X tại đầu truyền động bơm. |
| 27 | Pump DE FFT for 2X | mm/s | Phổ rung 2X tại đầu truyền động bơm. |
| 28 | Pump DE Synchronous | mm/s | Rung đồng bộ tại DE của bơm. |
| 29 | Pump DE Non-Synchronous | mm/s | Rung không đồng bộ tại DE của bơm. |
| 30 | Pump NDE FFT for 1X | mm/s | Phổ rung 1X tại đầu không truyền động bơm. |
| 31 | Pump NDE FFT for 2X | mm/s | Phổ rung 2X tại đầu không truyền động bơm. |
| 32 | Pump NDE Synchronous | mm/s | Rung đồng bộ tại NDE bơm. |
| 33 | Pump NDE Non-Synchronous | mm/s | Rung không đồng bộ tại NDE bơm. |
| **Thông số tính toán thủy lực** | | | |
| 34 | Discharge Head | m | Cột áp tại đầu xả – chiều cao chất lỏng được đẩy lên sau khi rời máy bơm. |
| 35 | Suction Head | m | Cột áp hút – chiều cao chất lỏng trước khi vào bơm. |
| 36 | Total Head | m | Tổng chiều cao mà bơm phải thắng (AJ = AH - AI). |
| 37 | Hydraulic Power | kW | Công suất thủy lực thực tế tạo ra bởi bơm. |
| 38 | NPSHa | m | *Net Positive Suction Head available* – mức áp suất ròng còn lại tại đầu hút, cần đủ để tránh cavitation. |
| 39 | Pump Efficiency | % | Hiệu suất thực tế của bơm, = Công suất thủy lực / Công suất đầu vào. |
| 40 | Pressure Vapour | kPa | Áp suất hóa hơi – nếu áp suất hút thấp hơn mức này thì xảy ra hiện tượng cavitation. |

Các chỉ báo lỗi hoặc bất thường (Yes/No)

|  |  |
| --- | --- |
| **Tên cột** | **Giải thích** |
| Flow Performance decrease | Giảm hiệu suất lưu lượng (so với lý tưởng). |
| Pump Efficiency decrease | Hiệu suất bơm giảm. |
| Motor Overheat, Motor Bearing Overheat | Động cơ hoặc ổ trục bị quá nhiệt. |
| Motor/Pump Vibration | Rung động cao vượt ngưỡng. |
| Rotating Loseness | Bộ phận quay bị lỏng. |
| Bearing Defect | Lỗi ổ trục (Motor hoặc Pump). |
| Unbalance | Mất cân bằng cơ học. |
| Cavitation | Hiện tượng xâm thực – nguy hiểm cho bơm. |
| Misalignment | Trục không thẳng hàng – gây rung, hao mòn. |

