

# Learning2Rank

排序學習

# Learning2Rank



# What is **Ranking**

---

# 排序問題

Google

吃到飽



全部

地圖

新聞

圖片

影片

更多

設定

工具



約有 5,690,000 項結果 (搜尋時間：0.59 秒)

499吃到飽竟有兩大陷阱神人算給你看- 中時電子報

[www.chinatimes.com/realtimenews/20180512001412-260405](http://www.chinatimes.com/realtimenews/20180512001412-260405) ▼

2 天前 - 中華電信加入499不限速**吃到飽**混戰，掀起全民申辦喪屍潮，電信門市大排長龍，一圈又一圈，有大媽寧願付解約金3萬元仍要申辦。有網友在爆廢 ...

499吃到飽讓中華電信賠錢？他神解「賺翻」內幕- 中時電子報

[www.chinatimes.com/realtimenews/20180514000008-260405](http://www.chinatimes.com/realtimenews/20180514000008-260405) ▼

16 小時前 - 三大電信業者祭出499元**吃到飽**活動，引發全台民眾瘋搶換約。對於此499之亂現象，有網友對中華電信這一波熱潮發表神見解，認為不但獲利沒影響 ...

中華電信499 吃到飽老客戶被犧牲？你其實還有：中華電信549 ... - 蘋果仁

<https://applealmond.com/posts/31955> ▼

17 小時前 - 中華電信的「499 **吃到飽**之亂」即將在5月15日劃下句點，越來越多人怕申請不到，母親節整個塞爆了門市，甚至還有人搬出帳篷來準備長期抗戰。

【台北10大BUFFET推薦】C/P值超高！台北吃到飽特搜- FunTime旅遊推薦

<https://www.funtime.com.tw/blog/funtime/台北10家cp值超高吃到飽buffet特搜> ▼

LIU| FunTime. 很多時候想吃各式各樣的美食，但又沒有太多預算嗎？那就去吃Buffet吧！擁有各種不同的品項、又可以**吃到飽**，滿足你的各種需求。小編帶著大家走遍 ...

吃到飽

[asp.iwpower.com/cpo/good01.html](http://asp.iwpower.com/cpo/good01.html) ▼

提供台北**吃到飽**餐廳資訊,詳列2018推薦店家的訂位電話及地址,包括日本料理、港式飲茶、中式料理、燒烤肉、火鍋、異國、歐式Buffet、蒙古烤肉、下午茶、海鮮、素食、 ...

八田頂級帝王蟹鍋物燒烤吃到飽（市民店） - Bar & Grill - Taipei, Taiwan ...

Query 1: 吃到飽

no. 4, 5, 6 hit!



# 排序問題



米其林



全部

圖片

新聞

地圖

影片

更多

設定

工具

約有 4,990,000 項結果 (搜尋時間：0.78 秒)

## 台灣米其林輪胎

[www.michelin.com.tw/TW/zh/homepage.html](http://www.michelin.com.tw/TW/zh/homepage.html) ▼

台灣**米其林**輪胎綜合各類汽車輪胎性能的平衡表現，包含輪胎省油、操控、抓地力、寧靜舒適與耐磨高里程等優異性能，**米其林**重視汽車輪胎性能，更守護您的行車 ...

[瀏覽所有輪胎](#) · [聯絡我們](#) · [我是否需要新輪胎？](#) · [選擇適合胎款](#)

## 米其林指南Taipei – 官方網站 - Michelin Guide

<https://guide.michelin.com/tw/taipei> ▼

**米其林**星級餐館，必比登推介以及所有**米其林**指南的各種選項Taipei。精緻餐飲fine dining、美食文章等內容的最佳提供者。

[My 灶](#) · [查看全部文章](#) · [Taipei米其林餐館– 米其林指南](#) · [建宏牛肉麵](#)

## Taipei米其林餐館– 米其林指南 - Michelin Guide

<https://guide.michelin.com/tw/taipei/restaurants> ▼

查詢全部**米其林**指南的餐館評級Taipei。餐館食評和各種有用信息與資料可在網上查閱。

## 台北米其林完整名單：這20家餐廳，為台灣摘下24顆星星 | 時尚生活 ...

<https://www.cw.com.tw/article/article.action?id=5088691> ▼

2018年3月14日 - 被視為美食聖經的《**米其林**指南》，今天（3月14日）發表首屆台北版名單，共有110家餐廳入榜，除了上週（6日）先行公布的36家必比登推介（Bib ...

## 米其林指南- 维基百科，自由的百科全书

<https://zh.wikipedia.org/zh-tw/米其林指南> ▼

《米芝蓮指南》（法语：Le Guide **Michelin**）是法国知名輪胎製造商米芝蓮公司所出版的美食及旅遊指南書籍的總稱，其中以評鑑餐廳及旅館、封面為红色的「紅色指南」（ ...



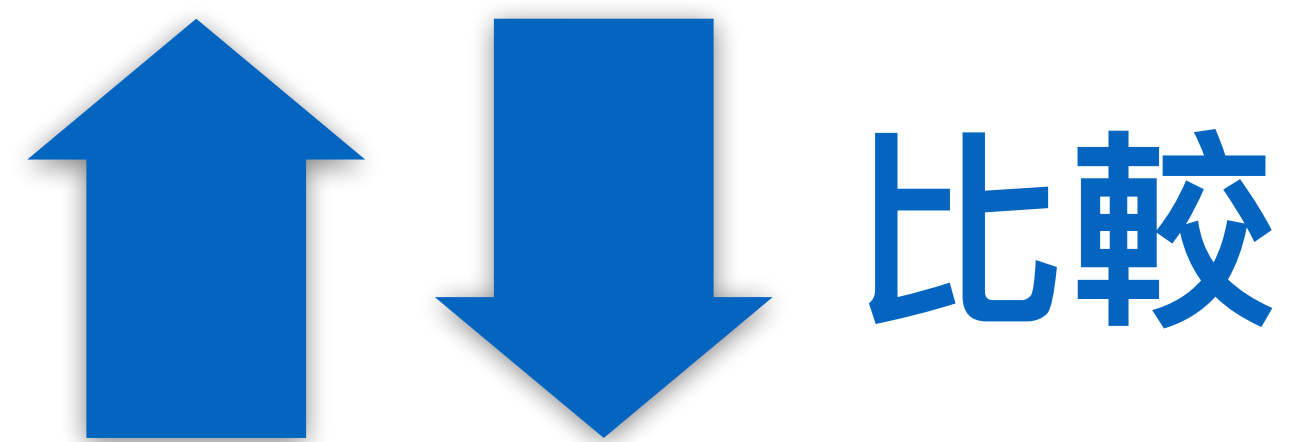
Query 2: 米其林

no. 2~9 hit!

**排序效果好不好？**

# 評價指標

**Ranking List: model**



**Ranking List: ground truth**

# 評價指標

## MAP (Mean Average Precision)

Query 1		
Rank	Relevance	P(k)
1	✓	1.00
2		
3	✓	0.67
4		
5		
6	✓	0.50
7		
8		
9		
10	✓	0.40
Average Precision:		0.6425

Query 2		
Rank	Relevance	P(k)
1	✓	1.00
2		
3	✓	0.67
4		
5		
Average Precision:		0.835

→ Precision at top-k

$$MAP = \frac{0.6425 + 0.835}{2} = 0.73875$$





# 評價指標

**NDCG** (Normalized Discounted **Cumulative** Gain)

**CG: 累計增益** ➡ **Discounted** ➡ **Normalized**

**高相關性的文檔比較有價值；**  
**同樣相關等級的文檔，排名在較**  
**前面的會比較有價值**

**缺點:** 等級相關性與排序位置無關

**缺點:** 不同 query 回傳的搜索結果數量不同，其 DCG 值差異大，無法比較

**優點:** 可跨 query 比較

**DCG** 考慮排序位置的影響，排序在越前面的文檔，其相關性表現對整體排序效果的影響越大

**Ideal-DCG** 為搜索結果按相關性排序之後，能得到的最大 DCG 值

$CG_p = \sum_{i=1}^p rel_i$

$DCG_p = \sum_{i=1}^p \frac{2^{rel_i} - 1}{\log_2(i+1)}$

$NDCG_p = \frac{DCG_p}{IDCG_p}$

**p** 為文檔在搜索結果列表中的排序位置，**rel** 為處在該位置文檔的等級相關性 (**graded relevance**)

# 評價指標

## NDCG

Query 1		
Doc ID	Relevance	分數 Gain
1	高度相關	3
2	中度相關	2
3	高度相關	3
4	不相關	0
5	輕度相關	1
6	中度相關	2

CG

累計增益

$$3 + 2 + 3 + 0 + 1 + 2 = 11$$

DCG

Discounted

gain function

$$\left(\frac{2^3 - 1}{\log_2^{1+1}}\right) + \left(\frac{2^2 - 1}{\log_2^{2+1}}\right) + \left(\frac{2^3 - 1}{\log_2^{3+1}}\right) + \left(\frac{2^0 - 1}{\log_2^{4+1}}\right) + \left(\frac{2^1 - 1}{\log_2^{5+1}}\right) + \left(\frac{2^2 - 1}{\log_2^{6+1}}\right)$$

discount function

NDCG

Normalized

按相關性排序可得到最優結果，即 **IDCG** 為文檔按照 **{3, 3, 2, 2, 1, 0}** 相關性排序

$$NDCG_p = \frac{DCG_p}{IDCG_p}$$

# 評價指標

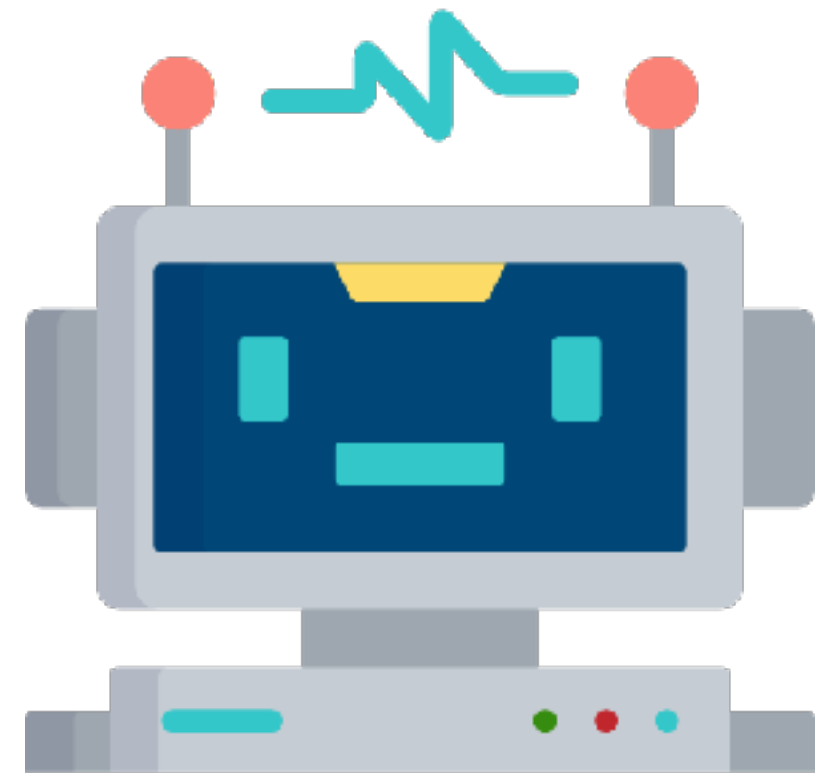
**MAP:** a **binary** degree of relevance assessment

判斷文檔的相關性程度，僅有**是否相關(1/0)**兩種值

**MAP & NDCG:** **最大值為 1**

值越接近 **1**，表示排序效果越好

# Learning2Rank



# How to Learn

---

**What is LTR**



# Machine Learning

## Ranking

### L2R

監督式學習

# Information Retrieval

# 資訊檢索：傳統排序

相關度排序模型 (Relevance Ranking Model)  
傳統排序模型，單一模型只考慮

單一面向 (相關度 or 重要性)  
重要性排序模型 (Importance Ranking Model)

常用的模型包括: PageRank, HITS...

