Search Process

Evaluation

Overview

X is better than Y on task Z along some dimension W

X ... a system / algorithm / method that you have created

Y ... a baseline system / current state-of-the-art

Z ... the specific problem you're trying to solve

W ... how do you measure what is "better"

For keyword-based searches in medical databases,

Pseudo-Relevance Feedback

will provide better search results than Topic Modeling

as measured by mean average precision of the ranked list.

Research Hypotheses

Evaluation

How do you decide if a search system is good?

Effectiveness:

- does your system find good results?

Efficiency:

- how fast does it find them?

Other considerations:

- do the users like the experience?
- easy to formulate the query?
- easy to browse the results?

Automated evaluation

No user in the loop → test early, test often

- our intuition about what works is often very wrong
- phrases, WordNet, "core" terms, linguistic processing
- term weighting, massive expansion, connectedness
- success stories: IR, MT, ASR...

Cranfield paradigm:

- fixed set of queries (topics)
- fixed set of documents (corpus)
- fixed set of relevance judgments
- effectiveness measure: results = relevant docs?

Topics and corpora

Topics

- intended to mimic real information-seeking tasks
- gov. information analysts, patent officers

Corpora

- news, scientific, legal, patents, web-pages
- beyond text: speech, images, videos, DB records

Annual competitions in IR (blind evaluation)

- US: Text REtrieval Conference (TREC) – 117 groups
- EU: Cross-Language Evaluation Forum (CLEFT)
- Asia: NTCIR
Topics and corpora: example

<table>
<thead>
<tr>
<th>Topic example</th>
<th>TREC query 704</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>cat in pet therapy</em></td>
<td></td>
</tr>
<tr>
<td><strong>Query Description</strong></td>
<td>How are cats or animals used in therapy for humans and what are the benefits?</td>
</tr>
<tr>
<td><strong>Query Narrative</strong></td>
<td>Relevant documents must include details of how pet- or animal-assisted therapy has or has not been used. Relevant details include information about pet therapy programs, experiences of the people in which pet therapy is used, the benefits of this type of therapy, the degree of success in this therapy, and any laws or regulations governing it.</td>
</tr>
</tbody>
</table>

**Research corpora**

<table>
<thead>
<tr>
<th>Collection</th>
<th>Number of documents</th>
<th>Number of queries</th>
<th>Number of words/doc</th>
<th>Average number of words/query</th>
</tr>
</thead>
<tbody>
<tr>
<td>CACM</td>
<td>43,000</td>
<td>3,000</td>
<td>14</td>
<td>140</td>
</tr>
<tr>
<td>AP</td>
<td>242,585</td>
<td>3,400</td>
<td>70</td>
<td>50</td>
</tr>
<tr>
<td>CLEF</td>
<td>25,200</td>
<td>400</td>
<td>15</td>
<td>25</td>
</tr>
<tr>
<td>CORR</td>
<td>800</td>
<td>100</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Class Size</th>
<th>980,400,000</th>
<th>210,000</th>
</tr>
</thead>
</table>

**Relevance judgments: sampling**

- Pooled (used in TREC)
  - top k docs from every participating system (50-200)
  - merge into a pool, remove duplicates
  - randomize, present to annotators

- Search-guided
  - run query, read until convinced no more rel. docs
  - re-formulate query using found rel. docs, repeat

- Sampling
  - estimate bounds on the size of relevant set

**Evaluation: building blocks**

<table>
<thead>
<tr>
<th>Retrieval?</th>
<th>Retrieved?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>True Positives (TP)</td>
</tr>
<tr>
<td>No</td>
<td>True Negatives (TN)</td>
</tr>
<tr>
<td>FF</td>
<td>False Positives (FP)</td>
</tr>
<tr>
<td>FN</td>
<td>False Negatives (FN)</td>
</tr>
</tbody>
</table>

**Why not accuracy?**

- Retrieval ... a kind of classification
  - document $\rightarrow$ (relevant, non-relevant)
  - standard measure: $Accuracy = \frac{correct}{total} = \frac{TP + TN}{N}$
  - or use Error = 1 - Accuracy

- Meaningless:
  - accuracy 99.99% for any search algorithm
  - for any query, almost all documents are non-relevant
  - often best strategy is to retrieve nothing