Phân nhóm tính năng theo chức năng kỹ thuật

|  |  |  |
| --- | --- | --- |
| **Nhóm** | **Tính năng** | **Giải thích ngắn gọn** |
| **Áp suất & Lưu lượng** | Suction Pressure, Discharge Pressure, Discharge FlowRate | Ảnh hưởng trực tiếp đến hiệu suất bơm. |
| **Nhiệt độ động cơ & bơm** | Motor Winding Temp (R, S, T), Motor Bearing Temp (DE, NDE), Pump Bearing Temp (DE, NDE) | Phản ánh tình trạng hoạt động, tải và mài mòn. |
| **Công suất & Nhiệt đầu vào** | Motor Power, Fluid Inlet Temp | Có liên hệ với tải và điều kiện vận hành. |
| **Rung động thô** | Motor/Pump Vibration (DE, NDE) | Chỉ báo sớm của lệch tâm, mất cân bằng hoặc hỏng ổ trục. |
| **Phân tích FFT & Rung động nâng cao** | FFT 1X, 2X, Synchronous, Non-Synchronous cho Motor/Pump | Phân tích sâu về tần số đặc trưng – quan trọng trong phát hiện bất thường cơ khí. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Catetory** | **Failure Mode** | **Data used to detection** | **Failure impact** | **Cause of Failure** | **Recommendations** |
| **Performance** | Flow Performance decrease | Pump Discharge Flow | Mechanical damage | Wear and Tear | Inspect and replace pump components (impellers, wear rings, seals, and bearings) |
|  | Increase fluid temperature that lead to cavitation problem | Blockages or Clogs | Check for and remove any blockages or clogs in the pump or suction line. |
|  |  | Worn Seals or Gaskets |  |
|  |  | System Changes (changes in flow rates, increased resistance due to added piping or fittings, or variations in the characteristics of the fluid being pumped, can affect pump performance if the pump is not adjusted accordingly) |  |
|  |  | Electrical or Motor Problem | Check Electrical Components: Inspect for any electrical issues such as voltage fluctuations or motor inefficiencies. |
| Pump Efficiency Decrease | Motor Power (Current) | Increase power consumption. | Wear and Tear | Inspect and replace pump components (impellers, wear rings, seals, and bearings) |
| Wet well level | Pump could not transfer fluid properly based on performance requirement | Cavitation | Address Cavitation Issues |
| Discharge Pressure |  | Blockages or Clogs | Check for and remove any blockages or clogs in the pump or suction line. |
| Discharge Flow Rate |  | Misalignment or Imbalance | Alignment and Balancing. |
|  |  | Worn Seals or Gaskets | Replace lubricants as per manufacturer recommendations. |
|  |  | System Changes (changes in flow rates, increased resistance due to added piping or fittings, or variations in the characteristics of the fluid being pumped, can affect pump performance if the pump is not adjusted accordingly) | Check Electrical Components: Inspect for any electrical issues such as voltage fluctuations or motor inefficiencies. |
|  |  | Inadequate or degraded lubrication |  |
|  |  | Electrical or Motor Problems |  |
| **Motor** | Motor Overheat | Motor winding temperature sensor phase (R,S & T) | Will lead to motor trip | Excessive Load | Check Load and Conditions: Ensure that the motor isn't being overloaded. |
|  |  | Poor Ventilation | Improve Ventilation |
|  |  | Faulty Cooling Systems | Inspect Cooling Systems |
|  |  | High Ambient Temperature | Check Electrical Issues (overvoltage, phase imbalances, or faulty connections) |
|  |  | Insufficient Lubrication |  |
|  |  | Electrical Issues (overvoltage, phase imbalances, or electrical faults) |  |
|  |  | Mechanical Issues (Misalignment of the motor shaft, worn-out parts) |  |
| Motor Bearing Overheat | Motor Vibration | Will lead to motor trip | Inadequate, degraded or contaminant lubrication. | Check Lubrication |
| Motor Bearing emperature |  | Incorrect installation of bearings | Inspect Mechanical Components (Check for misalignments, worn-out parts, or any mechanical issues causing increased friction |
|  |  | Overloading or Misalignment |  |
|  |  | Insufficient Clearance or Tightness |  |
|  |  | Aging or Wear Bearings |  |
|  |  | Environmental Factors |  |
| Motor Vibration | Motor Vibration | Will lead to motor trip. | Unbalance | Balancing |
|  | Component damage (bearing, seal or coupling) | Misalignment | Alignment coupling |
|  |  | Lubrication problem | Replace lubrication/perform oil purifier. |
|  |  | Bearing defect | Replace bearing. |
|  |  | Foundation | Repair the foundation. |
|  |  | Resonance |  |
|  |  | Looseness |  |
| Motor Rotating Looseness | Motor Bearing Vibration | Premature wear in mechanical drive components that leads to premature failures | Excessive motor bearing clearance | Replace bearing with proper installation Check and fit motor bearing housing |
| Motor Bearing Defect | Motor Bearing Vibration | Abnormal noise, high vibration led to pump cannot run | Bearing Lubrication contamination, | Replace bearing |
| Motor Bearing Temperature |  | misalignment |  |
|  |  |  |  |
| Motor Unbalance | Motor vibration with 1x dominant | Will lead to motor trip | Motor Manufacturing defect | Perform motor solo run to ensure balancing problem on motor |
| Motor current |  | Wear and Tear | Perform balancing process on motor |
|  |  | Accumulation of Deposits |  |
|  |  | Uneven Coating or Surface Damage |  |
|  |  | Incorrect assembly of components or improper fitting of parts during maintenance or repair. |  |
|  |  | Material Changes. |  |
|  |  | Excessive Loading or Stress |  |
|  |  | Foundation or Mounting Issues |  |
| **Pump** | Pump Bearing Overheat | Pum Vibration | Will lead to Pump trip | Inadequate, degraded or contaminant lubrication. | Check Lubrication |
| Pump Bearing emperature |  | Incorrect installation of bearings | Inspect Mechanical Components (Check for misalignments, worn-out parts, or any mechanical issues causing increased friction |
|  |  | Overloading or Misalignment |  |
|  |  | Insufficient Clearance or Tightness |  |
|  |  | Aging or Wear Bearings |  |
|  |  | Environmental Factors |  |
| Pump Bearing Defect | Pump Bearing Vibration | Abnormal noise, high vibration led to pump cannot run | Bearing Lubrication contamination, | Replace bearing |
| Pump Bearing Temperature |  | misalignment |  |
|  |  |  |  |
| Pump Vibration | Pump Vibration | Will lead to pump trip. | Unbalance | Balancing |
|  | Component damage (bearing, seal or coupling) | Misalignment | Alignment coupling |
|  |  | Lubrication problem | Replace lubrication/perform oil purifier. |
|  |  | Bearing defect | Replace bearing. |
|  |  | Foundation | Repair the foundation. |
|  |  | Resonance |  |
|  |  | Looseness |  |
| Pump Rotating Looseness | Pump Bearing Vibration | Premature wear in mechanical drive components that leads to premature failures | Excessive pump bearing clearance | Replace bearing with proper installation Check and fit pump bearing housing |
| Pump Unbalance | Pump vibration with 1x dominant | Will lead to pump trip | Pump Manufacturing defect | Perform balancing process on pump |
| Motor current |  | Wear and Tear |  |
|  |  | Accumulation of Deposits |  |
|  |  | Uneven Coating or Surface Damage |  |
|  |  | Incorrect assembly of components or improper fitting of parts during maintenance or repair. |  |
|  |  | Material Changes. |  |
|  |  | Excessive Loading or Stress |  |
|  |  | Foundation or Mounting Issues |  |
| Cavitation | Pump vibration | Abnormal noise | Insufficient Net Positive Suction Head (NPSH) | Increase inlet/suction pressure. |
| Inlet pressure | High vibration led to pump cannot run. | High Fluid Velocity | Inspect Impeller and Wear Rings |
| Vibration | Damage pump component | Blockages or Restrictions in the Suction Line |  |
|  |  | Worn-out pump components (impellers, wear rings, or volute casings, can create turbulence or irregular flow patterns, leading to cavitation |  |
|  |  | Operating a pump too far to the right of its performance curve (BEP) |  |
|  |  | Increase Temperatures |  |
| **Misalignment (Coupling/Power Transmission)** | Misalignment | Pump vibration | Increased Wear and Tear | Incorrect installation | Perform coupling alignment process. |
| Motor vibration | Vibration and Noise | Uneven or unstable foundations | Foundation repair |
| Motor current | Reduced Efficiency | Component Wear (bearings, couplings, or shafts) | Replace worn component |
|  | Seal and Gasket Damage | Thermal Expansion or Contraction |  |
|  | Bearing Failures | Vibration or Impact |  |
|  | Shaft and Coupling Damage | Inaccurate Machinery Assembly |  |
|  |  | Lack of Maintenance |  |
|  |  | Environmental Factors (seismic activity, changes in terrain, humidity or corrosive elements) |  |
|  |  | Overloading or Stress |  |