# 排序學習

Group 1: 吃到飽							
Doc ID	Cosine Similarity	PageRank	URL Length	Domain Age	...	Rank	Label
4	0.34	0.42	8	1	...	1	相關
5	0.83	0.54	6	5	...	2	相關
3	0.59	0.22	10	1	...	3	相關
1	0.38	0.15	12	0.5	...	4	不相關
2	0.12	0.78	6	3	...	5	不相關
6	0.11	0.33	20	0.7	...	6	不相關

將各個傳統排序模型的 **【輸出作為特徵】**，訓練出新的模型與參數，  
生成新的排序學習模型，使其輸出的文檔序列和真實序列儘可能相似

# **Approach of LTR**

---

# 排序學習

**1. Point-wise**

**2. Pair-wise**

**3. List-wise**

A light beige document icon with a folded top-right corner and horizontal lines representing text.

**Doc 1**

A light beige document icon with a folded top-right corner and horizontal lines representing text.

**Doc 2**

A light beige document icon with a folded top-right corner and horizontal lines representing text.

**Doc 3**

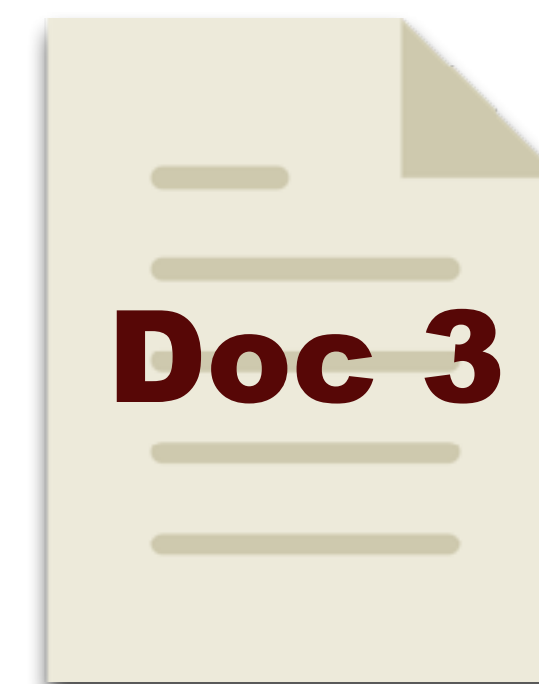
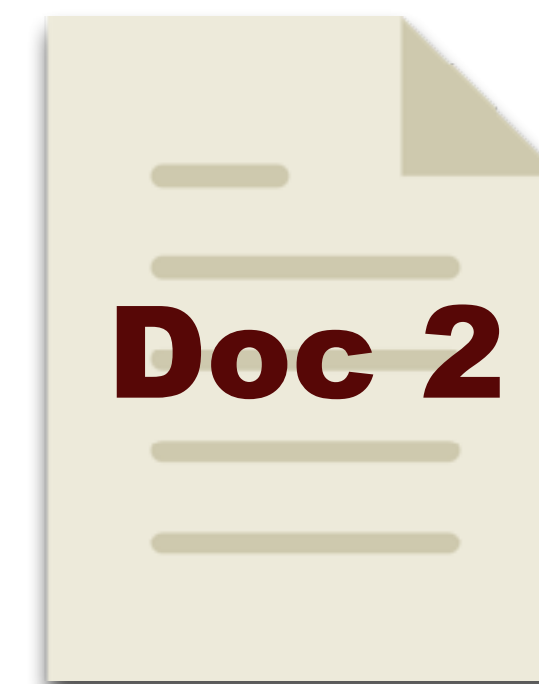
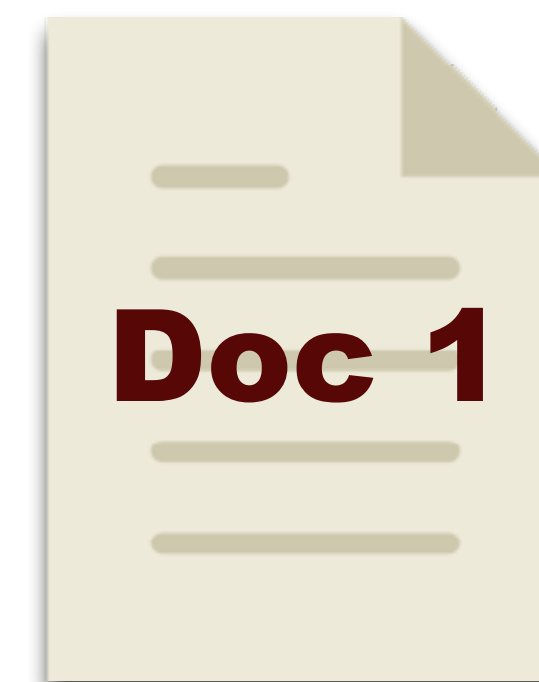


# Point-wise

Query 1:  
吃到飽



0.2分



# Point-wise

**Query 1:**  
**吃到飽**

高度相關

**Doc 2**

中度相關

**Doc 3**

輕度相關

**Doc 1**

# Point-wise

## 1. 僅使用傳統的分類、迴歸方法

- 僅考慮單個文檔和 Query 的絕對相關度
- 不考慮其他文檔和給定查詢的相關度

## 2. 未考慮排序的重要特徵

- 應針對給定查詢下的「文檔集合」做文檔之間的排序
- 在排序問題中，排在最前面的幾個文檔對排序效果的影響非常重要

# 排序學習

**1. Point-wise**

**2. Pair-wise**

**3. List-wise**

A light beige document icon with a folded top-right corner and horizontal lines representing text. The text "Doc 1" is centered in a dark red font.

**Doc 1**

A light beige document icon with a folded top-right corner and horizontal lines representing text. The text "Doc 2" is centered in a dark red font.

