

# 1. Phương pháp Pointwise

## 1.1. Mô tả

Phương pháp Pointwise xem bài toán xếp hạng như một bài toán dự đoán (regression hoặc classification) trên từng tài liệu độc lập.

Mỗi tài liệu được gán một nhãn hoặc một điểm số liên quan (relevance score) dựa trên mức độ phù hợp với truy vấn.

Mục tiêu là dự đoán chính xác nhãn hoặc điểm số này cho mỗi tài liệu.

## 1.2. Quy trình hoạt động

**Đầu vào:** Dữ liệu dưới dạng các tài liệu độc lập cùng nhãn (label) như 0, 1, hoặc 2 (không phù hợp, trung bình, phù hợp).

Ví dụ:

- Truy vấn: "mua laptop giá rẻ".
- Tài liệu 1: Laptop X giá rẻ nhất → Nhãn: 2.
- Tài liệu 2: Laptop Y không liên quan → Nhãn: 0.

**Hàm mục tiêu:** Nếu bài toán là classification, dùng hàm Cross-Entropy Loss.

Nếu bài toán là regression, dùng Mean Squared Error (MSE).

**Dự đoán:** Dựa trên mô hình học máy (linear regression, neural network,...) để gán điểm số cho từng tài liệu.

**Xếp hạng:** Sắp xếp tài liệu dựa trên điểm số dự đoán.

## 1.3. Ưu điểm và nhược điểm

### 1.3.1. Ưu điểm:

- **Đơn giản:** Chỉ xử lý từng tài liệu một cách độc lập.
- **Dễ triển khai:** Có thể áp dụng trực tiếp các mô hình học máy thông thường.
- **Ứng dụng rộng:** Phù hợp với dữ liệu không có mối quan hệ giữa các tài liệu.

### 1.3.2. Hạn chế:

**Không tối ưu cho thứ hạng:** Phương pháp này không xem xét mối quan hệ giữa các tài liệu trong cùng danh sách, dẫn đến kết quả xếp hạng không chính xác trong một số trường hợp.

## 2. Phương pháp Pairwise

### 2.1. Mô tả

Phương pháp Pairwise tập trung vào so sánh cặp tài liệu để xác định thứ tự tương đối giữa chúng.

Thay vì dự đoán trực tiếp điểm số cho từng tài liệu, phương pháp này đánh giá xem tài liệu nào nên xếp cao hơn trong một cặp.

### 2.2. Quy trình hoạt động

**Đầu vào:** Tạo ra các cặp tài liệu ( $document\_i$ ,  $document\_j$ ) với nhãn:

- 1 nếu  $document\_i$  có thứ hạng cao hơn  $document\_j$ .
- 0 nếu ngược lại.

Ví dụ:

- Truy vấn: "đặt vé máy bay giá rẻ".
- Tài liệu 1: Vé rẻ nhất  $\rightarrow$  Thứ hạng 1.
- Tài liệu 2: Vé có giá trung bình  $\rightarrow$  Thứ hạng 2.
- Nhãn cặp: (Tài liệu 1, Tài liệu 2)  $\rightarrow$  1.

**Hàm mục tiêu:** Dùng các hàm mất mát như:

- **Hinge Loss (SVM):** Thích hợp cho bài toán margin-based ranking.
- **Log Loss:** Phù hợp cho bài toán classification giữa các cặp.

**Dự đoán:** Mô hình dự đoán xác suất tài liệu nào nên xếp cao hơn trong cặp.

**Xếp hạng:** Từ các dự đoán trên, tạo danh sách thứ tự xếp hạng bằng cách tổng hợp thông tin từ các cặp.

### 2.3. Ưu điểm và nhược điểm

#### 2.3.1. Ưu điểm:

- **Tập trung vào quan hệ:** Tối ưu hóa thứ tự giữa các tài liệu, phù hợp với các ứng dụng cần so sánh trực tiếp (ví dụ: xếp hạng sản phẩm).
- **Đơn giản hơn Listwise:** Không cần tối ưu toàn bộ danh sách.

#### 2.3.2. Hạn chế:

- **Tốn tài nguyên:** Số lượng cập tăng theo cấp số nhân, gây khó khăn khi dữ liệu lớn.
- **Không toàn cục:** Không tận dụng thông tin toàn bộ danh sách.

### 3. Phương pháp Listwise

#### 3.1. Mô tả

Phương pháp Listwise coi bài toán xếp hạng là bài toán tối ưu hóa trên toàn bộ danh sách tài liệu.

Phương pháp này tập trung vào việc tối ưu hóa trực tiếp các chỉ số đánh giá xếp hạng (ranking metrics) như NDCG, MAP, hoặc MRR.

#### 3.2. Quy trình hoạt động

**Đầu vào:** Toàn bộ danh sách tài liệu liên quan đến một truy vấn, cùng với điểm số thực tế hoặc nhãn thứ hạng.

Ví dụ:

- Truy vấn: "học lập trình Python".
- Danh sách tài liệu:
  - Tài liệu 1: Học Python cơ bản → Điểm: 3.
  - Tài liệu 2: Python nâng cao → Điểm: 2.
  - Tài liệu 3: Tài liệu không liên quan → Điểm: 0.

**Hàm mục tiêu:**

- **Softmax Cross-Entropy Loss:** Tính xác suất phân phối thứ hạng trong danh sách.
- **LambdaRank / LambdaMART:** Tối ưu trực tiếp NDCG bằng cách sử dụng gradient-based approach.

**Dự đoán:** Mô hình dự đoán thứ tự xếp hạng tối ưu của toàn danh sách dựa trên các chỉ số như NDCG.

**Xếp hạng:** Sắp xếp tài liệu theo thứ tự dự đoán để đạt hiệu quả xếp hạng cao nhất.

#### 3.3. Ưu điểm và nhược điểm

##### 3.3.1. Ưu điểm:

- **Tối ưu toàn diện:** Phù hợp để cải thiện các chỉ số xếp hạng toàn cục.
- **Ứng dụng thực tế:** Thích hợp với hệ thống tìm kiếm và đề xuất hiện đại.

3.3.2. Hạn chế:

- **Phức tạp:** Đòi hỏi thiết kế mô hình và thuật toán chuyên biệt.
- **Cần nhiều dữ liệu:** Yêu cầu dữ liệu huấn luyện đầy đủ với thông tin về thứ tự danh sách.

BẢNG SO SÁNH:

Phương pháp	Đối tượng xử lý	Hàm mục tiêu	Mức độ phức tạp	Ứng dụng chính
Pointwise	Từng tài liệu độc lập	MSE, Cross-Entropy	Thấp	Dự đoán relevance score
Pairwise	Cặp tài liệu	Hinge Loss, Log Loss	Trung bình	So sánh tài liệu, hệ thống đề xuất
Listwise	Toàn bộ danh sách tài liệu	Softmax Loss, LambdaRank	Cao	Hệ thống tìm kiếm, tối ưu NDCG