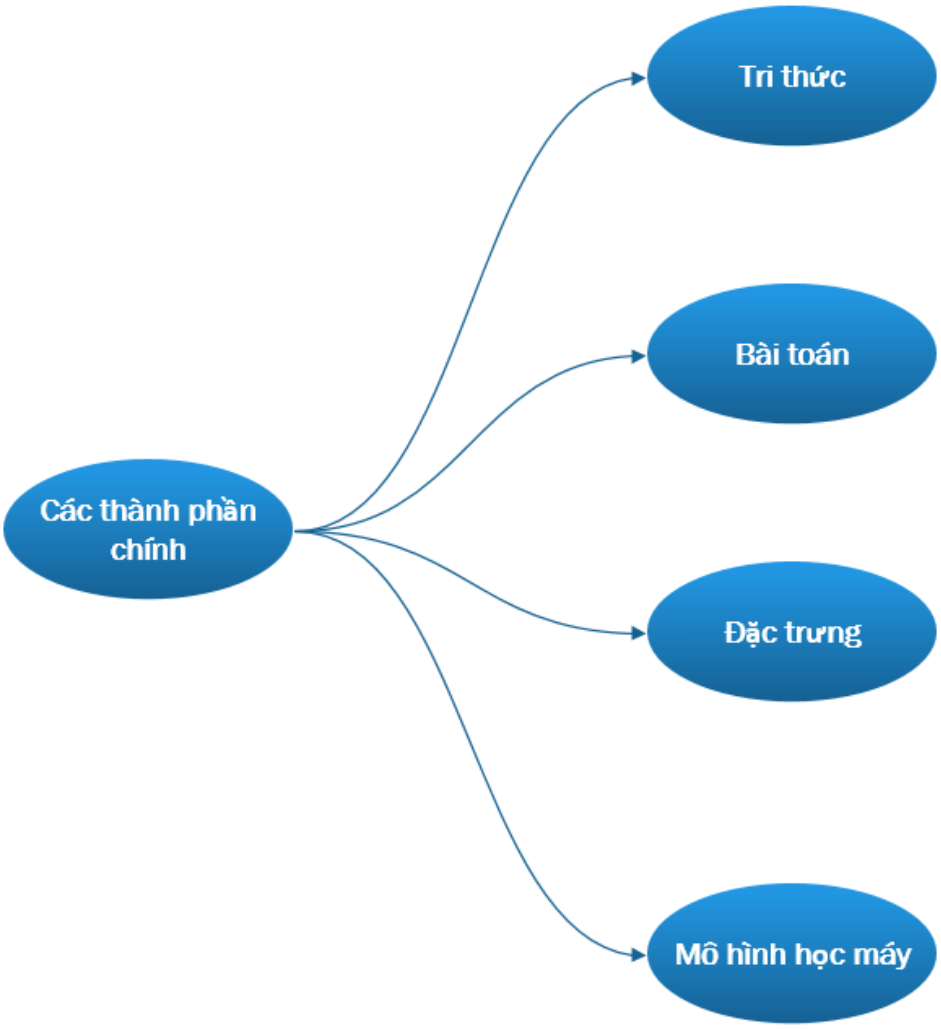
Interactive Machine Learning Framework

Ngày 17 tháng 12 năm 2018

1 Giới thiệu chung

IMLF được xây dựng nhằm mục đích quản lý các tri thức do người sử dụng tạo ra. Thông qua quá trình tương tác giữa người dùng và hệ thống, các tri thức sẽ được chuẩn hóa để tạo nên một hệ tri thức thống nhất, liên kết chặt chẽ với nhau. Những tri thức này sẽ được sử dụng để xây dựng các mô hình học máy giải quyết các bài toán mà người dùng đặt ra. Dưới đây là một số định nghĩa về các thành phần chính trong hệ thống.



