

# 5장

## \* Evaluation

- Retrieval system은 매우 복잡하기 때문에, performance check이 매우 중요함
  - 많은 ideas는 quantitative (정량적)인 실험을 통해 영향을 미치지 않는다고 판명할 수 있음.
- IR system의 실제 유용성 (utility)을 평가.
  - 실제 응용에서 쓰일 때의 utility 반영 정량적 평가.
  - User study. Interactive IR evaluation. Qualitative evaluation.
- 다른 system, method와 비교하기.
  - user의 utility와 관련된 measure만 반영. 실제 utility를 반영할 필요는 없음.
  - test collections (test set IR evaluation), quantitative evaluation. 정량적 평가

## \* Evaluating Model

- Usability 유용함
  - 실제 user의 사용에 얼마나 useful한가?
  - User study
- Effectiveness / Accuracy 정확도
  - 검색 결과가 얼마나 정확한가?
  - 관련 document ranking이 더 top에 가도록 하는 system의 ability 측정
- Efficiency 효율
  - 얼마나 빠르게 user가 결과를 얻을 수 있는가? or 필요한 computing resource가 얼마나 되는가?
  - Space and time overhead.

## \* Information Need의 관계성

- 주어진 query에 대해, user마다 "정확하다"의 기준이 다름.
  - ex 1) query의 모든 단어를 포함하는지. : 실제 답이 아닐 수도 있다.
  - ex 2) LH information needs에 관한 논의가 결과에 포함되는지.
- Subjective (주관적) / Objective (객관적) 판단.
  - ex) "Java" → language, coffee, island 등 많은 의미로 해석됨.
  - 다른 information need를 영역에 두면, relevance에 대해 다른 판단을 하게됨.
  - General한 IR evaluation을 위해 ambiguous (어매한) query는 피해야함.

정확도

효율성

## \* Effectiveness vs. Efficiency

- Effectiveness : 모델의 답변이 사용자 관련 ranking과 얼마나 잘 일치하는지.
- Efficiency : 모델이 답변을 내는데 소요되는 시간 및 space 비용.
- 여러 요소 (interface, query 제작, relevance feedback 등)가 effectiveness 및 efficiency에 영향을 줌.
- 여러 요소를 한번에 고려하기 어려워서, evaluation은 tight한 실험 환경에서 이루어짐.

- 정확도에 약간의 개선이 있지만 흐름성이 크게 떨어지는 경우, 실제 system에 적용되는 양을 수 있다.

반대로, 매우 빠르더라도 결과가 안좋으면 적용 불가.

- 2단계

1. 먼저, 정확도를 향상시킨다.

2. 사용 가능한 정도의 정확도를 가정하고 보면, 흐름적인 구현을 확장 한다.

### \* Cranfield Evaluation Methodology

- Retrieval system의 실험적 테스트 (1960s)

• 1998 개 요약, 225 query → 현재로선 적음.

- 자사용 가능한 test collection, measure define,

• 문서 set (문서 수집 시점) → topic/query set (사용자 query 시점, 50개 이상의 정보요구) → Relevance 판단 (query를 만든 user가 판단)  
→ 이상적인 ranked list (gold standard, ground truth). 기본 방법은 binary assessment (관련 O or X)

• test collection evaluation: 각 IR system의 결과와 ground truth를 비교.

### \* IR evaluation 데이터 종류

h21

- ACM : Bibliographic records. 320671 d, 6471 q. query 및 relevance는 computer scientists들이 생성.

- AP, GUV2 : NIST

- Reuters, Newsgroups

- TREC : Conference.

• 데이터 공개 주체: NTCIR, CLEF

### \* Relevance Judgements : Explicit 방법 : Pooling

- 모든 문서에 대해 relevance check는 불가능하다 : 일부 (pool) 만 하기

#### Pooling

• Search engine / algorithm 사용. 상위 K개 (20~50)는 pool로 병합. (중복은 제거).