**F-measure**

- A variant of accuracy not affected by negatives
  - single-value measure (compare, tune systems)
  - Harmonic mean of $P$ and $R$: $F_p = \frac{P + R}{P \cdot R}$
  - $\beta$ ... relative importance of recall and precision
    - popular setting: $\beta = 1$, which gives $F_1 = \frac{2PR}{P + R}$
    - heavily penalizes small values of $P$ and $R$
  - Geometric interpretation:
    - overlap between relevant, retrieved
      $F_1 = \frac{2PR}{P + R}$
    - recall $= \frac{TP + FN}{TP}$
    - precision $= \frac{TP}{TP + FP}$

**Comparing recall / precision**

- Which of the following is a better system?
  - system A: recall = 50%, precision = 57%, $F_1$ = 53%
  - system B: recall = 100%, precision = 40%, $F_1$ = 57%

- Could be the same exact system
  - using different threshold settings
  - $R/P$, $F_1$ comparisons often meaningless
  - more informative to compare ranking against ranking

**Cranfield Paradigm vs. User Study**

- **User study (interactive):**
  - U enters a query
  - S: set of search results
  - U interacts with results
  - scrols list, views items, tags them, copies from them, types different queries
  - S: monitors user actions
    - mouse movement, eye-tracking, click-log, search history
    - evaluates utility/satisfaction
      - accomplished the task?
      - satisfied? easy to use?

- **Cranfield (automated):**
  - fixed set of queries
  - S: set of search results
  - R: relevant documents
    - known set of useful items
    - independent of user
  - evaluate S vs. R
    - how much of R did we find?
    - how much non-R (junk)?
    - is R ranked at the top?
  - simulate use interaction
    - pick top-ranked R-term
    - relevance feedback (new S)

**Relevance judgments**

- Which documents are relevant for a query
  - usually binary: (relevant, non-relevant)
  - sometimes graded (more expansive to obtain)
  - multiple annotators: reduces accidental mistakes
  - issues: who? instructions? level of agreement?

- Exhaustive: judge every document
  - usually inflexible (25m docs x 150 queries)
  - occasionally done for a new task (e.g. TDT1)

- Why not judge just the top 10?
  - notion of coverage important for lawyers, analysts
  - want to reuse judgments to test new algorithms
  - which will find docs none else found in their top 10
Recall / Precision and ranking
- Search engine produces a ranking, not a set
  - can compute recall, precision at every rank
  - ranking #1
    - the relevant documents
  - ranking #2
    - the relevant documents

Recall / Precision / F1 vs rank
- Crossover point: \( P = R = P_{c} \)
- \( R(r) = \max \{ P : R \geq R \} \)
- \( (P_{c}, R_{c}) \) ... raw values

Recall-precision plot: interpolation
- Interpolation hard: \( P(0), \) not a function
- On average precision drops as recall increases
- Define interpolation to preserve monotonicity
  - max. precision observed at recall \( R \) or higher
  - \( P(R) = \max \{ P : R \geq R \} \)
- Average interpolated \( P \)
  - at standard levels of \( R \)

Interpolated recall-precision plot
- Averaged over 50 queries
- Performance for all user types
  - high-precision and high-recall
- Meaningful system comparisons
  - A worse than B if recall important
  - A always better than C
  - dominates at all recall levels

Mean Average Precision: example
- Based on relative utility of relevant documents
  - most useful at top ranks
  - utility decreases with rank
  - allows graded degrees of relevance (rel.)
- Normalized version (NDCG):\n  - divide by DCG of best possible ranking: rel. \( = 1 \ldots 10000 \ldots \)
- MAP has a similar effect:
  - \( MAP = \sum_{r} \frac{rel_{r}}{log(r+1)} \)
  - \( \text{utility of a relevant document at rank } r \)

Discounted Cumulative Gain
- \( DCG_{g} = \sum_{r} \frac{rel_{r}}{log(r+1)} \)
- \( \text{average precision query } 1 = (1.0 + 0.67 + 0.35 + 0.44 + 0.5)/5 = 0.62 \)
- \( \text{mean average precision } = (0.62 + 0.44)/2 = 0.53 \)

