

Dưới đây là bảng liệt kê tất cả các đặc trưng (features) chính của **Topological Data Analysis (TDA)**, bao gồm cả những đặc trưng đã đề cập trước đó và các đặc trưng khác, cùng với cột **Ứng dụng dạng dữ liệu** (đồ thị, văn bản, point cloud, v.v.) và cột **Thư viện** hỗ trợ. Tôi sẽ trình bày ngắn gọn, rõ ràng, và đầy đủ.

Đặc trưng	Mô tả	Ứng dụng dạng dữ liệu	Thư viện
<b>Persistent Homology (H0)</b>	Đếm số thành phần liên thông (connected components) trong point cloud, biểu thị các nhóm dữ liệu.	Point cloud (văn bản, hình ảnh, mạng xã hội), đồ thị, dữ liệu sinh học.	Ripser, GUDHI, Persim, Dionysus
<b>Persistent Homology (H1)</b>	Đếm số vòng lặp (loops) 1 chiều, biểu thị các chu kỳ trong dữ liệu.	Point cloud (văn bản, hình ảnh), mạng lưới, dữ liệu thời gian.	Ripser, GUDHI, Persim, Dionysus
<b>Persistent Homology (H2, H3, ...)</b>	Đếm các lỗ cao chiều (khoang rỗng, H2; lỗ 3D, H3), biểu thị cấu trúc phức tạp.	Point cloud (hình ảnh 3D, dữ liệu sinh học), dữ liệu không gian cao chiều.	Ripser, GUDHI, Dionysus
<b>Betti Numbers</b>	Số lượng đặc trưng topological ở mỗi chiều (Betti-0 = H0, Betti-1 = H1, Betti-2 = H2, ...).	Point cloud, đồ thị, văn bản, dữ liệu hình học.	GUDHI, Ripser (trích xuất từ persistence diagram)
<b>Number of Edges</b>	Số cạnh trong biểu diễn đồ thị của dữ liệu (thường từ Mapper hoặc Rips complex).	Đồ thị, point cloud (mạng xã hội, văn bản), mạng lưới.	GUDHI, Kepler-Mapper, NetworkX (kết hợp)
<b>Number of Strongly Connected Components</b>	Số thành phần liên thông mạnh trong đồ thị có hướng.	Đồ thị có hướng (mạng xã hội, web), dữ liệu quan hệ.	NetworkX, GUDHI (phân tích đồ thị)
<b>Persistence Landscapes</b>	Biến đổi persistence diagram thành hàm liên tục để so sánh dữ liệu.	Point cloud, văn bản, hình ảnh, dữ liệu thời gian.	GUDHI, Persim

<b>Persistence Images</b>	Chuyển persistence diagram thành hình ảnh 2D để dùng trong học máy.	Point cloud, văn bản, hình ảnh, dữ liệu sinh học.	Persim, GUDHI
<b>Mapper Algorithm</b>	Tạo biểu đồ đơn giản hóa dữ liệu với các nút (cụm) và cạnh (chồng lấn).	Point cloud, văn bản, hình ảnh, dữ liệu cao chiều.	Kepler-Mapper, GUDHI
<b>Bottleneck Distance</b>	Đo khoảng cách giữa hai persistence diagrams để so sánh dữ liệu.	Point cloud, văn bản, hình ảnh, dữ liệu sinh học.	Persim, GUDHI
<b>Wasserstein Distance</b>	Đo khoảng cách tổng quát hơn giữa persistence diagrams.	Point cloud, văn bản, hình ảnh, dữ liệu sinh học.	Persim, GUDHI
<b>Euler Characteristic</b>	Tóm tắt cấu trúc topological bằng tổng xen kẽ số đỉnh, cạnh, mặt, v.v.	Đồ thị, point cloud, hình ảnh, dữ liệu không gian.	GUDHI, Ripser (gián tiếp)
<b>Topological Entropy</b>	Đo độ phức tạp topological của dữ liệu, thường dùng trong hệ động lực.	Dữ liệu thời gian, point cloud, dữ liệu sinh học.	GUDHI (hỗ trợ tùy chỉnh)
<b>Homology Classes</b>	Biểu diễn cụ thể các đặc trưng topological (như vòng lặp cụ thể).	Point cloud, đồ thị, dữ liệu hình học.	GUDHI, Ripser (phân tích chi tiết)

### Giải thích ngắn gọn:

- **Ứng dụng dạng dữ liệu:**
  - **Point cloud:** Thường gặp trong văn bản (word embeddings), hình ảnh (pixel), hoặc dữ liệu sinh học (tọa độ phân tử).
  - **Đồ thị:** Mạng xã hội, mạng lưới giao thông, quan hệ từ trong văn bản.
  - **Văn bản:** Word embeddings (như trong code của bạn) hoặc biểu diễn đồ thị từ.
  - **Hình ảnh:** Pixel hoặc đặc trưng trích xuất từ CNN.
  - **Dữ liệu sinh học:** Cấu trúc protein, tế bào, hoặc mạng lưới gen.
- **Thư viện:**
  - **Ripser:** Tính persistent homology ( $H_0, H_1, H_2, \dots$ ), hiệu quả cho point cloud.
  - **GUDHI:** Hỗ trợ đa dạng đặc trưng (homology, Mapper, landscapes, ...).
  - **Persim:** Phân tích persistence diagrams (bottleneck, Wasserstein, images).
  - **Kepler-Mapper:** Tạo biểu đồ Mapper.
  - **NetworkX:** Phân tích đồ thị (edges, strongly connected components).
  - **Dionysus:** Tính homology, ít phổ biến hơn Ripser/GUDHI.

## Liên hệ với code của bạn:

- Code hiện tại chỉ dùng  **$H_0, H_1$**  (Ripser) trên point cloud từ văn bản (word embeddings).
- Bạn có thể mở rộng:
  - Tính  **$H_2$**  bằng cách sửa `maxdim=2` trong `ripser`.
  - Dùng **Persistence Images** với Persim để đưa vào mô hình học máy.
  - Áp dụng **Mapper** (Kepler-Mapper) để vẽ đồ thị các nhóm từ.
  - Tính **Number of Edges** hoặc **Strongly Connected Components** bằng cách xây dựng đồ thị từ point cloud (kết hợp NetworkX).

Nếu muốn thử một đặc trưng cụ thể (như Mapper hoặc Persistence Images) trong code của bạn, tôi có thể cung cấp code mẫu!