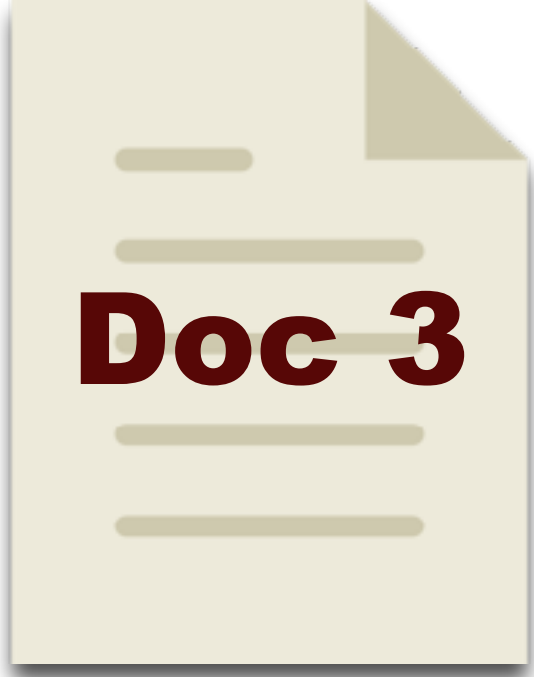
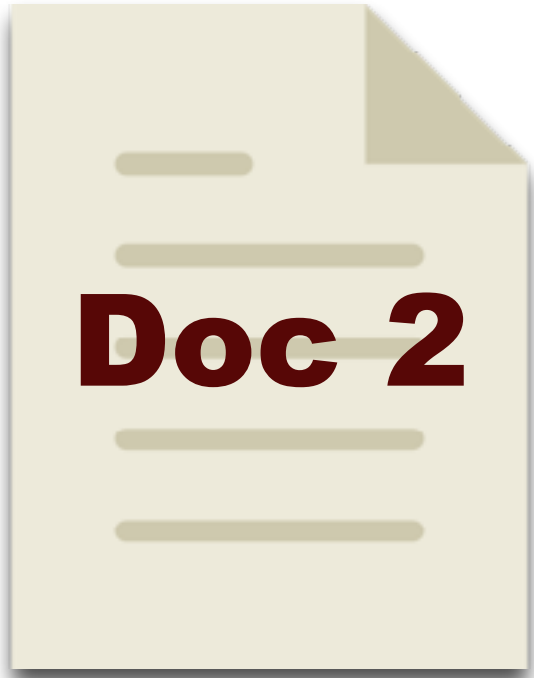
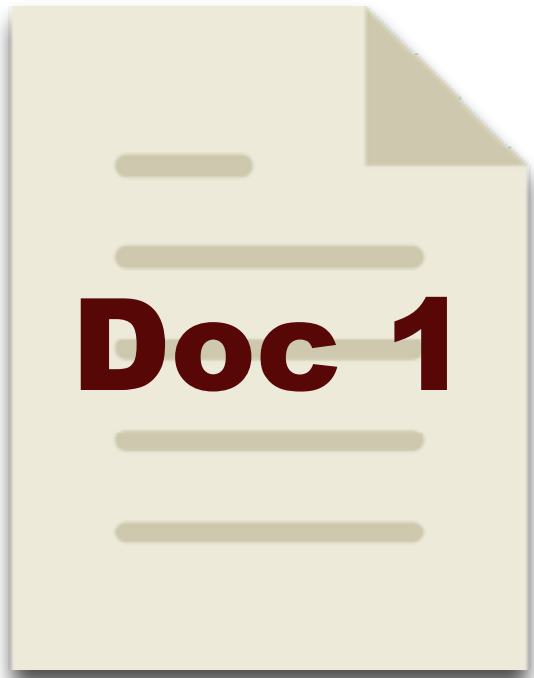
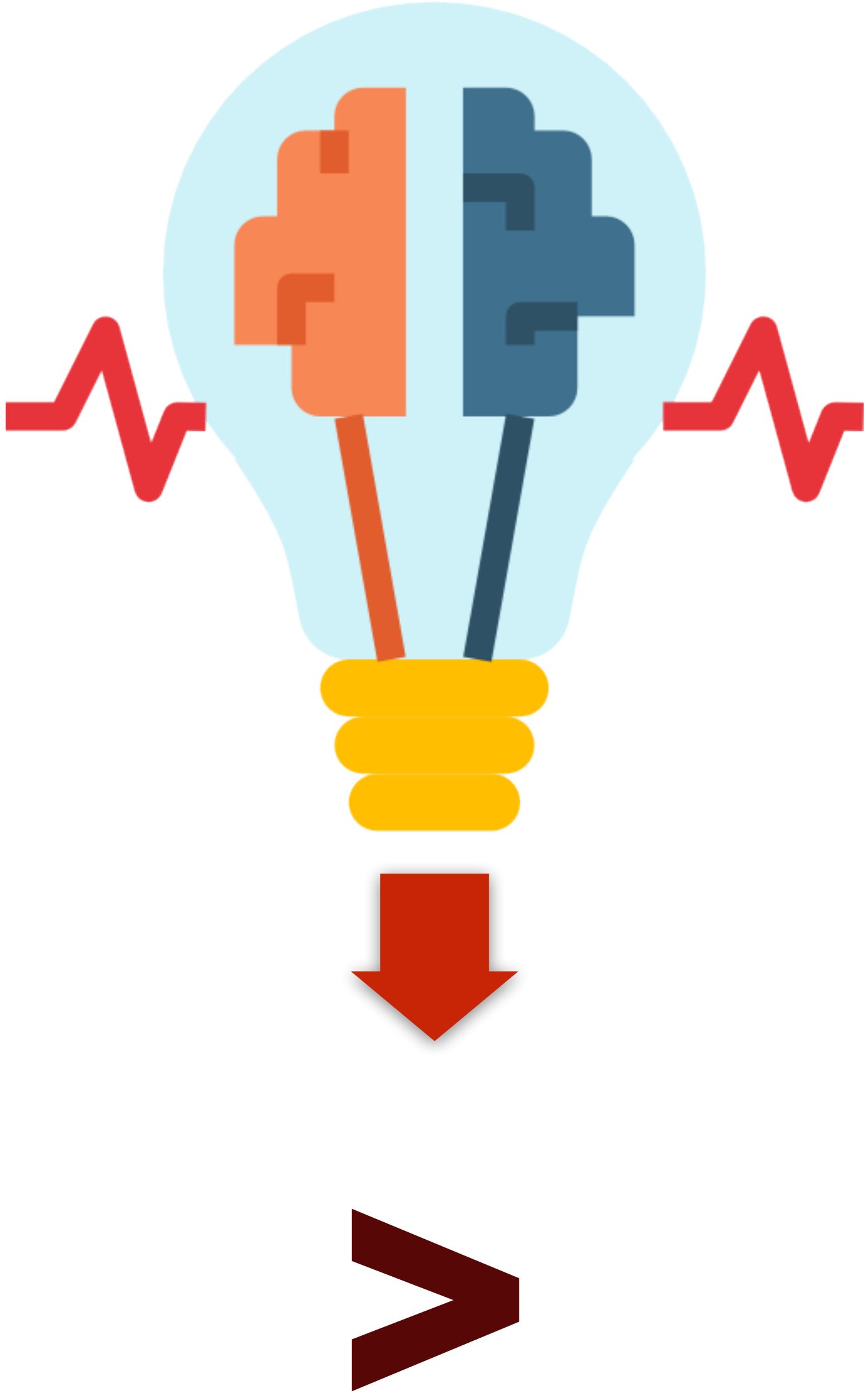
**Doc 2**

A light beige document icon with a folded top-right corner and horizontal lines representing text. The text "Doc 3" is centered in a dark red font.

**Doc 3**

# Pair-wise

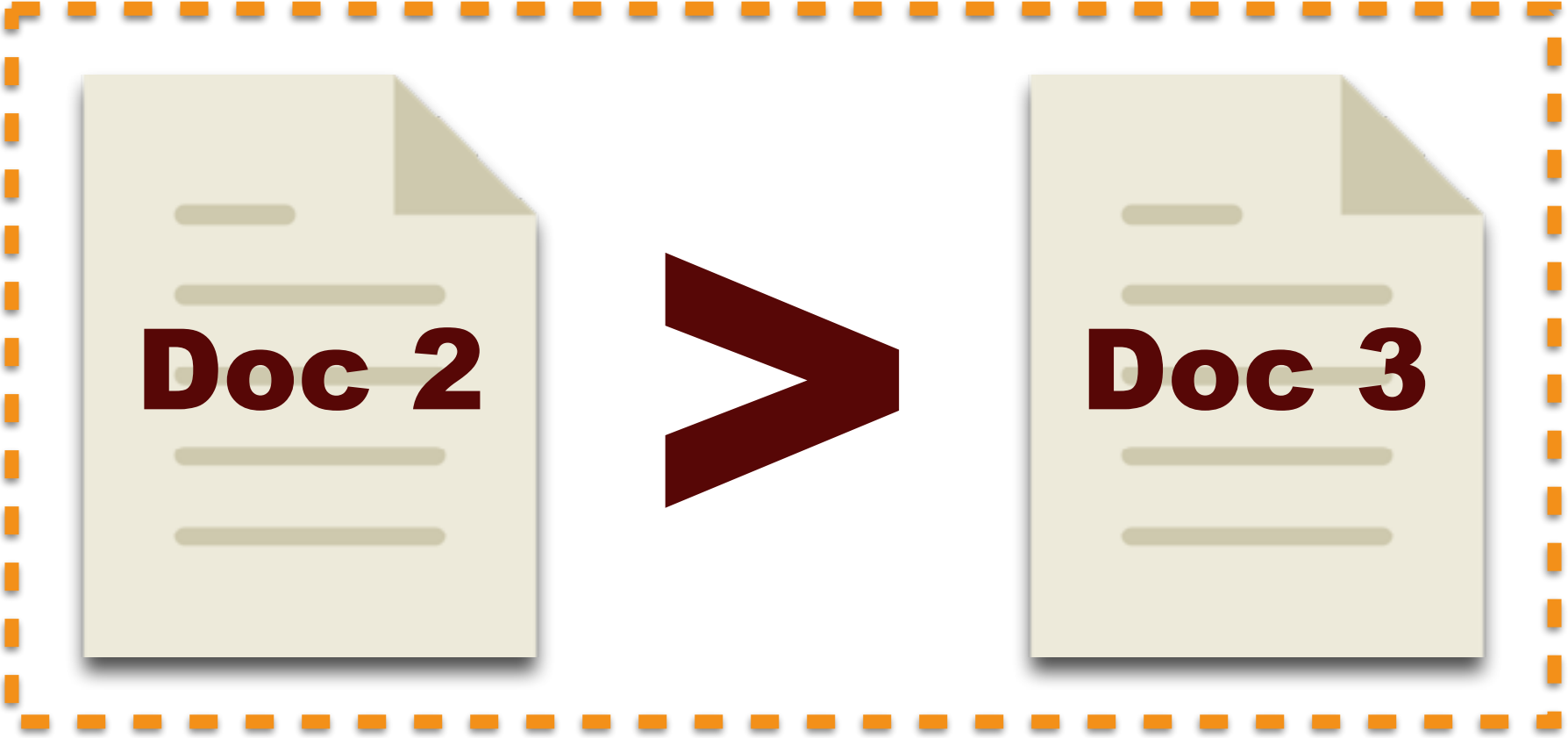
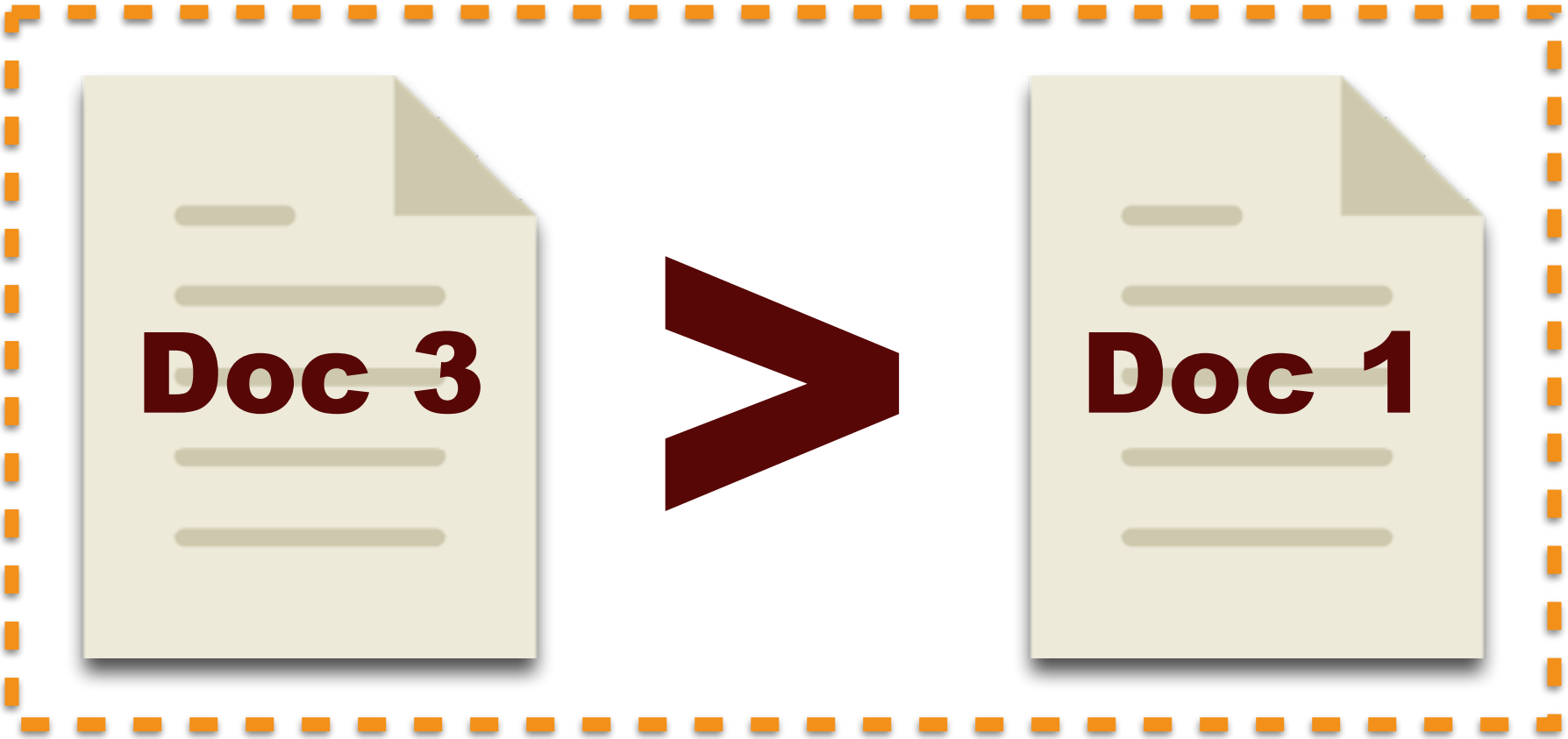
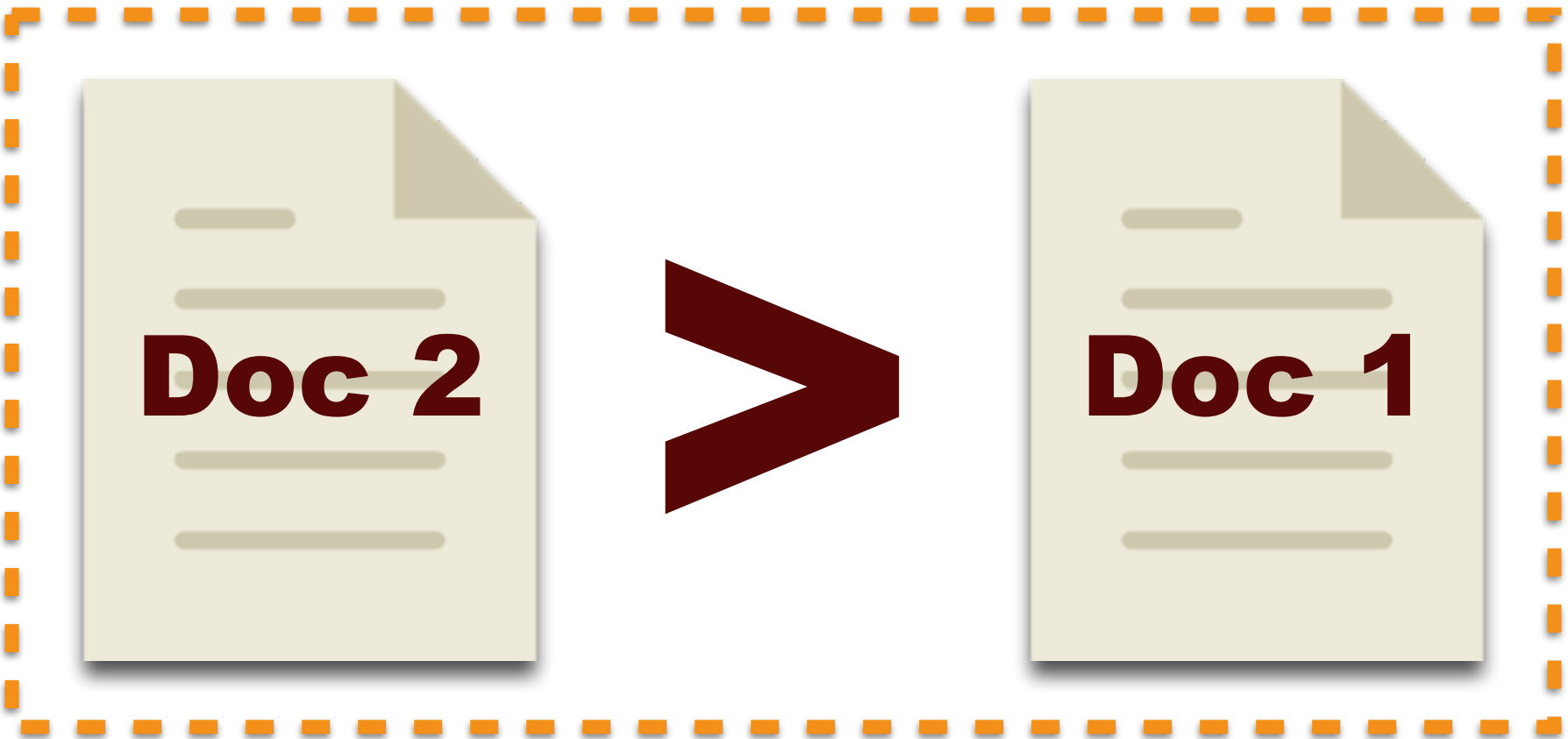
Query 1:  
吃到飽