Hình 1: Các thành phần chính của hệ thống IMLF

IMLF gồm 4 thành phần chính:

- Tri thức (Knowlegde): Là một mapping (X, y), X là dữ liệu đầu vào (xâu, file, event), y là nhán tương ứng. Ví dụ, đối với bài toán phát hiện malware, X là một file, y là nhãn tương ứng - malware hoặc benign.
- Bài toán (Problem): Là nhóm các đơn vị tri thức cũng biểu diễn một loại ánh xạ. Mỗi bài toán sẽ có một số kiểu dữ liệu đầu vào và nhãn tương ứng cố định.
- Đặc trưng (Feature): Là vector biểu diễn một đơn vị tri thức.
- Mô hình học máy (MLM): Là một chuỗi biến đổi toán học có khả năng khái quát, tổng hợp các tri thức đã biết, tạo ra tri thức mới.

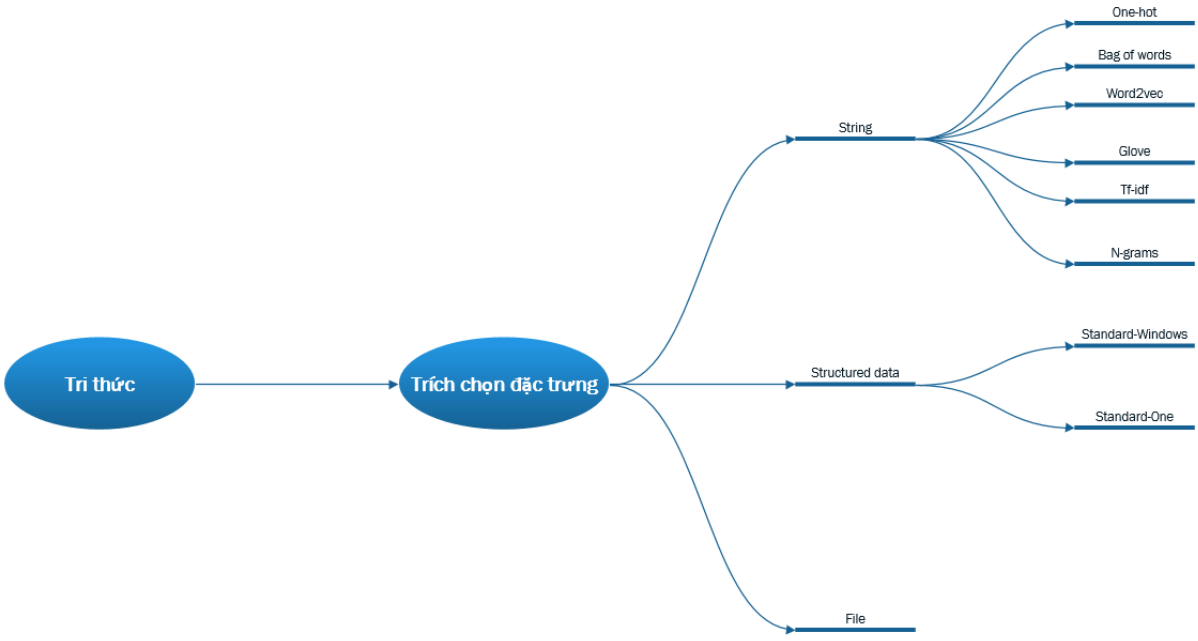
2 Các chức năng của hệ thống IMLF

IMLF có các chức năng sau đây:

- Lưu trữ, chuẩn hóa các tri thức.
- Tương tác với người dùng, bổ sung tri thức mới một cách đúng đắn.
- Tạo ra các tri thức mới.
- Tìm kiếm tri thức, sửa đổi tri thức.
- Xây dựng, cập nhật mô hình học máy một cách tự động.

2.1 Lưu trữ, chuẩn hóa tri thức

Các tri thức do người dùng tạo ra cần được chuẩn hóa để lưu trữ cũng như phục vụ cho việc xây dựng mô hình học máy và tìm kiếm một cách hiệu quả.