• ol pool LH 문서들을 analyst에게 임의의 순서로 제시.

- relevant judgement는 새로운 search technique에 대한 정확한 비교를 하기 충분함.

### \* Kappa Statistics : Explicit

- 같은 information need에 대해 여러 명의 Judge가 있으면 그 Judge들이 얼마나 연관성이 있는가?

- Categorical judgement을 위함. : 우연히 일치하는 것을 고려하기.

$$\cdot \text{Kappa} = \frac{P(A) - P(E)}{1 - P(E)}$$

•  $P(A)$ : Judge들이 동의한 비율

$P(E)$ : 우연히 동의한 비율

7.21.2

		Yes	No	Total	
		Yes	300	20	320
Judge 1	Yes	300	20	320	$\rightarrow \text{judge 1} : \begin{array}{l} \text{Yes} \\ \text{No} \end{array} = \frac{320}{400}$
	No	10	70	80	
		Total	310	90	400

		Judge 2		
		Yes	No	
Judge 2	Yes	$\frac{310}{400}$	$\frac{90}{400}$	
	No			

✓.  $P(A) = \frac{300+10}{400} = 0.925$  : 실험에서 나온 결과. 실제 두 judges가 일치한 확률

✓.  $P(E) = \frac{320}{400} \times \frac{310}{400} + \frac{80}{400} \times \frac{90}{400} = 0.665$  : 우연히 일치한 확률

•  $Kappa = \frac{P(A) - P(E)}{1 - P(E)} = \frac{0.925 - 0.665}{1 - 0.665} = 0.776$  : (agreement)  $\rightarrow$  두 judges 사이 통일성이 있다.

- Kappa value는 두 judges가 완벽 일치하면 1, 우연한 일치 뿐인 경우 0.

- random 보다 낮으면 ( $P(A) < P(E)$ ) negative : Negatively correlated. (-) 가능.

- Rule of thumb

- 0.61 ~ 0.8 정도는 대체로 fair agreement가 존재함.
- 0.61 이하는 평가에 있어 모호한 근거를 제공하는 data임을 나타냄.  $\rightarrow$  data의 문제라는 것인가?

- 2명 이상 judges면 2명씩 Kappa 구해서 average한 지표 쓸.

### \* 평가에서 Query Logs 사용. : Implicit 방법

- Implicit relevance feedback을 사용하여, 평가  $\alpha$ 로 생성하는 비용을 줄인다.

- 검색 후 사람이 click하는 문서 = relevant하다라고 판단.

- Query log data는 더 광범위하고 현실적인 평가를 지원한다.

- explicit relevance judgement 보다 정확하지 않다. (noise의 가능성 ↑)
- privacy concern : 개인 정보가 포함될 수 있는 query를 제거하여 익명성 보호.

- 종류

- user session identifier : user login id, search toolbar id, cookies  
; session은 제한된 시간 동안 search engine에 저장된 유저들.
- query terms : user가 입력한 그것.
- clickthrough data : list of URL, 결과 목록 순위, 클릭 여부 등.
- timestamps : 머문 시간.

- 다른 Implicit Relevance Judgement

- dwell time : 처음 click부터 page로 다시 돌아가거나 search it 일을 때까지 경과 시간.
- search exist action : search application 내 동작들. 다른 URL 입력, 브라우저 창 닫기, 시간 초과 등

- Clickthrough data

- 너무 유명한 pages는 평가는 되지 않음.
- 두 documents 사이 사용자 preference 예측으로 하면됨.
- $d_1, d_2, d_3, d_4$ 가 return된 후,  $d_3$  클릭 후 skip next -  $d_3 > d_4$ , skip above -  $d_3 > d_1, d_2$

## <Evaluation Measures>

\* Confusion Matrix

"그냥LF 내걸까?"