# Pair-wise

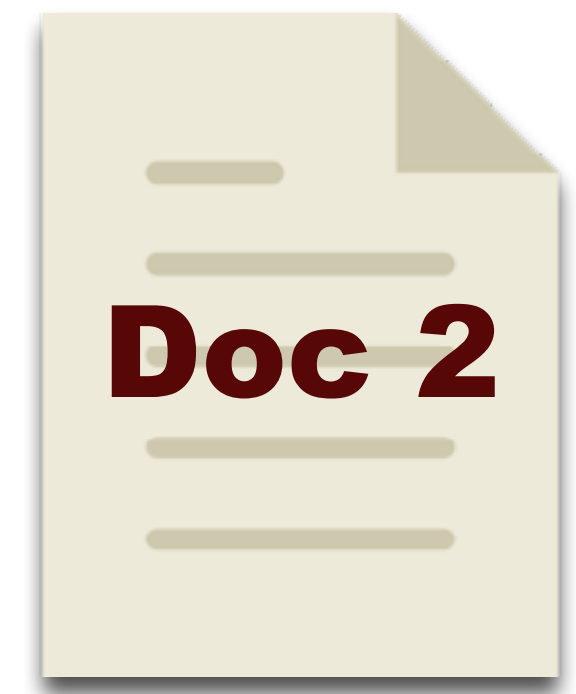
Query 1:  
吃到飽



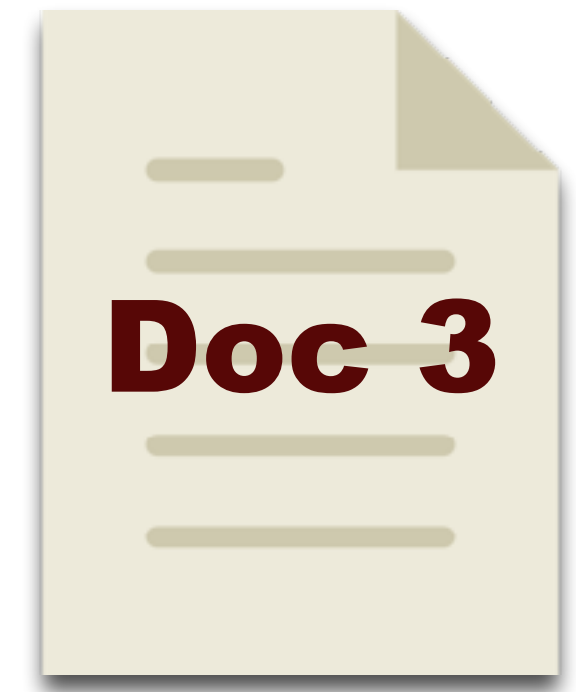
# Pair-wise

**Query 1:**  
**吃到飽**

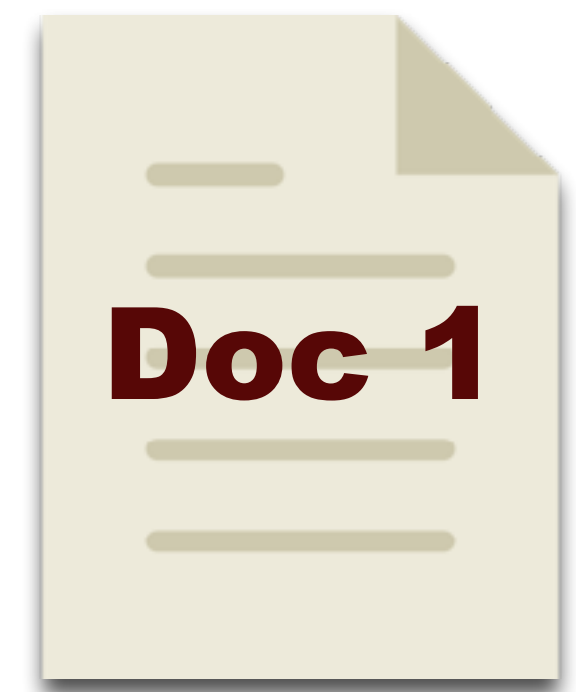
高度相關



中度相關



輕度相關



# Pair-wise

## 1. 判斷任意文檔對是否滿足順序關係

- 給定查詢下，考慮任意兩個相關度不同的文檔之間的相對相關度

## 2. 針對文檔對，進行二元分類學習

- $\text{Doc}_i > \text{Doc}_j$    $\text{Doc}_i - \text{Doc}_j > 0$ , **+1 (Positive instance)**

# Pair-wise

## 3. 未全盤考慮排序的重要特徵

- 基於兩兩文檔之間相對相關度的**損失函數**，和真正衡量排序效果的指標，有所落差
- 只考慮兩個文檔的相對先後順序，卻**沒有考慮文檔出現在排序列表中的位置**

# Pair-wise

**Grade: 3, 2, 1**

**Documents are represented by their grades**

**Ex 1:**

**ranking for query-1:** 3 2 1

**Ex 2:**

**ranking-1 for query-2:** 2 3 2 1 1 1 1

**ranking-2 for query-2:** 3 2 1 2 1 1 1

**Ex 3:**

**ranking for query-3:** 3 2 2 1 1 1 1

**ranking for query-4:** 3 3 2 2 2 1 1 1 1 1



# 排序學習

**1. Point-wise**

**2. Pair-wise**

**3. List-wise**

A light beige document icon with a folded top-right corner and horizontal lines representing text.

**Doc 1**

A light beige document icon with a folded top-right corner and horizontal lines representing text.