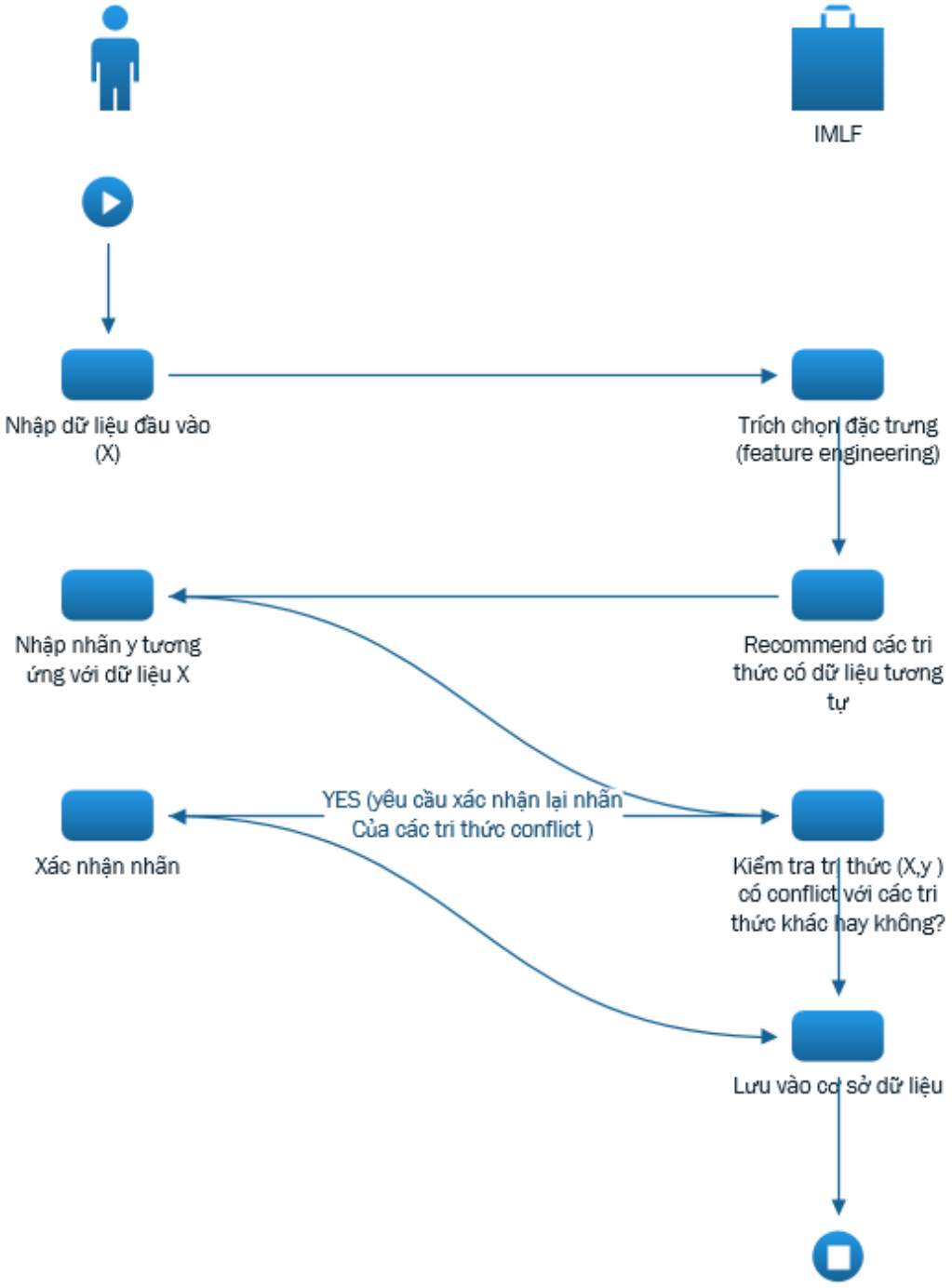
Hình 2: Vector hóa tri thức

Mỗi tri thức sẽ được vector hóa thông qua hệ thống **Trích chọn đặc trưng** (feature engineering). Với từng loại dữ liệu đầu vao khác nhau sẽ được vector hóa bằng các phương pháp khác nhau. Sau quá trình trình chọn đặc trưng, các tri thức này sẽ được lưu trữ dưới dạng vector đặc trưng.