Bảng mô tả nguyên nhân gây lỗi và khuyến cáo xử lý lỗi

|  |  |  |
| --- | --- | --- |
| **Sensor Data** | **Failure Mode** | **Formula** |
| Wet well Level (m) | Cavitation    Efficiency decrese | To calculate NPSHA: (Atmospheric presure/9,8)+(**Wet well Level**)-(System Friction Loss)-(Vapour Pressure/9,8)  To calculate Total Head: Discharge Pressure Head-**Wet Well Level**+System Friction Loss+System Dynamic Loss |
| Motor Power (kW) | Efficiency decrese   Misalignment | To Calculate Pump Efficiency: Hydrolic Power/**Motor Power** If Pump Vibration>5 mm/s AND Motor Vibration>4,5 mm/s AND **Motor Power**>105%\*Motor rated Power |
| Discharge Pressure (kPa) | Efficiency decrese | To calculate Discharge Pressure Head: 1000\***Discharge Pressure**/9,8/Fluid Density |
| Discharge FlowRate (L/s) | Efficiency decrese   Performance Decrease | To Calculate Hydrolic Power: (**Discharge Flow Rate**/1000) x Total Head x 9,8  If **Discharge Flow Rate**<Pump Flow Rate (BEP)\*95% |
| Motor Winding Temp R (℃) | Motor Overheat | If **Motor Winding Temp R**>Winding Temperature Limit |
| Motor Winding Temp S (℃) | Motor Overheat | If **Motor Winding Temp S**>Winding Temperature Limit |
| Motor Winding Temp T (℃) | Motor Overheat | If **Motor Winding Temp T**>Winding Temperature Limit |
| Motor Bearing Temp DE (℃) | Motor Bearing Overheat   Motor Bearing Defect | If **Motor Bearing Temp DE**>Motor Bearing Temperature Limit  If **Motor Bearing Temp DE**>Motor Bearing Temperature Limit AND Motor Vibration DE>Motor Bearing Vibration Limit |
| Motor Bearing Temp NDE (℃) | Motor Bearing Overheat   Motor Bearing Defect | If **Motor Bearing Temp NDE**>Motor Bearing Temperature Limit  If **Motor Bearing Temp NDE**>Motor Bearing Temperature Limit AND Motor Vibration NDE>Motor Bearing Vibration Limit |
| Pump Bearing Temp DE (℃) | Pumo Bearing Overheat   Pump Bearing Defect | If **Pump Bearing Temp DE**>Pump Bearing Temperature Limit  If **Pump Bearing Temp DE**>Pump Bearing Temperature Limit AND Motor Vibration DE>Pump Bearing Vibration Limit |
| Pump Bearing Temp NDE (℃) | Pumo Bearing Overheat   Pump Bearing Defect | If **Pump Bearing Temp NDE**>Pump Bearing Temperature Limit   If **Pump Bearing Temp NDE**>Pump Bearing Temperature Limit AND Motor Vibration NDE>Pump Bearing Vibration Limit |
| Motor Vibration DE (mm/s) | Motor Vibration  Motor Bearing Defect        Misalignment      Motor Unbalance   Motor Rotating Loseness | If **Motor Vibration DE**>Motor Bearing Vibration Limit  If Motor Bearing Temp DE>Motor Bearing Temperature Limit AND **Motor Vibration DE**>Motor Bearing Vibration Limit  To calculate Motor DE Non-Synchronous: FFT process (frequency of RPM Fractions and their multiples)  If Pump Vibration>5 mm/s AND **Motor Vibration DE**>4,5 mm/s AND Motor Power>105%\*Motor rated Power  To calculate Motor DE 2X: FFT process (frequency same with 2xRPM)  To calculate Motor DE 1X: FFT process (frequency same with RPM)  To calculate Motor DE Synchronous: FFT process (frequency same with RPM and its multiples) |
| Motor Vibration NDE (mm/s) | Motor Vibration  Motor Bearing Defect        Misalignment   Motor Unbalance   Motor Rotating Loseness | If **Motor Vibration NDE**>Motor Bearing Vibration Limit  If Motor Bearing Temp NDE>Motor Bearing Temperature Limit AND **Motor Vibration NDE**>Motor Bearing Vibration Limit  To calculate Motor NDE Non-Synchronous: FFT process (frequency of RPM Fractions and their multiples)  To calculate Motor NDE 2X: FFT process (frequency same with 2xRPM)  To calculate Motor NDE 1X: FFT process (frequency same with RPM)  To calculate Motor NDE Synchronous: FFT process (frequency same with RPM and its multiples) |