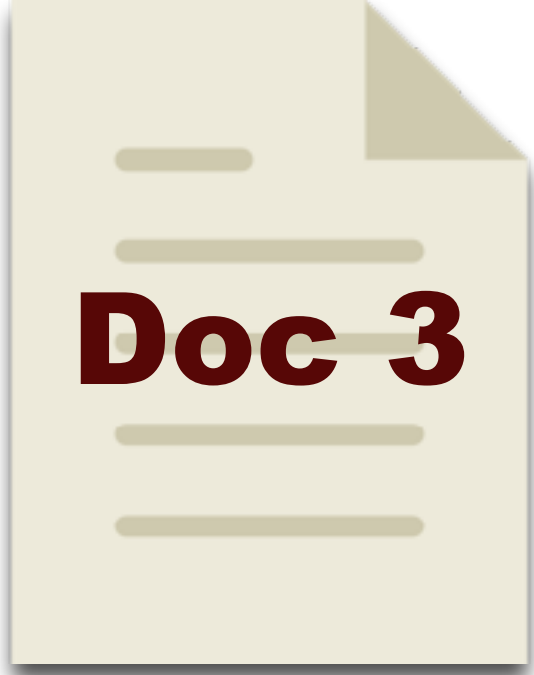
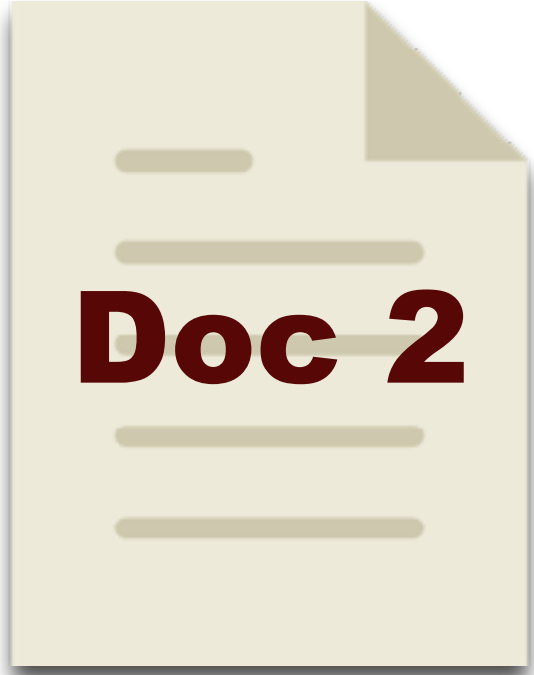
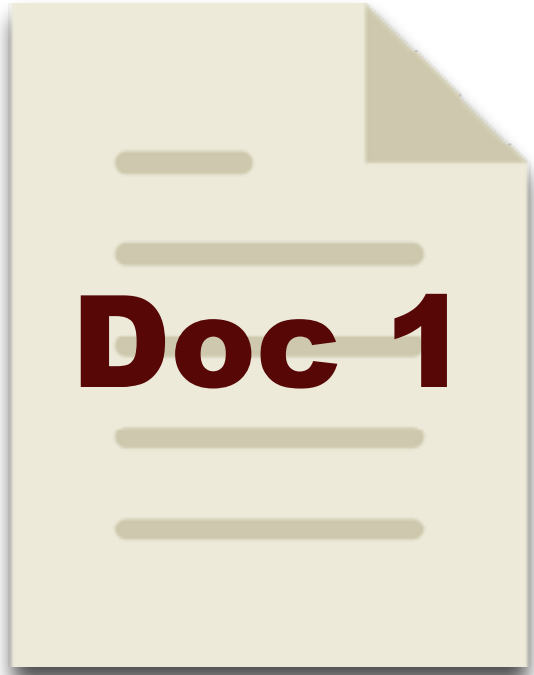
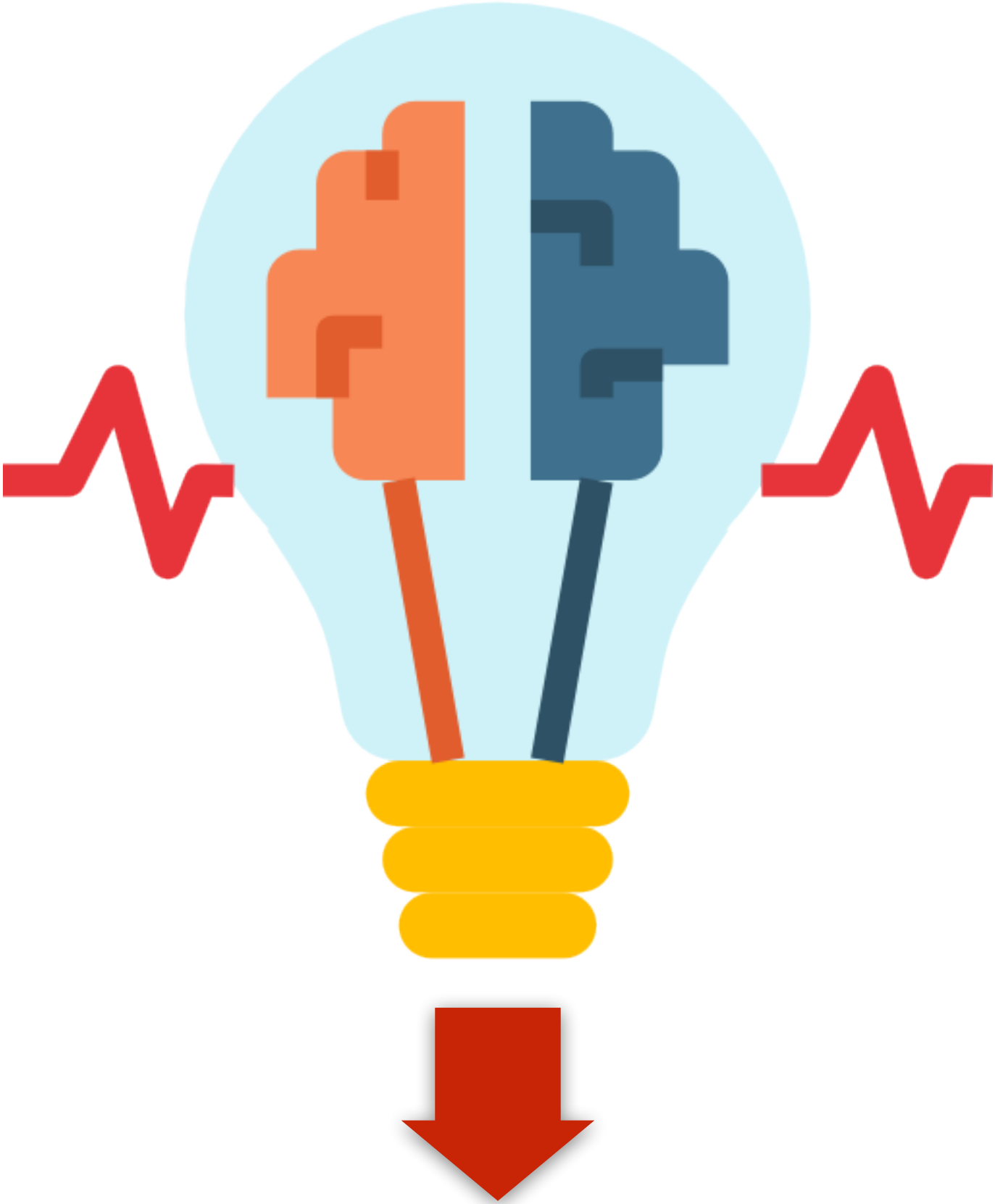
**Doc 2**

A light beige document icon with a folded top-right corner and horizontal lines representing text.