2.2 Bổ sung tri thức mới

Chức năng này hỗ trợ người dùng trong quá trình bổ sung tri thức mới .Người dùng sẽ tương tác với hệ thống để tạo ra tri thức mới có ý nghĩa. Với mỗi tri thức mới được bổ sung, hệ thống sẽ tự động kiểm tra sự ảnh hưởng của tri thức mới với các tri thức đã được xây dựng từ trước. Có hai trường hợp xảy ra:

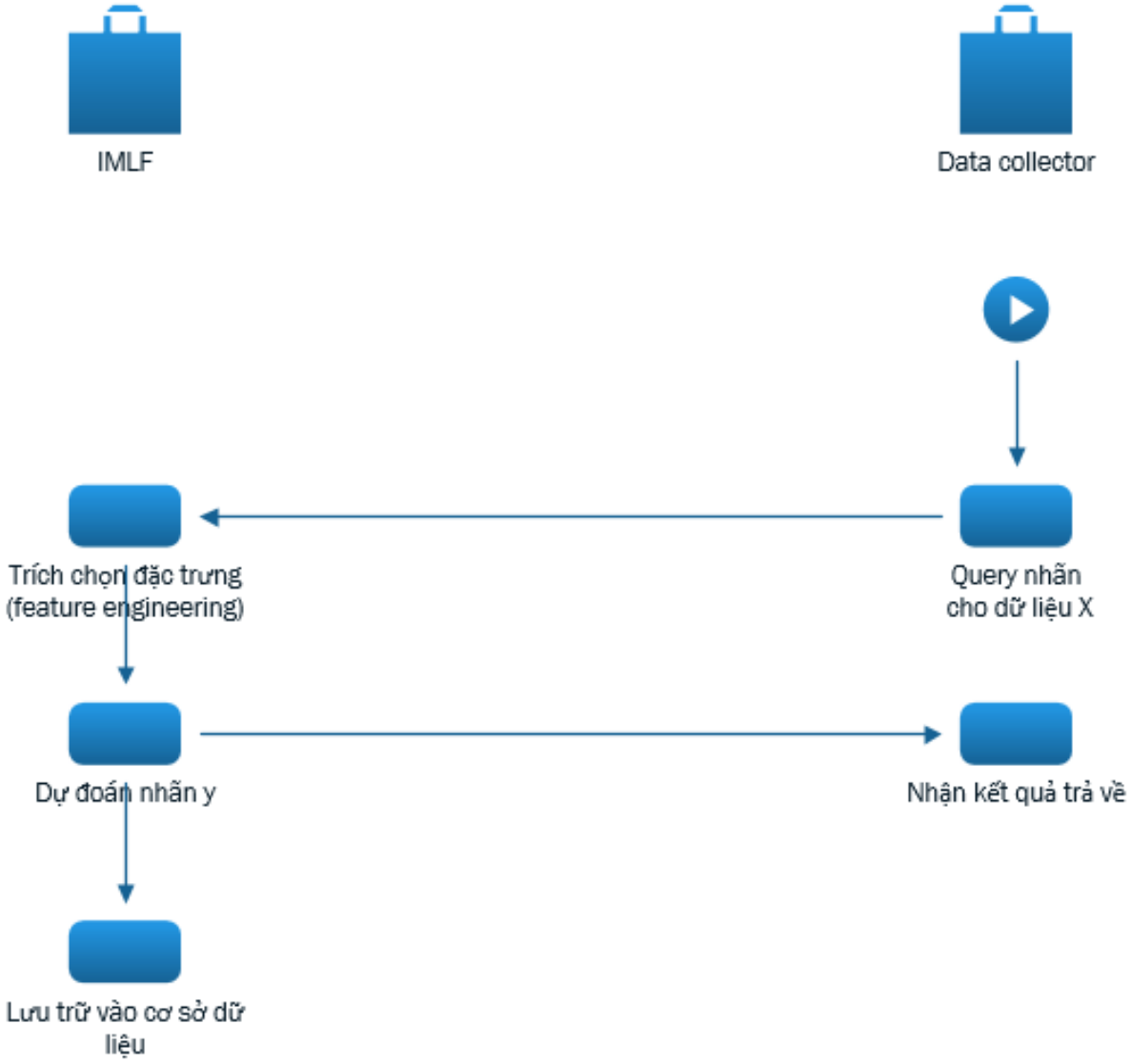
- TH1: Tri thức mới không xung đột (conflict) với các tri thức cũ - Hệ thống tự động cập nhật tri thức mới vào cơ sở dữ liệu.
- TH2: Tri Thức mới xung đột (conflict) với các tri thức cũ - Hệ thống yêu cầu người dùng xem xét lại tri thức mới cũng như các tri thức liên quan để tạo nên sự thống nhất giữa các tri thức.



