**ĐỀ TÀI "TÌM KIẾM ẢNH DỰA TRÊN ĐỒ THỊ TRI THỨC VÀ ĐỐI TƯỢNG LÁNG GIỀNG"**

**I. Mục tiêu của đề tài**

* Xây dựng một hệ thống tìm kiếm ảnh sử dụng đồ thị tri thức để biểu diễn mối quan hệ giữa các đối tượng trong ảnh.
* Tăng cường độ chính xác bằng cách kết hợp phương pháp dựa trên đối tượng láng giềng gần nhất (K-Nearest Neighbors - KNN).

**II. Các bước thực hiện**

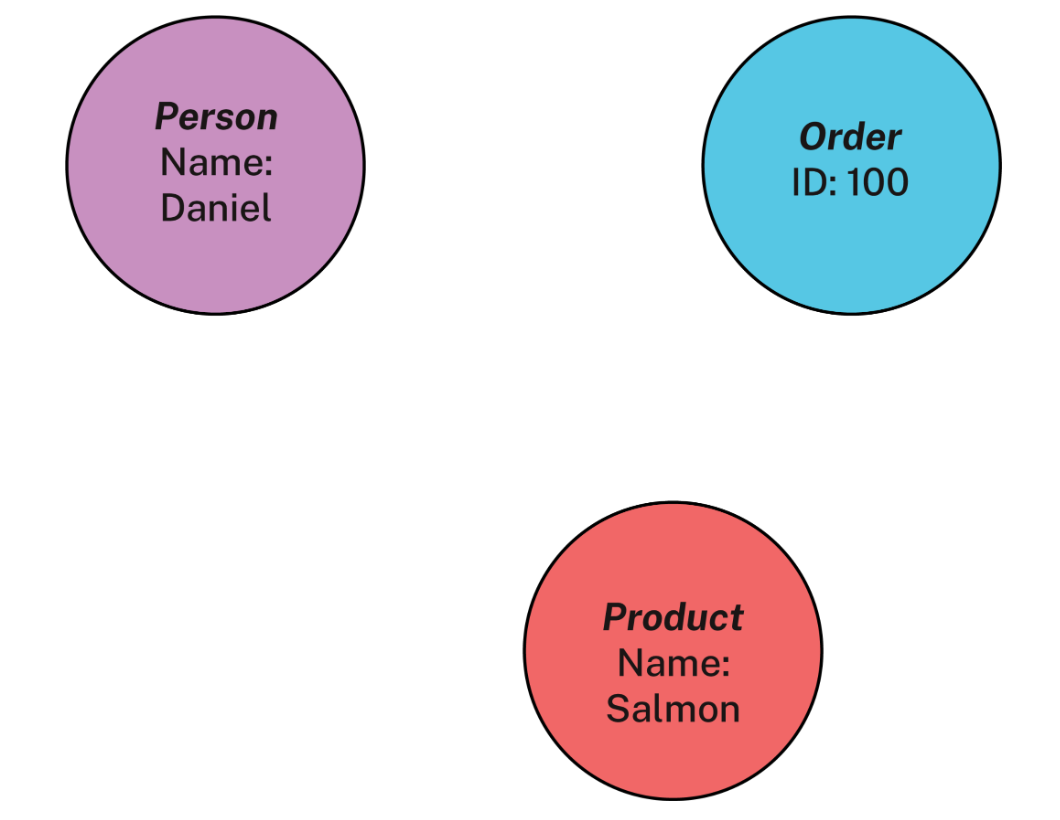
**1. Tìm hiểu lý thuyết nền tảng**

* **Đồ thị tri thức (Knowledge Graph):**
  + *Hiểu cấu trúc biểu diễn tri thức thông qua các thực thể (entities), quan hệ (relationships), và thuộc tính (attributes).*