**Doc 3**

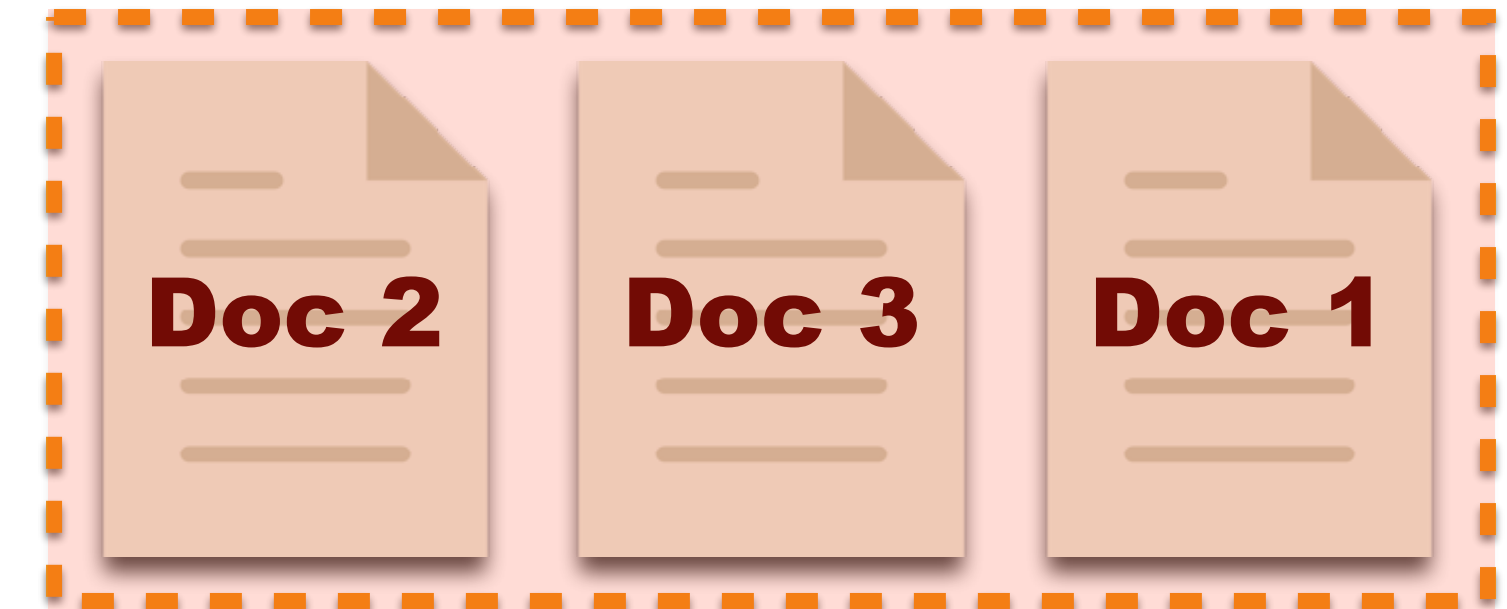
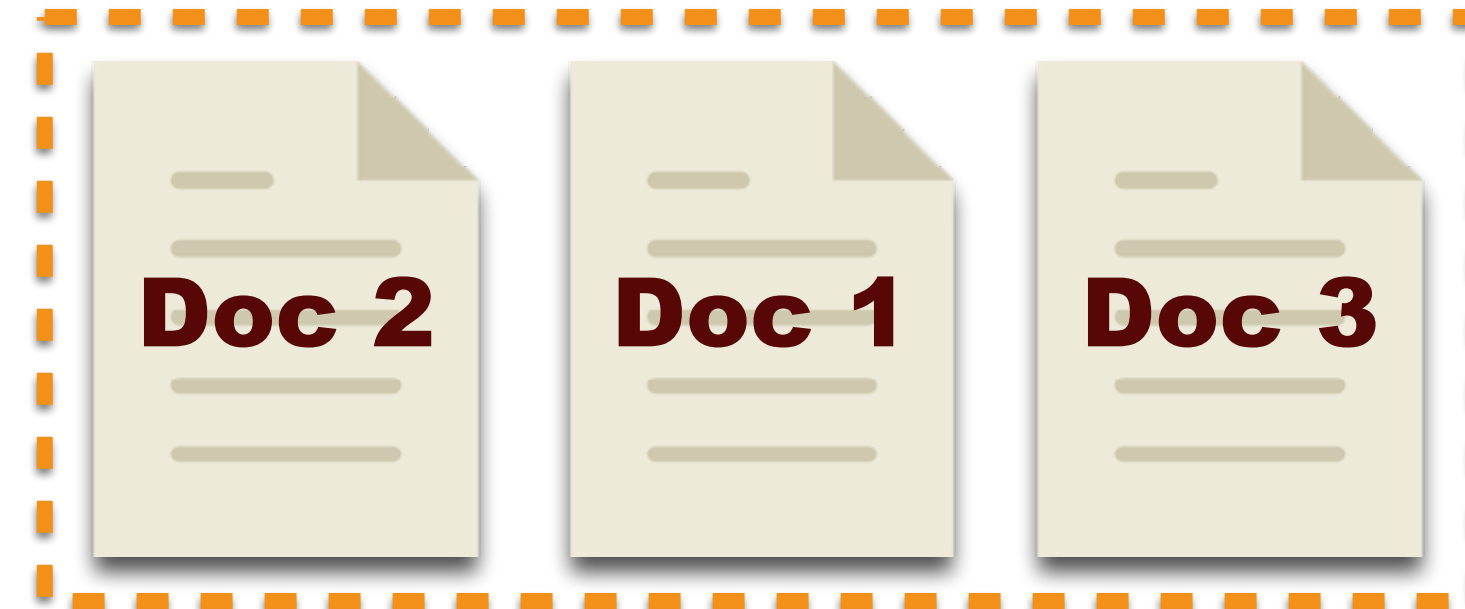
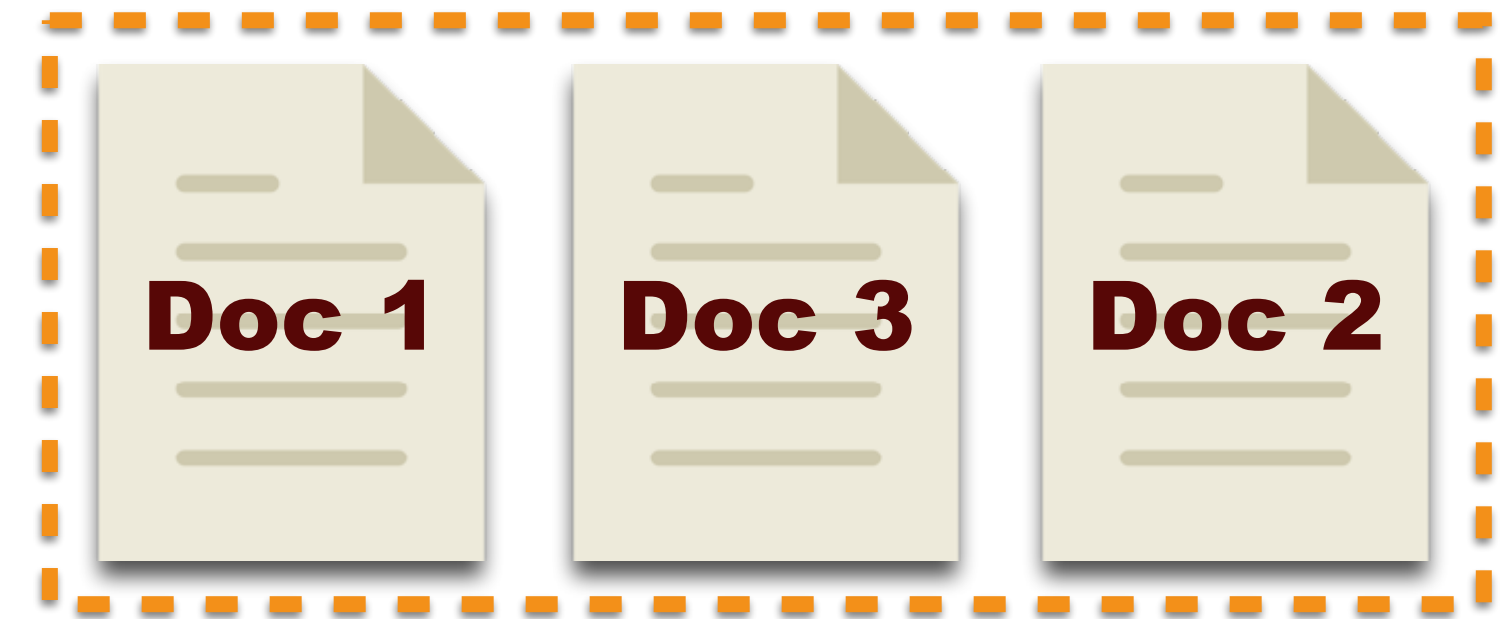
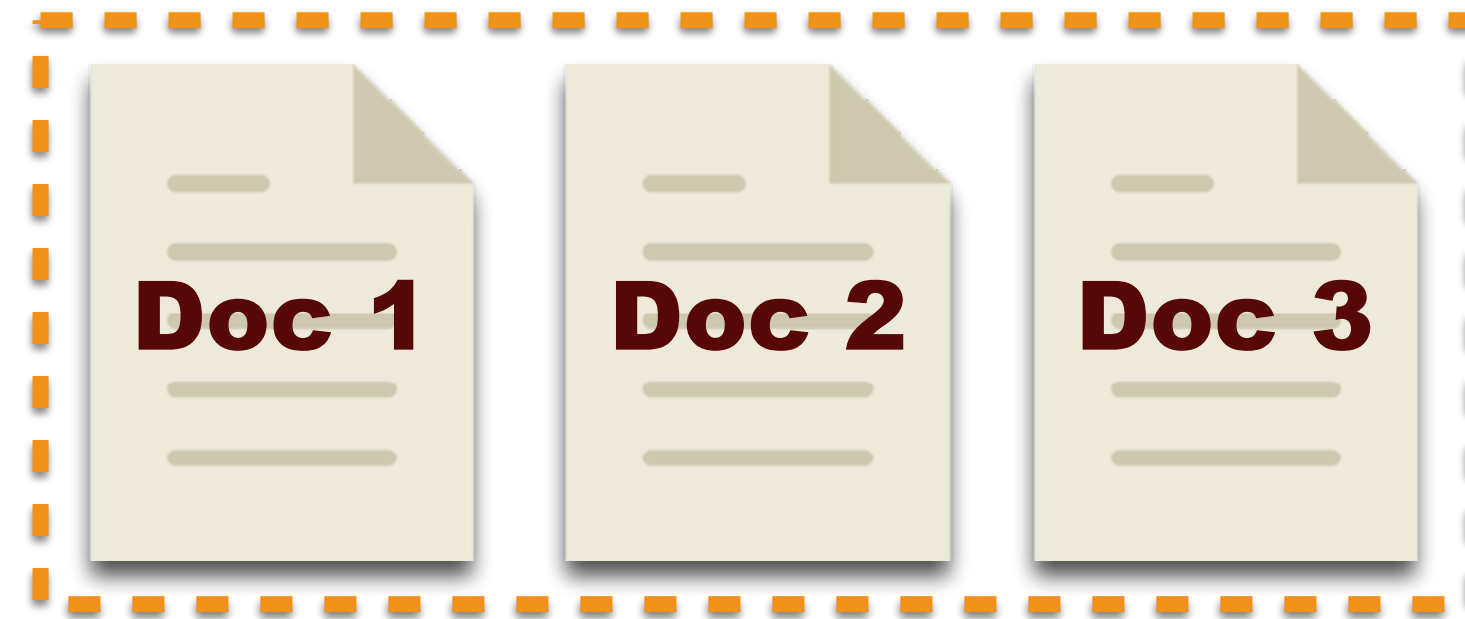
# List-wise