Hình 3: Thêm tri thức mới

2.3 Tạo ra tri thức mới

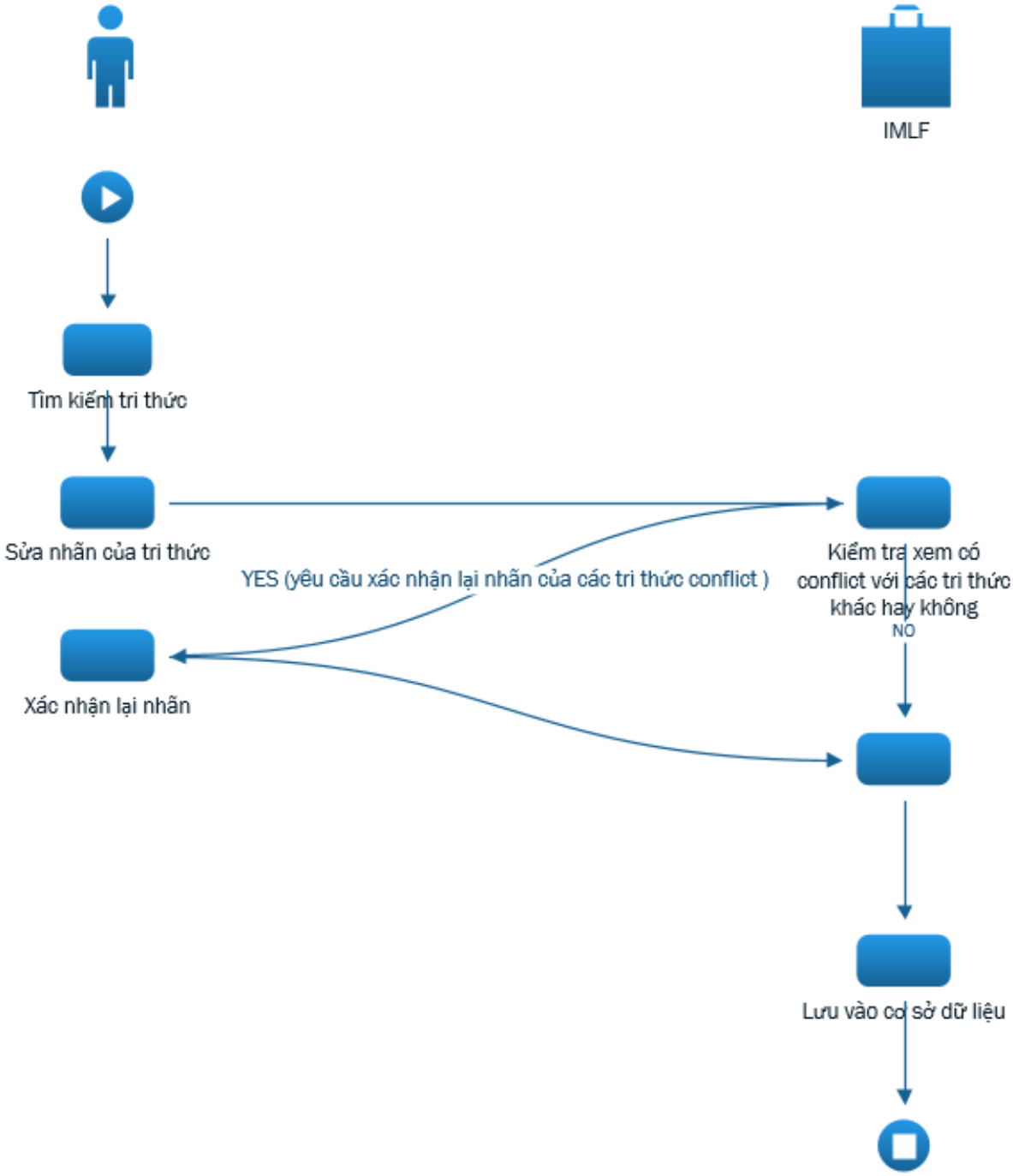
Khác với **Bổ sung tri thức mới** 2.2 - do người dùng thêm một tri thức mới, ở chức năng này, hệ thống sẽ tạo ra tri thức mới dựa trên dòng dữ liệu đầu vào được cung cấp bởi *Data Collector*. *Data Collector* - hình 4 là một hệ thống cung cấp dữ liệu đầu vào một cách real-time, ví dụ hệ thống logging. Dựa trên những tri thức đã được tạo ra trong quá khứ, IMLF sẽ xây dựng một mô hình học máy có khả năng khái quát, tổng hợp các tri thức này. Với mỗi đơn vị dữ liệu từ *Data Collector*, mô hình sẽ dự đoán nhãn tương ứng và cập nhật tri thức mới vào cơ sở dữ liệu.