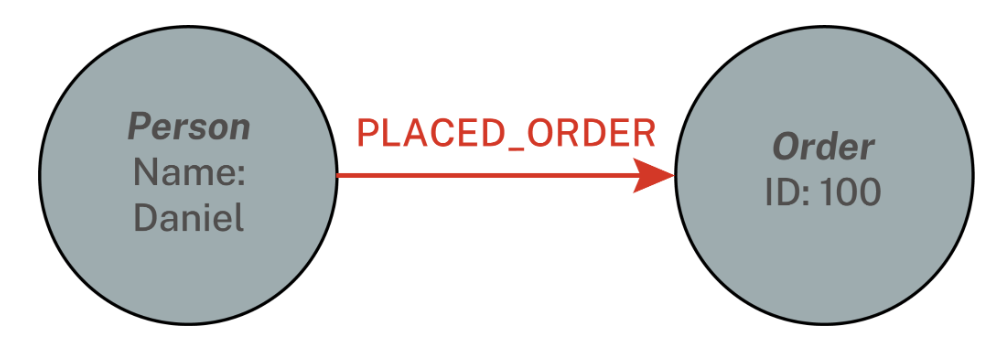
Đồ thị tri thức là một biểu diễn có tổ chức của các thực thể trong thế giới thực và các mối quan hệ của chúng. Biểu đồ này thường được lưu trữ trong cơ sở dữ liệu biểu đồ, nơi lưu trữ các mối quan hệ giữa các thực thể dữ liệu. Các thực thể trong biểu đồ kiến ​​thức có thể biểu diễn các đối tượng, sự kiện, tình huống hoặc khái niệm. Các mối quan hệ giữa các thực thể này nắm bắt bối cảnh và ý nghĩa về cách chúng được kết nối.

Nó giúp thể hiện cách dữ liệu liên kết với nhau, mang lại ngữ cảnh và ý nghĩa. Được sử dụng rộng rãi trong AI, tìm kiếm thông tin, và phân tích dữ liệu.

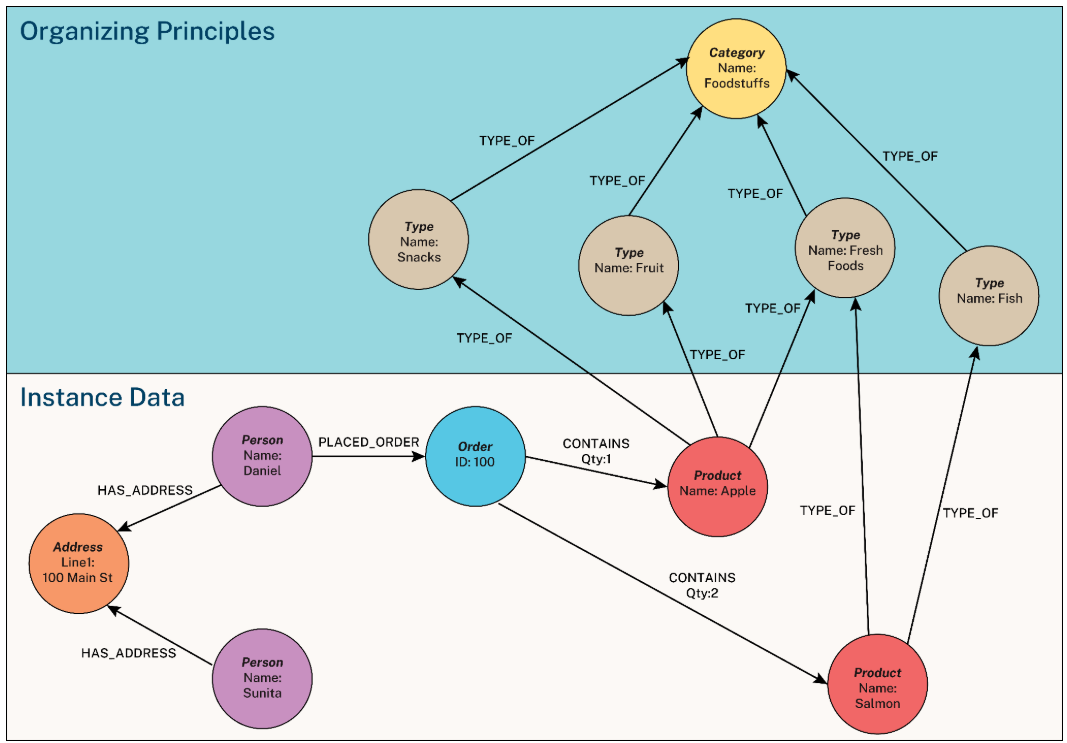
Thực thể (Entities) biểu thị và lưu trữ thông tin chi tiết về con người, địa điểm, đối tượng hoặc tổ chức. Mỗi thực thể là một nút có một (hoặc đôi khi là nhiều) nhãn để xác định loại nút và tùy chọn có thể có một hoặc nhiều thuộc tính (attributes). Các nút đôi khi cũng được gọi là đỉnh. Ví dụ các thực thể trong biểu đồ tri ​​thức thương mại điện tử thường biểu diễn các thực thể như khách hàng, sản phẩm và đơn hàng:



Quan hệ (relationships) xác định cách các thực thể liên kết với nhau. Giống như các nút, mỗi mối quan hệ có một nhãn xác định loại mối quan hệ và có thể tùy chọn có một hoặc nhiều thuộc tính. Mối quan hệ đôi khi cũng được gọi là cạnh. Trong ví dụ thương mại điện tử, tồn tại mối quan hệ giữa khách hàng và các nút đặt hàng, nắm bắt mối quan hệ “đã đặt hàng” giữa khách hàng và đơn hàng của họ:



Thuộc tính (Attributes): Chứa thông tin bổ sung về thực thể và mối quan hệ.



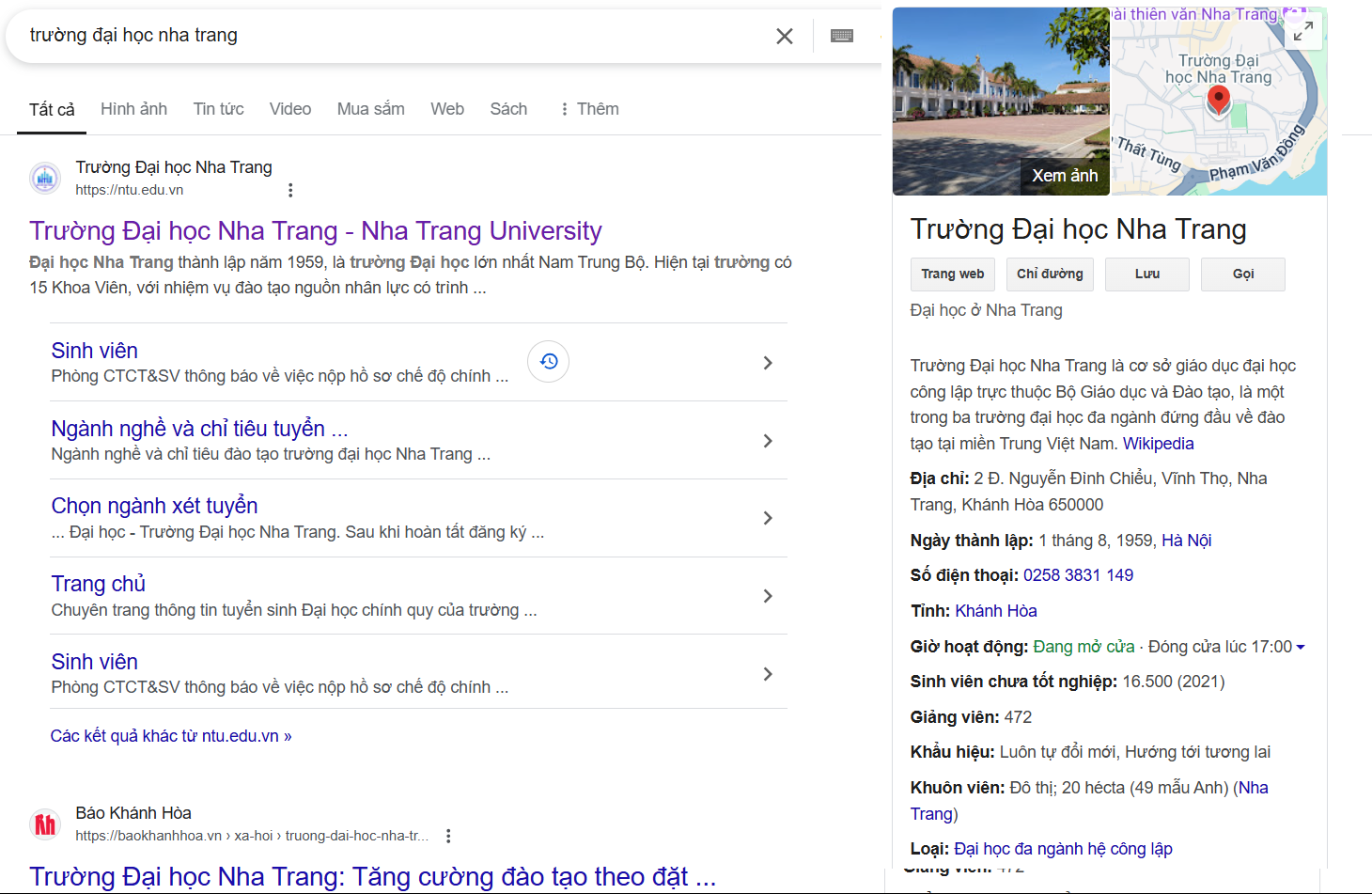
Organizing Principles (Nguyên tắc tổ chức): thể hiện cấu trúc phân loại của các loại sản phẩm:

* Category (Danh mục): Foodstuffs
* Types (Loại sản phẩm): Snacks, Fruit, Fresh Foods, Fish
* Các loại sản phẩm này được liên kết bằng quan hệ TYPE\_OF, cho thấy thực phẩm có nhiều loại con như trái cây, cá, đồ ăn vặt.

Instance Data (Dữ liệu thực tế): thể hiện dữ liệu cụ thể về khách hàng, đơn hàng, địa chỉ và sản phẩm:

* Khách hàng (Person): Daniel và Sunita. Cả hai có quan hệ HAS\_ADDRESS với cùng một địa chỉ 100 Main St.
* Đơn hàng (Order): Daniel đã đặt một đơn hàng (PLACED\_ORDER) có ID: 100.
* Đơn hàng này chứa sản phẩm Apple (1 đơn vị) và Salmon (2 đơn vị), thể hiện bằng quan hệ CONTAINS.
* Sản phẩm (Product): Apple thuộc loại Fruit, Salmon thuộc loại Fish.

Google Knowledge Graph đã thay đổi cách chúng ta tìm kiếm và tìm thông tin trên Web. Nó thu thập các dữ kiện về con người, địa điểm và sự vật thành một mạng lưới các thực thể có tổ chức. Khi bạn thực hiện tìm kiếm thông tin trên Google, nó sử dụng các kết nối giữa các thực thể để đưa ra các kết quả có liên quan nhất trong ngữ cảnh, ví dụ, trong hộp mà Google gọi là "Knowledge Panel.”



* + *Các ngôn ngữ biểu diễn đồ thị tri thức như RDF (Resource Description Framework) và OWL (Web Ontology Language).*

RDF (Resource Description Framework): Biểu diễn tri thức dưới dạng bộ ba gồm: Chủ ngữ (Subject), Vị ngữ (Predicate), và Tân ngữ (Object).

Ví dụ: <Hà Nội> <là thủ đô của> <Việt Nam>.

OWL (Web Ontology Language): Mở rộng RDF, cung cấp ngữ nghĩa phong phú hơn, hỗ trợ mô tả các mối quan hệ phức tạp và các ràng buộc logic.

Ví dụ: Định nghĩa "mọi người dân sống ở Hà Nội đều sống ở Việt Nam".

* **Phương pháp láng giềng gần nhất (KNN):**
  + *Tìm hiểu thuật toán KNN, cách hoạt động và cách áp dụng vào tìm kiếm ảnh dựa trên đặc trưng.*

a. Khái niệm cơ bản: KNN (K-Nearest Neighbors) là một thuật toán máy học không cần huấn luyện. Khi dự đoán nhãn của một điểm dữ liệu mới, KNN sẽ tìm K điểm dữ liệu gần nhất trong tập huấn luyện và sử dụng nhãn của chúng để đưa ra quyết định..

b. Cách hoạt động:

* Tính khoảng cách giữa điểm cần dự đoán và tất cả các điểm trong dữ liệu (thường dùng Euclidean, Manhattan, hoặc Cosine).
* Lấy ra K điểm gần nhất.
* Quyết định nhãn cho điểm cần dự đoán dựa trên nhãn chiếm đa số (nếu là phân loại) hoặc trung bình giá trị (nếu là hồi quy).

c. Ứng dụng vào tìm kiếm ảnh:

* Trong tìm kiếm ảnh, đặc trưng của ảnh (được trích xuất bằng SIFT, SURF, hoặc CNN) sẽ được sử dụng làm tọa độ trong không gian đa chiều.
* KNN sẽ tìm các ảnh có đặc trưng gần giống nhất với ảnh đầu vào.
* **Kỹ thuật trích xuất đặc trưng ảnh:**
  + *Sử dụng các phương pháp trích xuất đặc trưng như SIFT, SURF, hoặc CNN (Convolutional Neural Networks) để tạo đặc trưng ảnh.*

**a. SIFT (Scale-Invariant Feature Transform):**

* Phát hiện và mô tả các điểm đặc trưng quan trọng của ảnh.
* Bất biến với thay đổi về tỷ lệ, xoay, và ánh sáng.
* Ứng dụng trong nhận diện đối tượng, ghép ảnh.

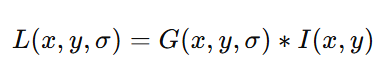
Bước 1: Phát hiện điểm đặc trưng

SIFT tìm các điểm đặc trưng trong ảnh bằng cách phát hiện các vùng có sự thay đổi mạnh về cường độ pixel ở nhiều tỉ lệ khác nhau.

1. Xây dựng không gian đa tỷ lệ (Scale-Space)

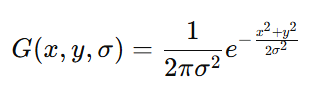
SIFT sử dụng biến đổi Gaussian để tạo ra các ảnh ở nhiều mức tỉ lệ khác nhau.

Ảnh gốc 𝐼(𝑥,𝑦) được làm mờ bằng bộ lọc Gaussian:



Trong đó:

* *G(x,y,σ)* là bộ lọc Gaussian:



* *σ* quyết định độ mờ (scale).

Tạo ra nhiều phiên bản của ảnh với giá trị *σ* khác nhau.

2. Tính toán DoG (Difference of Gaussians)

Tạo ra ảnh **DoG (Difference of Gaussians)** bằng cách lấy hiệu giữa hai ảnh Gaussian liên tiếp:



* Trong đó *k* là một hằng số tỉ lệ.
* DoG giúp tìm ra các điểm có sự thay đổi mạnh về độ sáng (cạnh và góc).

Bước 2: Lọc và chọn điểm đặc trưng ổn định

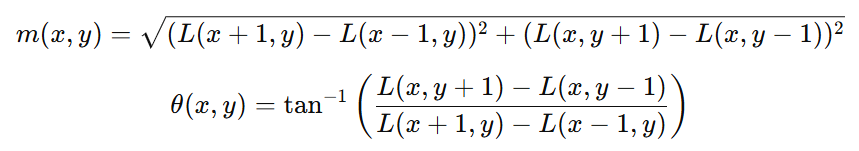
Các điểm đặc trưng là **cực đại hoặc cực tiểu** của DoG trên không gian 3D (tọa độ x,y,σ).

So sánh mỗi pixel với 26 điểm lân cận (8 điểm xung quanh + 9 điểm trên và 9 điểm dưới trong ảnh khác mức Gaussian).

Nếu điểm này là **lớn nhất hoặc nhỏ nhất**, nó được chọn làm điểm đặc trưng.

Bước 3: Xác định hướng của điểm đặc trưng

Với mỗi điểm đặc trưng (x,y,σ), tính toán gradient của ảnh:



Tạo histogram hướng gradient xung quanh điểm đặc trưng, gom nhóm hướng vào các bin 0° - 360°.

Chọn hướng có tần suất cao nhất làm hướng chính của điểm đặc trưng.

Bước 4: Tạo vector mô tả đặc trưng

Chia vùng xung quanh điểm đặc trưng thành lưới 4x4 ô.

Trong mỗi ô, tính hướng và cường độ gradient.

Gom nhóm các hướng vào 8 bin (0° - 360°).

Kết quả: Một vector có 128 phần tử (4x4x8), gọi là SIFT descriptor.

**Các bước thực hiện**

Bước 1: Trích xuất đặc trưng ảnh bằng SIFT

* Mỗi ảnh được xử lý bằng SIFT để phát hiện và mô tả điểm đặc trưng.
* Mỗi ảnh sẽ có một tập hợp **vector đặc trưng SIFT**.