실제 결과

		Relevant		Non-relevant			
		Retrieved	True Positive TP	False Positive FP	True Negative TN	True / False : 맞음 / 틀림	
LH	Retrieved	True Positive TP	False Positive FP	True Negative TN	Pos / Neg : 내 결과		
결과	Not retrieved	False Negative FN					

### ① Accuracy

Not good.

$$\cdot \text{accuracy} = \frac{TP+TN}{TP+TN+FP+FN} : \text{실제 맞음 비율}$$

- IR에서는 좋은 measure가 아니다.

ex)	Doc	0 <sub>21</sub>	0 <sub>1</sub>	0 <sub>2</sub>	0 <sub>1</sub>	0 <sub>2</sub>
	D <sub>1</sub>	+	+	-	+/-	+/-
	D <sub>2</sub>	-	+	-	+/-	+/-
	D <sub>3</sub>	-	-	-	+/-	+/-
	D <sub>4</sub>	-	+	-	+/-	+/-
	D <sub>5</sub>	-	+	-	+/-	+/-
	D <sub>6</sub>	-	-	-	+/-	+/-

accuracy =  $\frac{3}{6} < \frac{5}{6}$

실제유동성      0<sub>1</sub>      >      0<sub>2</sub>

IR에서는 대부분이 relevance 문제이다.

accuracy 가지고는 IR평가 어렵다.

### ② Precision and Recall

$$\text{precision} = \frac{\text{실제} +}{\text{내가 찾은} +} = \frac{TP}{TP+FP}, \quad \text{Recall} = \frac{\text{내가 찾은} +}{\text{실제} +} = \frac{TP}{TP+FN}$$

- Precision : 내가 찾은 것 중 얼마나 정답인가.
- Recall : 실제 정답 중 내가 얼마나 많이 찾았나.
- 

		Relevant	Non-relevance	
		Retrieved	Non-retrieved	
		TP	FP	precision
Retrieved	Non-retrieved	FN	TN	
				recall

### ③ F-measure

- harmonic mean of precision and recall.

$$\cdot F = \frac{1}{\alpha \cdot \frac{1}{P} + (1-\alpha) \cdot \frac{1}{R}} = \frac{(\beta^2 + 1) PR}{\beta^2 P + R}, \quad \beta^2 = \frac{1-\alpha}{\alpha}$$

$$\cdot \alpha = 0.5 (\beta=1) \text{ 이면 } F = \frac{2PR}{P+R} : \text{F1 score.}$$

•  $\alpha < 0.5 : \beta > 1 : R$ 의 영향이 더 큼.

•  $\alpha > 0.5 : \beta < 1 : P$ 의 영향이 더 큼.

- Harmonic mean의 이유?

• harmonic mean은 arithmetic mean과 geometric mean의 중간.

• P, R이 각각의 범위는 minimum과 maximum

정답: 6개	Ranking:	■	□	■	□	■	□	□	□	■
Precision	$\frac{1}{6}$	$\frac{1}{2}$	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{4}{5}$	$\frac{5}{6}$	$\frac{5}{7}$	$\frac{5}{8}$	$\frac{5}{9}$	$\frac{6}{10}$
Recall	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{2}{6}$	$\frac{3}{6}$	$\frac{4}{6}$	$\frac{5}{6}$	$\frac{5}{6}$	$\frac{5}{6}$	$\frac{5}{6}$	$\frac{6}{6}$

- 단계별(짧는 쪽쪽) precision과 recall을 비교할 수 있다. : list of recall - precision values.

- 장점: 훨씬 더 상세한 정보.

- 단점: 너무 길다. 비교가 어렵다.