Mean Average Precision
- Averaged over 50 queries
- Performance for all user types
  - high-precision and high-recall
- Meaningful system comparisons
  - A worse than B if recall important
  - A always better than C
  - dominates at all recall levels

Query Logs
- Logs used to tune / evaluate search engines
  - session id, query, documents, click, timestamps
- Click # relevance judgement
  - correlated, but noisy
  - generate preferences
- Aggregate clicks to reduce noise
  - click deviation: \( CD(d, p) = O(d, p) - E(p) \)
  - \( O(d, p) \) ... click rate for document \( d \) in position \( p \)
  - \( E(p) \) ... expected click rate for position \( p \)
  - average click deviation correlates with relevance
    - consistently clicked despite low rank → probably relevant
    - consistently not clicked despite high rank → probably not
Evaluation with preferences

- Kendall tau rank correlation coefficient: \( r = \frac{P - Q}{P + Q} \)
  - \( P, Q \) number of concordant / discordant pairs
  - allows partial relevance judgements (unlike Spearman’s \( p \), BPREFP)
  - allows partial orderings (unlike Spearman’s \( p \), BPREFP)
  - \( d_1, d_2 \) (clicked) \( d_3 > d_4, d_2 > d_4 \)
  - \( d_4 \) (non-clicked) \( d_1 > d_3, d_2 > d_3 \)
  - \( a_{11}, a_{12}, a_{21}, a_{22} \)
  - \( P = 2 \)
  - \( Q = 1 \)
  - \( \frac{P - Q}{P + Q} = 0.33 \)

- Binary preference
  - \( R \): relevant, \( N \): non-relevant above \( d \)
  - allows partial judgements, assumes full ordering
  - \( BPREFP = \frac{1}{R} \sum_{i=1}^{R} (1 - \frac{N_i}{R}) \)
  - penalty: \( 0.5 \)

Testing significance: Sign test

<table>
<thead>
<tr>
<th>Query</th>
<th>Q1</th>
<th>Q2</th>
<th>Q3</th>
<th>Q4</th>
<th>Q5</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>System A</td>
<td>0.61</td>
<td>0.52</td>
<td>0.12</td>
<td>0.73</td>
<td>0.22</td>
<td>0.44</td>
</tr>
<tr>
<td>System B</td>
<td>0.32</td>
<td>0.95</td>
<td>0.13</td>
<td>0.32</td>
<td>0.12</td>
<td>0.29</td>
</tr>
</tbody>
</table>

- No assumptions about differences, just (+) or (-)
- \( H_0: P(+) = P(-) = 0.5 \) five “coin tosses”
- odds of observing 3 or more (+): \( \sum_{i=3}^{5} \binom{5}{i} (0.5)^i (0.5)^{5-i} \) 35%
- 50% likelihood by pure chance → cant reject \( H_0 \)
- A and B are not significantly different

Other tests: Wilcoxon signed rank test, T-test
- more sensitive, make assumptions about data

Hypotheses Testing

- \( A \) is better than \( B \) on task \( Z \) along some dimension \( W \)
- for keyword-based searches in medical databases
- Pseudo-relevance Feedback will provide better search results than Topic Modeling as measured by mean average precision of the ranked list.

- Run systems \( A, B \) on 5 queries
  - observe: \( A \) has 50% higher MAP than \( B \)
  - can I conclude \( A \) is better?

- Differences could be purely accidental
  - 5 queries enough to be confident?
  - 50% a significant difference?

Training and testing

- Search systems require parameter tuning
  - run/evaluate many times on the same data
  - tempting to report the best result (bad idea)
  - training set: find the best parameter values
  - testing: run your system (once), report

- Splitting the data
  - by query, topics → (training/testing), run over the same docs
    - different from what’s traditional in ML, avoids vocabulary bias
  - by document: docs → (training/testing), over the same topics
    - typically streaming applications: monitoring news, detecting events

Summary: evaluation

- Evaluation is key: test early, test often
- Cranfield paradigm:
  - automated evaluation based on test collection
- Evaluation measures:
  - accuracy (meanings, 96.9%)
  - set-based measures (R,P) depend on threshold
  - use recall-precision plots, MAP
- Query logs → preferences → Kendall tau / BPREFP
- Always test significance (sign test)
- Never report best of \( N \) trials
  - run only once on testing data (or report every single run)