Query 1:  
吃到飽

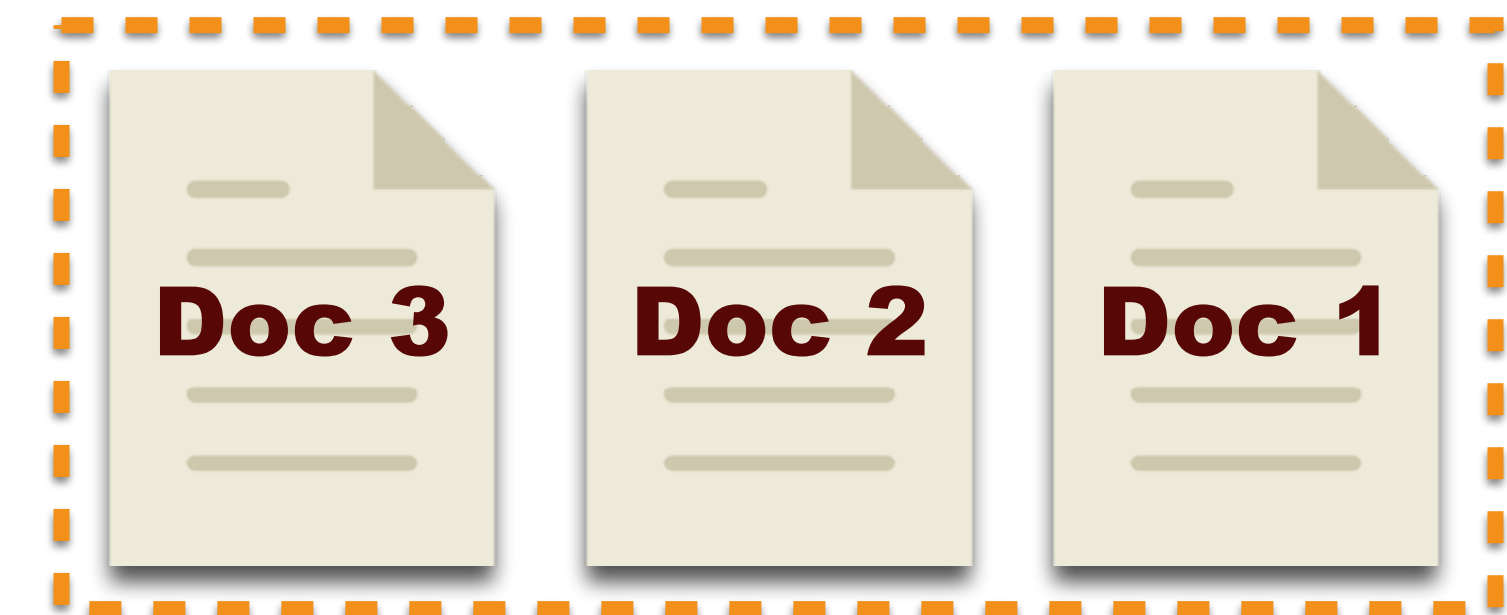
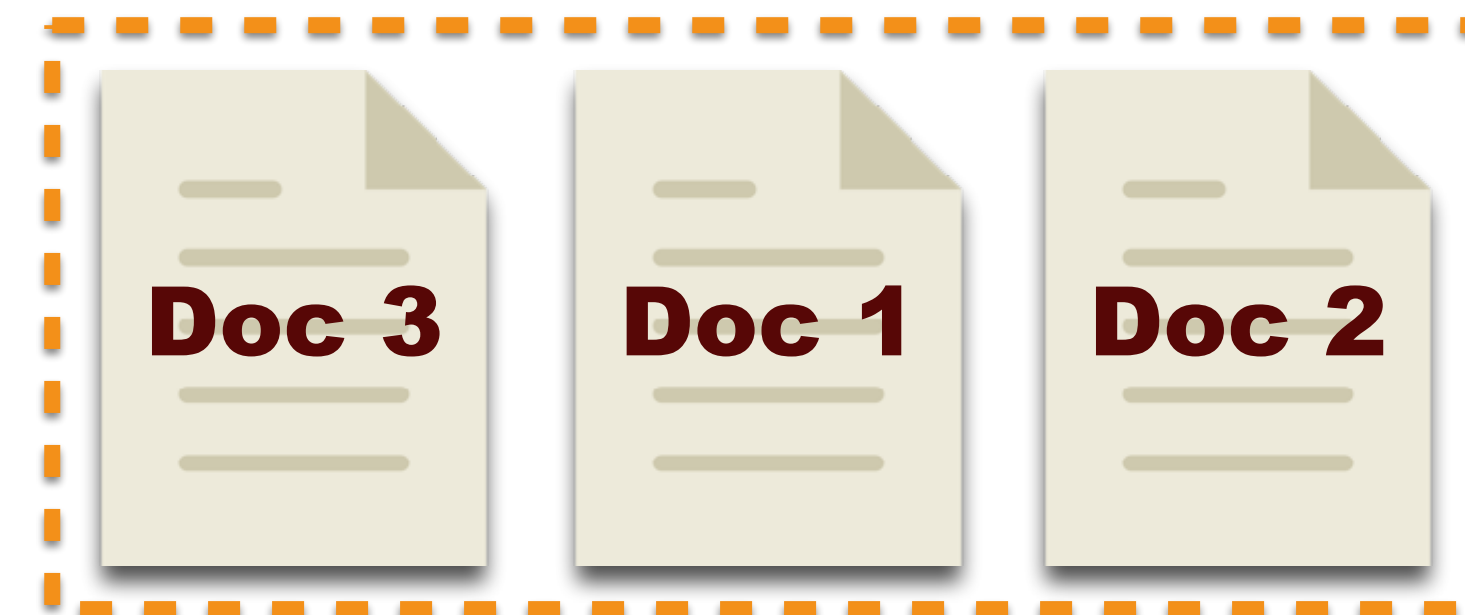


# List-wise

Query 1:  
吃到飽



機率分佈最接近真實序列



# List-wise

**Query 1:**  
**吃到飽**

高度相關

**Doc 2**

中度相關

**Doc 3**

輕度相關

**Doc 1**

# List-wise

## 1. 考慮給定查詢下的文檔集合的整體序列

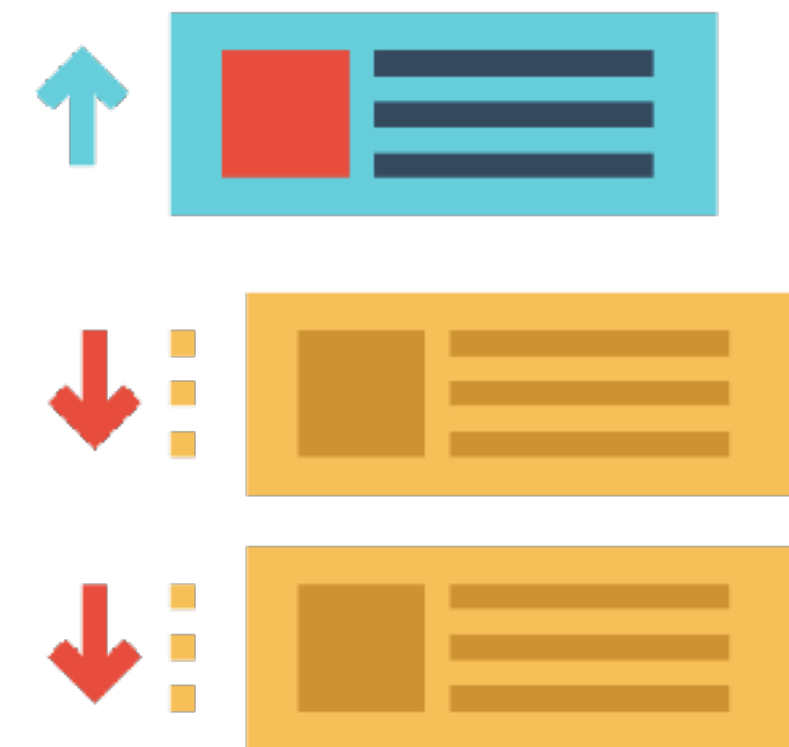
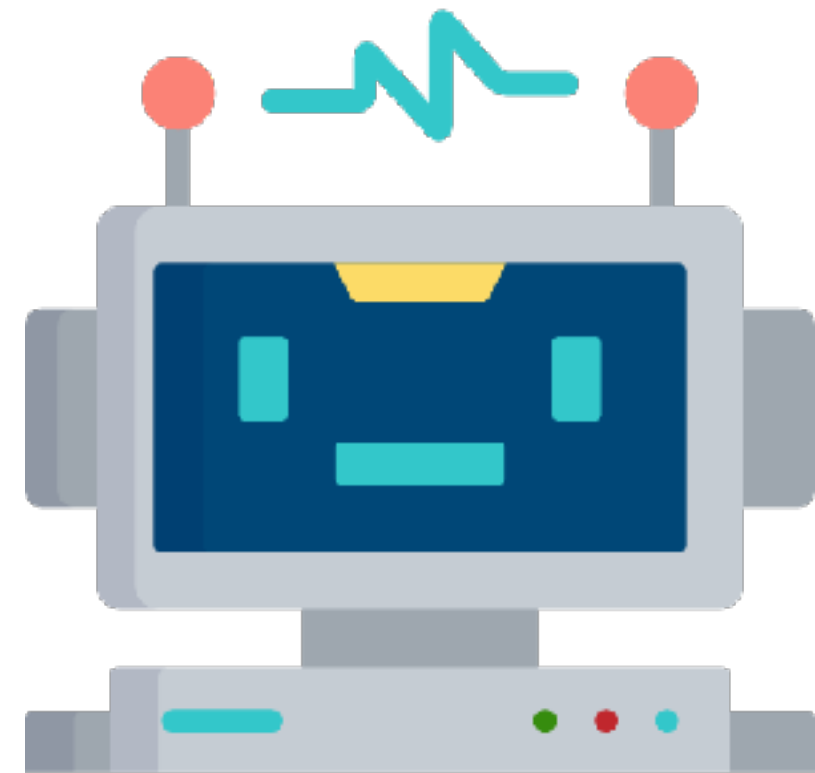
---

- 將一個 **ranking list** 作為一個 **instance** 來進行訓練
- 會考慮每個 **query** 下，所有 **doc** 之間的順序關係
- 直接優化模型輸出的文檔序列，使其盡可能接近真實文檔序列

## 2. 透過評價指標針對文檔的排序結果優化

- 如: **MAP**、**NDCG**

# Learning2Rank



**BUT**



**MAP, NDCG**

**不平滑、不連續**

**無法求梯度**

**Ranking** 的評價指標無法求梯度，  
故無法直接對評價指標做梯度下降

So...