| Pump Vibration DE (mm/s) | Pump Vibration  Pump Bearing Defect        Misalignment      Pump Unbalance   Pump Rotating Loseness | If **Pump Vibration DE**>Pump Bearing Vibration Limit  If Pump Bearing Temp DE>Pump Bearing Temperature Limit AND **Pump Vibration DE**>Motor Bearing Vibration Limit  To calculate Pump DE Non-Synchronous: FFT process (frequency of RPM Fractions and their multiples)  If Pump Vibration>5 mm/s AND **Motor Vibration DE**>4,5 mm/s AND Motor Power>105%\*Motor rated Power  To calculate Pump DE 2X: FFT process (frequency same with 2xRPM)  To calculate Pump DE 1X: FFT process (frequency same with RPM)  To calculate Pump DE Synchronous: FFT process (frequency same with RPM and its multiples) |
| Pump Vibration NDE (mm/s) | Pump Vibration  Pump Bearing Defect        Misalignment   Pump Unbalance   Pump Rotating Loseness | If **Pump Vibration NDE**>Pump Bearing Vibration Limit  If Pump Bearing Temp NDE>Pump Bearing Temperature Limit AND **Pump Vibration NDE**>Pump Bearing Vibration Limit  To calculate Pump NDE Non-Synchronous: FFT process (frequency of RPM Fractions and their multiples)  To calculate Pump NDE 2X: FFT process (frequency same with 2xRPM)  To calculate Pump NDE 1X: FFT process (frequency same with RPM)  To calculate Pump NDE Synchronous: FFT process (frequency same with RPM and its multiples) |
| Fluid Inlet Temp (℃) | Cavitation | To calculate vapour pressure: Refer to water vapor pressure table |
| Discharge Pressure head (m) | Cavitation | To calculate Total Head: **(Discharge Pressure**-Wet Well Level)+System Friction Loss+System Dynamic Loss |
| Total Head (m) | Efficiency decrese | To Calculate Hydrolic Power: (Discharge Flow Rate/1000) x **Total Head** x 9,8 |
| Suction Pressue(kPa) | Cavitation |  |
| Inlet Temp Rounding number (℃) | Cavitation | To calculate vapour pressure: Refer to water vapor pressure table |
| Hydrolic Power (KW) | Efficiency decrese | To Calculate Pump Efficiency: **Hydrolic Power**/Motor Power |
| Pressure Vapour (kPa) | Cavitation | To calculate NPSHA: (Atmospheric presure/9,8)+(Wet well Level)-(System Friction Loss)-(**Vapour Pressure**/9,8) |
| NPSHa (m) | Cavitation | NPSHA<NPSHR |
| Pump Efficiency (%) | Efficiency decrese | If Standard Pump Efficiency-**Pump Efficiency**>3% |
| Motor DE FFT for 1X (mm/s) | Motor Unbalance | If **Motor DE FFT for 1X**>2,5 |
| Motor DE FFT for 2X (mm/s) | Misalignment | If **Motor DE FFT for 2X**>50%\*Motor Vibration DE |
| Motor DE Synchronous (mm/s) | Motor Rotating Loseness | If **Motor DE Synchronous**>50%\*Motor Vibration DE |
| Motor DE Non-Synchronous (mm/s) | Motor Bearing Defect | If **Motor DE Non-Synchronous**>50%\*Motor Vibration DE |
| Motor NDE FFT for 1X (mm/s) | Motor Unbalance | If **Motor NDE FFT for 1X**>2,5 |
| Motor NDE FFT for 2X (mm/s) | Misalignment | If **Motor NDE FFT for 2X**>50%\*Motor Vibration NDE |
| Motor NDE Synchronous (mm/s) | Motor Rotating Loseness | If **Motor NDE Synchronous**>50%\*Motor Vibration NDE |
| Motor NDE Non-Synchronous (mm/s) | Motor Bearing Defect | If **Motor NDE Non-Synchronous**>50%\*Motor Vibration NDE |
| Pump DE FFT for 1X (mm/s) | Pump Unbalance | If **Pump DE FFT for 1X**>2,5 |
| Pump DE FFT for 2X (mm/s) | Misalignment | If **Pump DE FFT for 2X**>50%\*Pump Vibration DE |
| Pump DE Synchronous (mm/s) | Pump Rotating Loseness | If **Pump DE Synchronous**>50%\*Pump Vibration DE |
| Pump DE Non-Synchronous (mm/s) | Pump Bearing Defect | If **Pump DE Non-Synchronous**>50%\*Pump Vibration DE |
| Pump NDE FFT for 1X (mm/s) | Pump Unbalance | If **Pump NDE FFT for 1X**>2,5 |
| Pump NDE FFT for 2X (mm/s) | Misalignment | If **Pump NDE FFT for 2X**>50%\*Pump Vibration NDE |
| Pump NDE Synchronous (mm/s) | Pump Rotating Loseness | If **Pump NDE Synchronous**>50%\*Pump Vibration NDE |
| Pump NDE Non-Synchronous (mm/s) | Pump Bearing Defect | If **Pump NDE Non-Synchronous**>50%\*Pump Vibration NDE |