- Pre-defined 된 position에서의 precision, recall을 비교할 수 있다. : 10 or 20 정도 사용

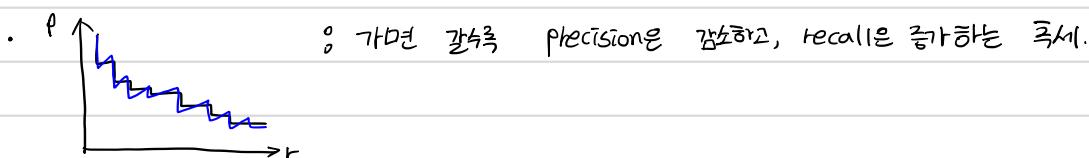
: 특정 위치에서의 precision이 높으면, recall도 높다 : recall은 불모가 고정되어 있기 때문  
두 ranking에 대해

#### (4) Precision - Recall Curve

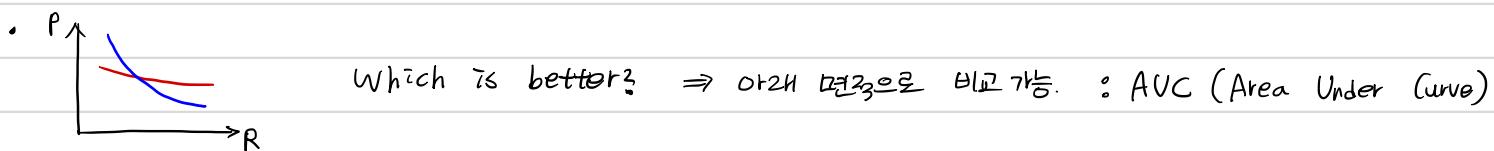
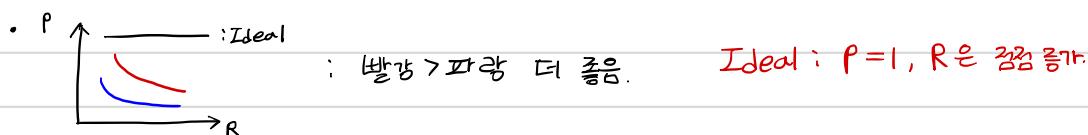
Recall은 non-decreasing function.

- KI 번째 document가 irrelevant면, recall은 K번재와 똑같지만 precision은 떨어진다.  
relevant면, precision과 recall 다 올라간다.

) → 각 단계마다 (P, R) 쪽을.



- 어떤게 좋은가?



- 11-point interpolated Precision

- Recall이 0.1씩 증가할 때마다 쪽. (0~1 : 11개의 점)

multiple query : MAP

순위: NDCG

#### (5) Average Precision (AP)

: list 보다 AP

- 특정 조건을 만족하는 점들에 대한 precision value의 평균값. =  $\frac{\sum \text{precision}}{\# \text{ of relevant}}$

ex) When recall increase. (맨 위 오른쪽 예시 : 빨간 부분)

$$\Rightarrow (1 + \frac{2}{3} + \frac{3}{4} + \frac{4}{5} + \frac{5}{6} + \frac{6}{10}) / 6 = 0.78 : \text{Recall 맨처음 포함.}$$

- 높을수록 더 좋은 결과이다.

- 왜 AP를 쓰는가?

- list of recall - precision values를 다 보기엔 어렵다. 단 1개의 Value로 요약 가능함.

- 단점

- Multiple query에 대해 measure 불가능.

- 실제로는 top rank가 가장 중요한데, AP는 position의 중요도를 고려하지 못함.

#### (6) MAP (Mean Average Precision)

- 각 query들에 대한 AP의 평균값. 가정: 사용하는 각 query에 대해 여러 문서를 찾길 원함.

## \* Reciprocal Rank (상호 순위)

- 정답 문서가 앞에서 등장하는게 좋음.
- Reciprocal rank : 첫번째 관련 문서가 등장(찾음)한 위치. ex) Tr, T, Tr, Tr  $\Rightarrow \frac{1}{2} = \frac{1}{\text{Rank}_i}$
- MRR (Mean Reciprocal Rank)
  - 모든 query들에 대한 Reciprocal rank의 평균.

$$\text{MRR} = \frac{1}{|\Omega|} \sum_{i=1}^{|Q|} \frac{1}{\text{Rank}_i}$$

## \* R-Precision

- 가정 : 모든 relevant 문서 Rel을 알고 있음.
- Precision @ Rel : top relevant document의 precision 값.
- Break-even point : precision과 recall이 같은 relevant 문서 등장 지점. ( $FP = FN$ )
  - 가장 처음 precision.
  - 지금까지 찾은 것 중 관련 없는 것 개수.
  - 앞으로 찾아야 할 관련 있는 문서 수.