# RankNet

---

**Pair-wise LTR**

# 優化目標

## 最小化: 錯誤的 **pair** 數目

- 錯誤的相對關係: 如果根據模型輸出 **doc<sub>i</sub>** 排在 **doc<sub>j</sub>** 前面，但真實 **label** 為 **doc<sub>i</sub>** 的相關性小於 **doc<sub>j</sub>**，便記為一個錯誤 **pair**
- 當**排序的效果越好**，有**錯誤相對關係的 **pair** 數目**就越少

# 損失函數

## 引入機率概念

- 不是直接判斷 $\text{doc}_i$ 排在 $\text{doc}_j$ 前面，而是說 $\text{doc}_i$ 以一定的機率 $P$ 排在 $\text{doc}_j$ 前面
- 以預測機率與真實機率的差距最小化，作為優化目標
- 使用 **Cross Entropy** 作為 **Cost Function**

# RankNet: 核心

## 提出一種機率損失函數

- 此為可微的代價函數
- 將不適用梯度下降求解的 **Ranking** 問題，轉為對「機率的 **cross entropy**」損失函數的優化問題，從而適用隨機梯度下降法(**Gradient Descent**)迭代更新模型參數



# **LambdaRank**

---

**List-wise LTR**

**Lambda** 代表下一次迭代  
優化的方向和強度=梯度

真正需要的是右邊紅色箭頭代表的  
方向和強度，即：更關注靠前位  
置的相關文檔的排序位置的提升

RankNet

cost=13

cost=11

橘色表示相關文檔，灰色表示不相關文檔

# Lambda

每個文檔 **下一次調序的方向和強度**

取決於同一**Query**中，

其他所有與之**Label**不同的文檔

# Lambda

**Lambda 梯度**由兩部分相乘得到:

(1) RankNet 中，交叉熵機率損失函數的梯度

(2) 交換  $\text{doc}_i, \text{doc}_j$  位置後，**IR 評價指標  $Z$  的變化差值**

pair-wise loss 概念

**X**

Delta-NDCG

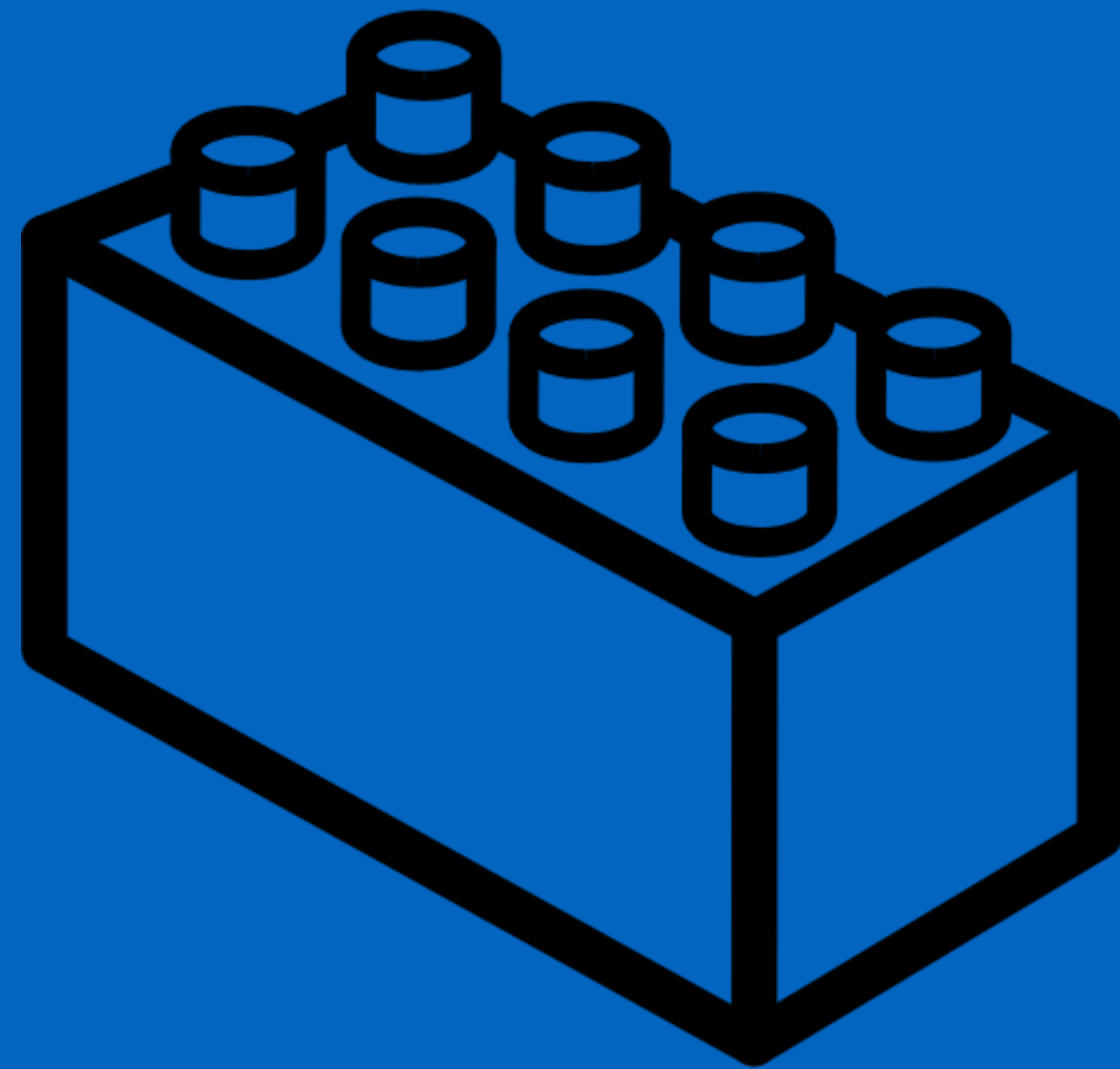
# **LambdaMART**

---

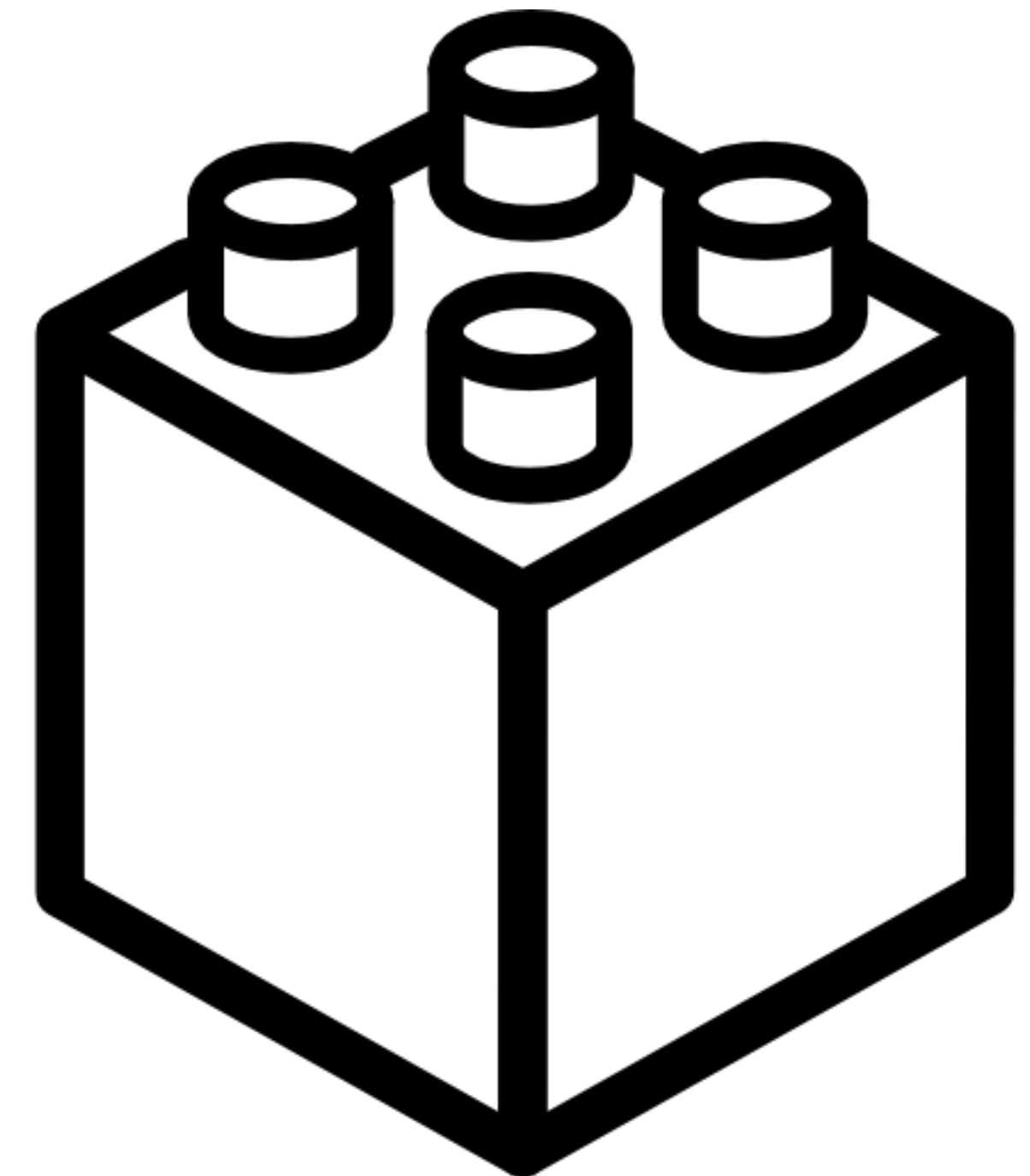
**List-wise LTR**

**Lambda** 梯度可應用於任何  
使用**梯度下降法**求解的模型

**Objective  
Function**



**Model  
Framework**



**將Lambda梯度和MART結合，  
這就是LambdaMART**



MART 不對損失函數的形式做具體規定，  
只需滿足「可導」的條件即可

梯度

框架

Objective Function

Model Framework

Lambda MART

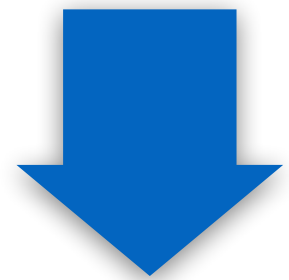
使用 **Lambda** 值代替「損失函數的梯度」，其物理含義是一個待排序文檔下一次迭代應該排序的方向。

**MART**(Multiple Additive Regression Tree)，即為：梯度提升決策樹(GBDT, Gradient **Boosting** Decision Tree)算法

**RankNet:** 引入機率概念



**LambdaRank:** 引入 Lambda



**LambdaMART:** 結合 Lambda 與 GBDT

用 boosting tree 做 gradient descent



**XGBoost:** 結合 Lambda 的 pair-wise loss 項與 GBDT

# XGBoost: pair-wise ranking

lambdaRank 到 lambdaMART 中間的過渡期

# Input

1. **Query groups** (per group ranking)
2. **Ordinal labels** of each query group
3. **Feature**: index & value

# Objective Function

**rank:pairwise**

# Evaluate Metric

**MAP**

**NDCG**

**MAP@k**

**NDCG@k**

# Output

## Predicted Ranking Score

- 針對每一個文檔，會有一個預測的排序分數
- 再透過排序分數，便可將文檔由大至小排序
- **What xgboost gives, in any case, is a value which has to be interpreted on the whole group. There is no clear threshold between values: you only know one value can be higher than the other.**

**Code**



# **Recap**

**排序不是迴歸問題**

排序關注的是各文檔間的  
相對順序關係，  
不是各文檔的預測分數最準確

**排序不能預測類別**

更不知道哪個位置  
是什麼 **Label**

# 金融業運用情境

---

# 出行銷名單

排序問題

# 在 **Ranking** 中，數據根據不同的 真實場景被區分 **Groups**

搜尋引擎: **Different Query**

建模: 切割 **segment**



**The End.**