Bảng phân loại và công thức tính toán lỗi liên quan đến các cảm biến

**Mối liên hệ giữa các chỉ số và bất thường**

|  |  |
| --- | --- |
| **Dạng bất thường** | **Biểu hiện có thể thấy từ dữ liệu** |
| Mất cân bằng | Tăng mạnh ở FFT 1X, đặc biệt tại Motor hoặc Pump DE. |
| Hỏng ổ trục | Nhiệt độ Bearing tăng, rung động Non-Synchronous tăng bất thường. |
| Tắc nghẽn hoặc rò rỉ | Suction Pressure giảm, nhưng Motor Power hoặc nhiệt độ vẫn cao. |
| Quá tải | Motor Power và Winding Temp tăng, có thể đi kèm FlowRate cao. |
| Mất pha hoặc sự cố điện | Nhiệt độ R/S/T không đồng đều hoặc giảm dòng chảy bất thường. |

Tiền xử lý & chọn đặc trưng cho mô hình

**Phân loại các nhóm feature ( áp dụng cho cả 3 mô hình IF, VAE, GAN)**

|  |  |  |
| --- | --- | --- |
| **Nhóm** | **Cột ví dụ** | **Giải thích** |
| Áp suất & lưu lượng | Suction Pressure, Discharge Pressure, Discharge FlowRate | Liên quan trực tiếp đến hiệu suất và hoạt động bất thường của máy. |
| Công suất & nhiệt độ motor | Motor Power, Motor Winding Temp R/S/T, Motor Bearing Temp DE/NDE | Cho thấy máy chạy nặng, mất cân bằng, hoặc sắp hỏng. |
| Rung động tổng quát | Motor/Pump Vibration DE/NDE | Rung cao thường báo trước lỗi cơ học. |
| Nhiệt độ bơm | Pump Bearing Temp DE/NDE | Quá nhiệt có thể liên quan đến mất bôi trơn, lệch trục. |

**Feature nâng cao (nên dùng cho VAE/GAN nếu muốn tăng độ nhạy)**

|  |  |  |
| --- | --- | --- |
| **Nhóm** | **Cột ví dụ** | **Giải thích** |
| FFT rung động | FFT 1X, 2X, Synchronous, Non-Synchronous tại DE/NDE | Giúp mô hình học rõ các dạng lệch tâm, mất cân bằng, hỏng ổ trục. |
| Nhiệt đầu vào | Fluid Inlet Temp | Góp phần đánh giá điều kiện đầu vào thay đổi bất thường. |
| Đầu hút/xả theo mét | Discharge Head, Suction Head, Total Head | Cho biết công suất thực tế bơm đạt được. |

**Isolation Forest**

* Đặc trưng gợi ý: Motor Power, Suction/Discharge Pressure, tất cả rung động DE/NDE.
* Ưu điểm: Không cần gán nhãn, phát hiện outliers theo khoảng cách/độ hiếm.
* Lưu ý: Chuẩn hóa dữ liệu (MinMaxScaler / RobustScaler).

**VAE (Variational Autoencoder)**

* Đặc trưng gợi ý: Full feature set (không bỏ biến nào), vì VAE học phân phối tổng thể.
* Ưu điểm: Mô hình hóa sự “bình thường”, điểm bất thường có MSE cao.
* Lưu ý: Phải train bằng dữ liệu bình thường (không chứa abnormal).

**GAN (Anomaly GAN / BiGAN)**

* Đặc trưng gợi ý: Như VAE.
* Ưu điểm: Phát hiện các điểm dữ liệu không thể sinh ra (out-of-distribution).
* Lưu ý: Dữ liệu phải có tính thời gian hoặc mối quan hệ phức tạp mới thấy rõ sức mạnh của GAN.

Tổng quan mục tiêu mô hình

|  |  |  |  |
| --- | --- | --- | --- |
| **Mô hình** | **Loại** | **Ưu điểm** | **Yêu cầu dữ liệu** |
| **Isolation Forest** | Unsupervised | Nhanh, mạnh cho phát hiện outlier | Không yêu cầu dữ liệu time-series |
| **VAE** | Semi-supervised / Unsupervised | Học biểu diễn "bình thường", phát hiện bằng lỗi tái tạo | Yêu cầu chuẩn hóa & dữ liệu sạch |
| **GAN (Anomaly GAN)** | Semi-supervised / Unsupervised | Phát hiện điểm dữ liệu không thể sinh ra | Cần nhiều dữ liệu bình thường, tránh nhiễu |

4.4 Phân tích dữ liệu

A screenshot of a graph

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.