## \* DCG (Discounted Cumulative Gain)

- 강하게 관련있는 문서는 Marginal(평균)보다 중요.
- 관련 문서의 순위가 낮을수록, user에게 더 중요한 정보임.
- Relevance(관련성)을 매기는 measure
  - ranking 앞에서 더해지고, 뒤에서 줄어드는 gain.
  - $DCG_p = \sum_{i=1}^p \frac{\text{rel}_i}{\log_2(i+1)} = \text{rel}_1 + \sum_{i=2}^p \frac{\text{rel}_i}{\log_2(i+1)}$ 
    - rel<sub>i</sub> : Binary relevance score at position i.  $\in \{0, 1\}$
    - $\log$  안쓰면 나중 등장 문서의 gain 감소.
    - 밑이 증가하면 같은 감소량 : 0이든 증가.
    - $\log$  : 크게 줄어들지 않는 것을 의미. ex) 8번째 위치  $1 \rightarrow \log: \frac{1}{8}$   
 $\log: \frac{1}{3}$

## \* NDCG (Normalized DCG)

- $nDCG_p = \frac{DCG_p}{IDCG_p}$
- IDCG<sub>p</sub> : Ideal한 경우 (ex. 1 1 1 1 1 1 0 0 0 ... )
  - relevant 문서가 K개면,  $IDCG_p = \sum_{i=1}^K \frac{1}{\log_2(i+1)}$
- NDCG with graded scores
  - NDCG식의 rel<sub>i</sub> = {0, 1} 이 아니라, 관련도를 매긴 점수들이 됨.
  - 이례면 IDCG는 높은 점수부터 차차 찾는 경우가 됨.
    - $\rightarrow 3, 3, 2, 4, 1, 2, 3, \dots$
    - $\rightarrow 4, 3, 3, 3, 2, 2, 1, \dots$

## \* Rank Correlation

- 수치가 아닌, ordering 기반. 서로 다른 두 ranking oil 대한 correlation 측정

### 1) Kendall's Tau distance

- $K(\sigma) = |r_i(x_i) < r_i(x_j) \text{ and } r^*(x_i) > r^*(x_j)|$
- Ideal과 비교했을 때, 각 두 pair에 대해 order가 바뀌는 경우의 개수.
- time:  $O(n^2)$  : 오래 걸림.

## 2) Spearman's footrule distance

- 특정 순서에 대해, Ideal 대비 떨어진 거리를 더함.

$$- F(\sigma) = \sum_i |i - \sigma(i)|$$

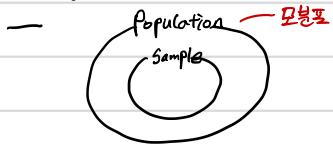
- time :  $O(N)$

### Kendal vs. Spearman

- $K(\sigma) \leq F(\sigma) \leq 2K(\sigma)$  : 충분하다.

- Weight 통합 방법
  - Element weights : 중요한 2개 swap, 중요하지 않은 2개 swap 따로 고려.
  - Position weights : header의 2개 swap, tail의 2개 swap 따로 고려.

## \* Significance Test



- Sample 들의 분포와 Population 의 분포(모본포) 와 같은지, 다른지를 check.

- Statistical test = Null hypothesis ( $H_0$ , 기존 분포를 따른다고 가정)

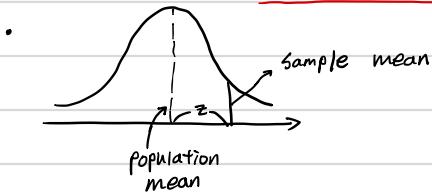
$\hookrightarrow$  Null hypothesis를 반증하는 강력한 증거를 위한 test = significance test.