Bước 2: Xây dựng đồ thị tri thức lưu trữ ảnh và đặc trưng

* Các ảnh và đặc trưng của chúng được biểu diễn trong **đồ thị tri thức**.
* Thực thể trong đồ thị tri thức:
  + Ảnh (Image) → Nút chính.
  + Đặc trưng SIFT (SIFT Features) → Các nút con của ảnh.
* Mối quan hệ:
  + (Ảnh) → có đặc trưng → (Vector SIFT).
  + (Ảnh) → thuộc loại → (Danh mục).

**Bước 3: Tìm kiếm ảnh tương tự bằng KNN**

* Khi có một ảnh truy vấn, ta trích xuất đặc trưng SIFT của nó.
* So sánh các vector đặc trưng của ảnh truy vấn với các ảnh trong cơ sở dữ liệu bằng **KNN**.
* Tìm **K ảnh gần nhất** dựa trên khoảng cách giữa các vector đặc trưng.

b. SURF (Speeded-Up Robust Features):

* Cải tiến từ SIFT, chạy nhanh hơn.
* Sử dụng phương pháp tích phân ảnh để tính toán nhanh hơn.
* Thích hợp với ứng dụng thời gian thực.

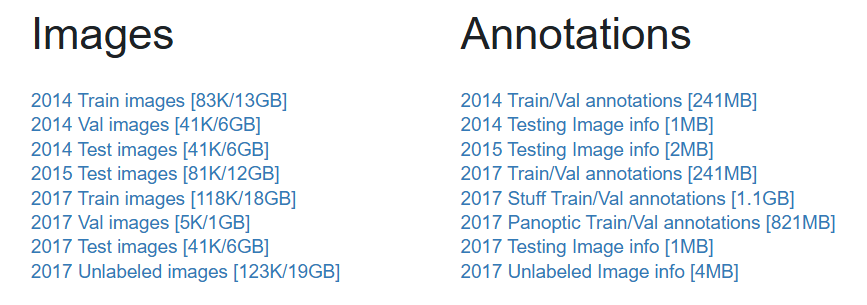
c. CNN (Convolutional Neural Networks):

* CNN là một mạng nơ-ron sâu có thể tự động học đặc trưng từ ảnh.
* Các lớp quan trọng trong CNN:
  + Lớp tích chập (Convolutional Layer): Trích xuất đặc trưng cục bộ.
  + Lớp pooling: Giảm chiều dữ liệu, giữ đặc trưng quan trọng.
  + Lớp Fully Connected: Biến đổi thành vector đặc trưng.
* Ứng dụng CNN trong tìm kiếm ảnh:
  + Mỗi ảnh được đưa qua một CNN để trích xuất vector đặc trưng.
  + So sánh vector của ảnh cần tìm với vector trong cơ sở dữ liệu bằng KNN để tìm ảnh tương tự.

**2. Thu thập dữ liệu**

* Xây dựng bộ dữ liệu ảnh:
  + Thu thập ảnh từ các nguồn như ImageNet, COCO, hoặc tạo tập dữ liệu riêng.

COCO chứa 118.000 ảnh với 90 loại đối tượng khác nhau được gán nhãn chi tiết theo dạng bounding box, segmentation, keypoints, giúp xây dựng đồ thị tri thức chính xác hơn.



* + Gắn nhãn các đối tượng trong ảnh (chú thích bằng bounding boxes, tên đối tượng, và quan hệ).
* Chuẩn hóa dữ liệu:
  + Tạo cấu trúc đồ thị tri thức từ bộ dữ liệu bằng cách ánh xạ các đối tượng, mối quan hệ và thuộc tính.

**3. Xây dựng đồ thị tri thức**

* Thiết kế mô hình đồ thị tri thức:
  + Thực thể: Các đối tượng (người, vật, địa điểm, v.v.).
  + Quan hệ: Các mối quan hệ giữa các đối tượng (ở gần, nằm trong, tương tác với, v.v.).
  + Thuộc tính: Mô tả chi tiết các đối tượng (màu sắc, hình dạng, kích thước).
* Triển khai đồ thị tri thức bằng công cụ:
  + Sử dụng **Protégé** hoặc **Neo4j** để tạo và quản lý đồ thị tri thức.