A screenshot of a computer screen

AI-generated content may be incorrect.

A screenshot of a computer screen

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer screen

AI-generated content may be incorrect.

A screen shot of a graph

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.

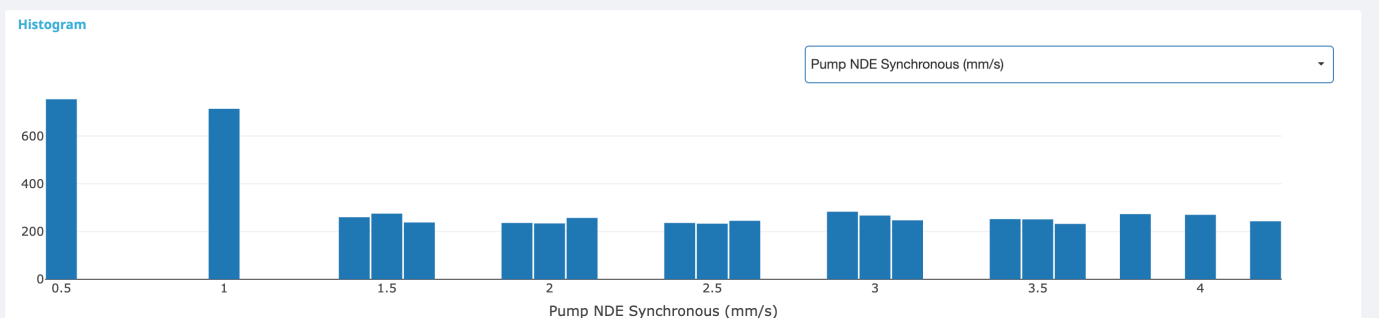
A screenshot of a data graph

AI-generated content may be incorrect.

A screenshot of a graph

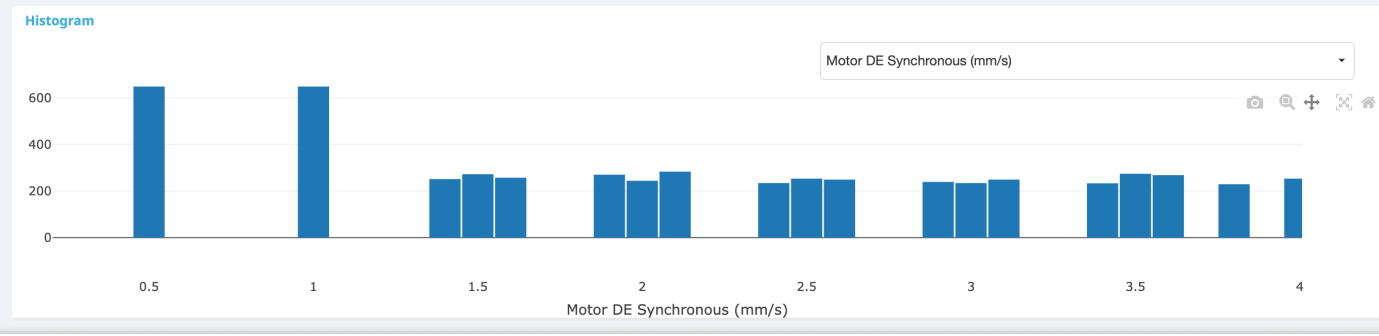
AI-generated content may be incorrect.

Histogram phân phối dữ liệu



A graph with blue bars

AI-generated content may be incorrect.



A screenshot of a graph

AI-generated content may be incorrect.

A graph with blue bars

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.

PCA

A green ball of dots

AI-generated content may be incorrect.

A graph with blue and white text

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

A graph of a number of machines

AI-generated content may be incorrect.

Modeling Results

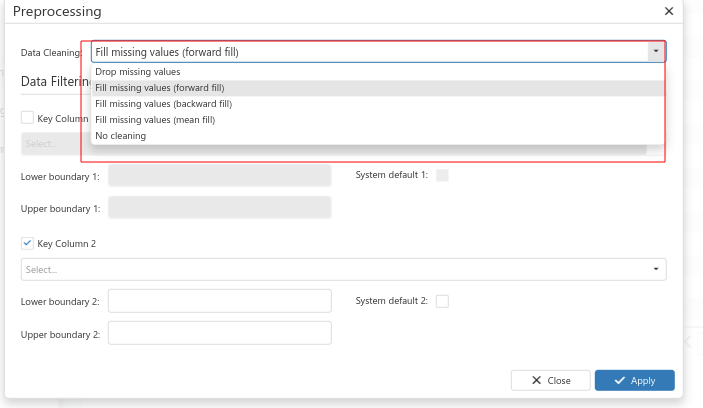
A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