Hình 4: Tạo tri thức mới

2.4 Tìm kiếm, chỉnh sửa tri thức

Chức năng này hỗ trợ người dùng tìm kiếm và chỉnh sửa tri thức. Người dùng có thể tìm kiếm các tri thức được tạo ra bởi hệ thống trong một khoảng thời gian hoặc tìm kiếm tập các tri thức tương tự với một dữ liệu đầu vào. Nếu cho rằng tri thức chưa đúng đắn, người dùng có thể sửa đổi nhãn của tri thứ đó. Sau khi hoàn thành việc sửa nhãn, hệ thống sẽ kiểm tra sử ảnh hưởng của hành động này với các tri thức trong cơ sở dữ liệu và phản hồi lại cho người dùng xử lý. Sau đó, các tri thức được sửa đổi sẽ được cập nhật lại vào cơ sở dữ liệu.



Hình 5: Tìm kiếm và chỉnh sửa tri thức

2.5 Xây dựng, cập nhật mô hình

Đây là chức năng quan trọng nhất của hệ thống, là nền tảng cho các chức năng đã được trình bày ở trên. Với một tập tri thức được cung cấp, hệ thống sẽ xây dựng một mô hình học máy có khả năng khái quát, tổng hợp các tri thức này. Mô hình sẽ được cập nhật dựa trên những tri thức mới được người dùng bổ sung và những tri thức được người dùng chỉnh sửa.

3 Một số vấn đề khó khăn cần giải quyết

Dưới đây là một số khó khăn cần làm rõ:

- Có nên fix cứng mỗi bài toán sẽ tương ứng với 1 loại dữ liệu hay không? Ví dụ:
  - Với bài toán Malware detection, dữ liệu đầu vào là 1 file.
  - Với bài toán Entity classification, dữ liệu đầu vào là một văn bản text (string).
  - Với bài toán Anomaly detection, dữ liệu đầu vào là một tập các event.
- Lưu trữ các vector đặc trưng như thế nào? Dùng database gì? Dùng công nghệ nào để tìm kiếm các phần tử similar một cách nhanh nhất (faiss)?
- Ở chức năng bổ sung tri thức mới 2.2, Hệ thống sẽ chỉ đơn thuần recommend các tri thức tương tự tri thức mới hay recommend cái gì?
- Khi nào thì một tri thức mới conflict với các tri thức khác? (các tri thức xung quanh tri thức mới không cùng nhãn với tri thức mới)
- Đối với 1 bài toán khi có thêm label mới chưa từng xuất hiện trong các tri thức cũ thì việc cập nhật mô hình được thực hiện như thế nào? Có thể dựa trên mô hình cũ hay không?
- Khi số lượng tri thức lớn lên, cần có chiến lược sampling tri thức phục vụ cho việc xây dựng và cập nhật mô hình. Việc cập nhật mô hình phải đảm bảo mô hình mới không mất đi những tri thức cũ (catastrophic forgetting) và phải cover thêm tri thức mới.

Tài liệu