- 두 model 사이 평가로도 쓰임.

- population mean =  $\mu_0$ , population standard deviation =  $\sigma$

sample mean =  $\bar{x}$ , sample size =  $n$  이라 하면,

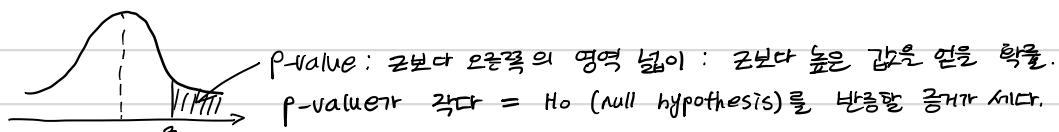
Critical value  $Z = \frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}}$  : 원래 모본포에서, 내가 찾은 sample의 평균이 어디쯤인가?



- 이후, 구한 z가 충분히 가능한 값인지 아닌지를 판단해야 함.

### Finding P-Value

- $H_0$ 가 true일 때 z를 얻을 확률을 보자.



### Significance level (유의 수준) : $\alpha$

- $H_0$ 를 reject 할 수 있는 최대 p-value ( $=\alpha$ )

- $p\text{-value} > \alpha$  :  $H_0$  Accept.

$p\text{-value} \leq \alpha$  :  $H_0$  Reject :  $p\text{-value}$ 가  $\alpha$ 보다 작다면, 이 모본포를 뛰어 넘을 것이다.

- 순서 : sample mean ( $\bar{x}$ ), 개수 ( $n$ )  $\rightarrow$  z-score :  $\frac{\bar{x} - \mu_0}{\sigma / \sqrt{n}}$   $\rightarrow$  p-value  $\rightarrow$  (증해인)  $\alpha$ 보다 작은가?

yes  
사실 분포!

No  
기존 분포 ( $H_0$ )

## \* Significance test in IR

- 두 알고리즘이 성능적으로 다른가?

- 같다 : Null hypothesis ( $H_0$ )
- 다르다 (baseline A 보다 B가 좋다)

- ranking 결과는 query sample에 기반한 것이다.

- significant test 결과

- true :  $H_0$ 가 reject됨. 새 불포함.
- false :  $H_0$ 가 accept됨. 기존 불포함.

- 두 Model 비교

- data : 각 query에 따른 효율성 특징값.

• query A B → B-A

$$\begin{array}{cccc} 1 & 25 & 35 & 10 \\ 2 & 43 & 84 & 41 \end{array}$$

$$\begin{array}{cccc} 3 & 39 & 15 & -24 \\ 4 & 15 & 75 & 0 \end{array}$$

$$\vdots \quad \vdots \quad \vdots \quad \vdots$$

$$\textcircled{2} \quad \overline{B-A} = 21.4$$

$$\textcircled{3} \quad \sigma_{B-A} = 29.1 \quad (\text{표준편차})$$

$$\textcircled{4} \quad Z = \frac{21.4}{\sigma_{B-A} / \sqrt{10}} = 2.33$$

$$\textcircled{5} \quad \sqrt{\frac{21.4^2}{29.1^2 / 10}}$$

$$\text{대푯값 표준화.}$$

$$\downarrow$$

→ p-value <  $\alpha = 0.05$  이므로 B is better than A.

- O(1)

- 데이터는 정규분포에서 sampling 된다고 가정. ⇒ randomization (분포가 없다) test에서 비슷한 결과를 내낸다.
- 평가 data가 interval scale을 measured되었다고 가정. ; 크의 크기 등을 중요하지 않다. 단지 순서만 중요. ; Wilcoxon signed-rank test, sign test는 가정이 적지만 더 강력하다.

- A/B test

- 현재 system과 새로운 system을 딱 1개만 대조.
- traffic 비율로 평가. (사람들이 더 많이 쓰는 것)