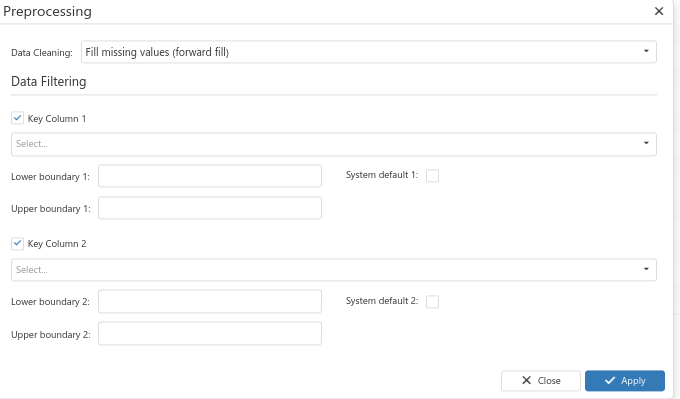
### **4.1.2. Tiền xử lý dữ liệu**



Tùy chọn "Data Cleaning" (Làm sạch dữ liệu)

|  |  |  |
| --- | --- | --- |
| **Tùy chọn** | **Giải thích** | **Khi nào nên dùng?** |
| **Drop missing values** | Xóa toàn bộ dòng dữ liệu có giá trị bị thiếu (NaN). | Khi chắc chắn rằng thiếu dữ liệu là không chấp nhận được hoặc chỉ thiếu rất ít. |
| **Fill missing values (forward fill)** | Dùng giá trị trước đó để điền vào ô bị thiếu. | Phù hợp với dữ liệu theo thời gian (time-series), giá trị thường thay đổi liên tục. |
| **Fill missing values (backward fill)** | Dùng giá trị sau đó để điền vào ô bị thiếu. | Khi cần lấp dữ liệu mà phía sau gần hơn hoặc đáng tin hơn. |
| **Fill missing values (mean fill)** | Tự động điền giá trị trung bình của cột. | Phù hợp khi dữ liệu có phân phối đều và không có xu hướng rõ ràng theo thời gian. |
| **No cleaning** | Không làm gì cả – giữ nguyên dữ liệu như ban đầu. | Dùng khi dữ liệu đã được làm sạch trước hoặc bạn muốn tự xử lý sau bằng script. |

Data Filtering – Bộ lọc dữ liệu



Giao diện này cho phép chọn 1 hoặc 2 cột dữ liệu (Key Columns) để lọc ra các giá trị nằm ngoài ngưỡng cho phép. Điều này cực kỳ quan trọng trong bài toán phát hiện bất thường vì:

* Loại bỏ các outlier vật lý không hợp lệ (do lỗi cảm biến, nhập sai, hỏng thiết bị).
* Giữ lại tập dữ liệu ổn định hơn để mô hình học chuẩn hơn.

|  |  |
| --- | --- |
| **Thành phần** | **Mô tả** |
| ✅ Key Column 1 / Key Column 2 | Chọn các **cột giá trị quan trọng** cần giới hạn (VD: Discharge Pressure, Motor Power, Vibration,...) |
| **Lower boundary** | Ngưỡng dưới: giá trị nhỏ nhất được chấp nhận. |
| **Upper boundary** | Ngưỡng trên: giá trị lớn nhất được chấp nhận. |
| System default | Nếu tick vào, phần mềm sẽ tự động ước tính ngưỡng từ phân phối dữ liệu, thường dùng z-score hoặc IQR. |

## **4.2. Huấn luyện mô hình**

### **4.2.1. Mô hình Isolation Forest**

* **Thư viện:** scikit-learn
* **Thông số chính:**
  + n\_estimators = 100
  + contamination = 0.01 (tỷ lệ bất thường)
* **Chiến lược đánh giá:**
  + Tính anomaly score
  + Cắt ngưỡng dựa trên phân vị 99%

### **4.2.2. Mô hình VAE**

* **Thư viện:** TensorFlow / Keras
* **Cấu trúc mạng:**
  + Encoder: Dense(128) → Dense(64) → latent space (z)
  + Decoder: Dense(64) → Dense(128) → Output
* **Tham số huấn luyện:**
  + Optimizer: Adam
  + Batch size: 32
  + Epochs: 100
* **Chỉ số đánh giá:** Reconstruction error  
   RE(x)=∥x−x^∥2\text{RE}(x) = \|x - \hat{x}\|^2RE(x)=∥x−x^∥2

### **4.2.3. Mô hình GAN**

* **Thư viện:** Keras
* **Cấu trúc:**
  + Generator: Mạng fully connected
  + Discriminator: Mạng phân loại nhị phân
* **Quá trình huấn luyện:**
  + Chỉ huấn luyện với dữ liệu "bình thường"
  + Đánh giá mẫu mới dựa trên khả năng sinh và phân biệt
* **Loss tổng hợp:** Loss=α⋅Reconstruction Loss+β⋅Discriminator Score\text{Loss} = \alpha \cdot \text{Reconstruction Loss} + \beta \cdot \text{Discriminator Score}Loss=α⋅Reconstruction Loss+β⋅Discriminator Score

## **4.3. Đánh giá và so sánh kết quả**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Mô hình** | **Ưu điểm** | **Nhược điểm** | **Tốc độ xử lý** | **Độ chính xác tương đối** |
| Isolation Forest | Nhanh, không cần nhãn | Kém với chuỗi phức tạp | ★★★★☆ | ★★★☆☆ |
| VAE | Nhận diện tốt chuỗi thời gian | Cần dữ liệu sạch, nhiều epoch | ★★☆☆☆ | ★★★★☆ |
| GAN | Sinh dữ liệu, mạnh với imbalance | Khó huấn luyện | ★☆☆☆☆ | ★★★★★ |

Ghi chú: Độ chính xác ở đây được đo lường tương đối qua **phân tích số điểm phát hiện đúng** và **kiểm tra thủ công** trong tập dữ liệu được gán nhãn sơ bộ.

## **4.4. Triển khai hệ thống**

* **Kết nối cảm biến → MQTT broker**
* **Tiền xử lý dữ liệu real-time → InfluxDB**
* **ML Service** gọi các mô hình để phân tích
* **Giao diện Web:**
  + Biểu đồ real-time
  + Cảnh báo màu đỏ/vàng/xanh
  + Đề xuất bảo trì định kỳ khi bất thường liên tục xảy ra

# **CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

## **5.1. Kết luận**

Đề tài **"Phát hiện bất thường trong máy móc công nghiệp sử dụng các thuật toán máy học"** đã tập trung vào việc ứng dụng ba mô hình phổ biến trong lĩnh vực học máy để nhận diện các bất thường trong dữ liệu cảm biến công nghiệp. Thông qua quá trình khảo sát, tiền xử lý, huấn luyện và triển khai, một số kết quả đáng chú ý đã đạt được:

* ✅ **Mô hình Isolation Forest** cho khả năng phát hiện nhanh, phù hợp với dữ liệu tabular và môi trường yêu cầu phản hồi thời gian thực. Tuy nhiên, khả năng biểu diễn mối quan hệ phức tạp của chuỗi thời gian còn hạn chế.
* ✅ **Mô hình VAE** cho phép học đặc trưng tiềm ẩn của dữ liệu và phát hiện các sai lệch thông qua lỗi tái tạo. Đây là một lựa chọn hiệu quả với dữ liệu chuỗi thời gian, đặc biệt trong các hệ thống có sự dao động nhỏ.
* ✅ **Mô hình GAN** thể hiện hiệu quả vượt trội khi làm việc với dữ liệu mất cân bằng, đồng thời có khả năng sinh dữ liệu để bổ sung cho các trường hợp thiếu dữ liệu huấn luyện. Tuy nhiên, nhược điểm của GAN là khó huấn luyện và cần nhiều tài nguyên.
* Hệ thống phát hiện bất thường đã được tích hợp với nền tảng IoT sử dụng **MQTT**, **InfluxDB**, và **giao diện Web** để hiển thị dữ liệu và cảnh báo. Thử nghiệm cho thấy hệ thống có thể xử lý hơn **100.000 điểm dữ liệu** với **thời gian xử lý dưới 1 giây**, đáp ứng yêu cầu thời gian thực trong công nghiệp.

## **5.2. Hạn chế**

Mặc dù đạt được nhiều kết quả khả quan, đề tài vẫn tồn tại một số hạn chế:

* ⚠️ **Chưa tích hợp mô-đun dự đoán thời điểm hỏng hóc**, vốn là phần quan trọng của bảo trì dự đoán (đã được triển khai riêng trong dự án công ty).
* ⚠️ **Việc lựa chọn ngưỡng phát hiện bất thường** còn mang tính thủ công, chưa có chiến lược tối ưu hóa tự động.
* ⚠️ **Thiếu đánh giá định lượng hoàn chỉnh** như Precision, Recall, F1-Score do không có tập dữ liệu gán nhãn đầy đủ.

## **5.3. Hướng phát triển**

Trong tương lai, đề tài có thể được mở rộng theo các hướng sau:

### **🔮 1. Kết hợp phát hiện với dự đoán hỏng hóc**

* Phát triển mô hình hybrid vừa phát hiện bất thường, vừa dự đoán thời điểm máy móc có thể bị hỏng (Remaining Useful Life – RUL).

### **🔧 2. Tối ưu hóa lựa chọn ngưỡng**

* Áp dụng kỹ thuật học có giám sát hoặc Bayesian Optimization để tự động xác định ngưỡng phân loại bất thường phù hợp với từng thiết bị.

### **☁️ 3. Triển khai hệ thống trên nền tảng đám mây**

* Dùng các dịch vụ như AWS IoT, Google Cloud Platform để mở rộng quy mô, hỗ trợ nhiều điểm đo cùng lúc và phục vụ nhiều nhà máy từ xa.

### **🧠 4. Học liên kết (Federated Learning)**

* Cho phép huấn luyện mô hình mà không cần tập trung dữ liệu – một xu hướng phù hợp với dữ liệu phân tán trong công nghiệp và có yêu cầu bảo mật cao.

# **✅ Tổng kết**

Đề tài đã hoàn thành đầy đủ các mục tiêu đặt ra:

* Xây dựng và đánh giá ba mô hình phát hiện bất thường.
* Tích hợp mô hình vào hệ thống IoT trong môi trường công nghiệp.
* Đưa ra các cảnh báo kịp thời phục vụ hoạt động bảo trì.

Kết quả nghiên cứu có giá trị thực tiễn cao và tiềm năng mở rộng cho nhiều lĩnh vực sản xuất và công nghiệp 4.0 tại Việt Nam 🇻🇳.