Retrieval Experiments using Pseudo-desktop Collections

Jinyoung Kim and W. Bruce Croft

CIIR, UMass Amherst

CIKM’09
Outline

1. Introduction
2. Building Pseudo-desktop Collections
3. Retrieval Models for Desktop Search
4. Retrieval Experiments
5. Conclusions
Desktop Search

Significance
- Most common search system for personal information

Characteristics & Relevant Problems
- People mostly do ‘re-finding’
  - Known-item search
- Many document types
  - Meta-search
- Unique metadata for each type
  - Semi-structured document retrieval

This definition holds true even for the age of cloud computing!
Past Works on Desktop Search

Focuses

- User interface issues [Dumais03,06]
- Desktop-specific features [Solus06] [Cohen08]

Limitations

- Each based on different user study
- None of them performed comparative evaluation

Table: Statistics of desktop collections from previous research

<table>
<thead>
<tr>
<th>Previous Work</th>
<th>#Desktops</th>
<th>#Files</th>
<th>Query Length</th>
<th>Document Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dumais et al.</td>
<td>225</td>
<td>36182</td>
<td>1.6</td>
<td>e-mails: 80%</td>
</tr>
<tr>
<td>Chernov et al.</td>
<td>14</td>
<td>3433</td>
<td>1.7</td>
<td>e-mails: 82.7%</td>
</tr>
<tr>
<td>Cohen et al.</td>
<td>19</td>
<td>N/A</td>
<td>N/A</td>
<td>documents: 41.2%</td>
</tr>
</tbody>
</table>

Key missing piece here is a reusable collection!
Contributions

Pseudo-desktop Collections
- Suggested a new query generation method with higher validity
- Suggested a new way to evaluate the validity

Retrieval Experiments
- Compared semi-structured document retrieval methods
- Improved and analyzed the performance of PRM-S [Kim09]
Outline

1. Introduction
2. Building Pseudo-desktop Collections
3. Retrieval Models for Desktop Search
4. Retrieval Experiments
5. Conclusions
Collecting Documents

Criteria
- Reasonable size and variety
- Availability of metadata
- No privacy concern

Procedure
- Filter public email collection by person name
- Add office/web documents by web search for each person’s profile

<table>
<thead>
<tr>
<th>Type</th>
<th>Jack</th>
<th>Tom</th>
<th>Kate</th>
</tr>
</thead>
<tbody>
<tr>
<td>email</td>
<td>6067 (555)</td>
<td>6930 (558)</td>
<td>1669 (935)</td>
</tr>
<tr>
<td>html</td>
<td>953 (3554)</td>
<td>950 (3098)</td>
<td>957 (3995)</td>
</tr>
<tr>
<td>pdf</td>
<td>1025 (8024)</td>
<td>1008 (8699)</td>
<td>1004 (10278)</td>
</tr>
<tr>
<td>doc</td>
<td>938 (6394)</td>
<td>984 (7374)</td>
<td>940 (7828)</td>
</tr>
<tr>
<td>ppt</td>
<td>905 (1808)</td>
<td>911 (1801)</td>
<td>729 (1859)</td>
</tr>
</tbody>
</table>
Generating Queries

Assumption
- Users take query terms from the target document [Azzopardi07]

Procedure
- Choose a target document
- Choose the **extent** of term selection
- Choose **term** from the extent based on some distribution

Parameters
- Choice of **extent**: Document vs. Field
- Choice of **term**: Uniform / TF / IDF / TF*IDF
Validating Generating Queries

By Comparison with Manual Queries

- Compare Query-terms (Predictive Validity)
  - Generation probability $P_{\text{term}}(Q)$ of the manual query $Q$
    
    $$P_{\text{term}}(Q) = \prod_{q_i \in Q} P_{\text{term}}(q_i)$$  

- Compare the Distribution of Retrieval Scores (Replicative Validity)
  - Two-sided Kolmogorov-Smirnov test [Azzopardi07]

Evaluation Settings:

- W3C mailing list (200k docs)
  - Manual: 150 TREC known-item queries
  - Generated: 1000 queries for each method
Validating Generating Queries

By Comparison with Manual Queries

- Compare Query-terms (Predictive Validity)
  - Generation probability $P_{\text{term}}(Q)$ of the manual query $Q$
    \[ P_{\text{term}}(Q) = \prod_{q_i \in Q} P_{\text{term}}(q_i) \]  

- Compare the Distribution of Retrieval Scores (Replicative Validity)
  - Two-sided Kolmogorov-Smirnov test [Azzopardi07]

Evaluation Settings:

- W3C mailing list (200k docs)
  - Manual: 150 TREC known-item queries
  - Generated: 1000 queries for each method
Evaluation of Predictive Validity

Table: Sum of generation probabilities for different generation methods

<table>
<thead>
<tr>
<th>Extent: Document</th>
<th>Extent: Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_{\text{term}}$</td>
<td>$\sum \log P_{\text{term}}(Q)$</td>
</tr>
<tr>
<td>Uniform</td>
<td>-26.457</td>
</tr>
<tr>
<td>TF</td>
<td>-22.782</td>
</tr>
<tr>
<td>IDF</td>
<td>-21.876</td>
</tr>
<tr>
<td>TF*IDF</td>
<td>-18.269</td>
</tr>
</tbody>
</table>

- Field-based extent selection results in higher validity
Evaluation of Replicative Validity

Table: p-values of KS-test against the score distribution of manual queries

<table>
<thead>
<tr>
<th>Extent</th>
<th>$P_{term}$</th>
<th>DLM</th>
<th>PRM-S</th>
<th>PRM-D</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Document</td>
<td>Uniform</td>
<td>0.003</td>
<td>0.000</td>
<td>0.041</td>
<td>0.015</td>
</tr>
<tr>
<td></td>
<td>TF</td>
<td>0.090</td>
<td>0.000</td>
<td>0.005</td>
<td>0.032</td>
</tr>
<tr>
<td></td>
<td>IDF</td>
<td>0.000</td>
<td>0.023</td>
<td>0.000</td>
<td>0.008</td>
</tr>
<tr>
<td></td>
<td>TF*IDF</td>
<td>0.000</td>
<td>0.160</td>
<td>0.000</td>
<td>0.053</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.023</td>
<td>0.046</td>
<td>0.012</td>
<td>0.027</td>
</tr>
<tr>
<td>Field</td>
<td>Uniform</td>
<td>0.085</td>
<td>0.323</td>
<td>0.276</td>
<td>0.228</td>
</tr>
<tr>
<td></td>
<td>TF</td>
<td>0.105</td>
<td>0.667</td>
<td>0.570</td>
<td>0.447</td>
</tr>
<tr>
<td></td>
<td>IDF</td>
<td>0.068</td>
<td>0.013</td>
<td>0.008</td>
<td>0.030</td>
</tr>
<tr>
<td></td>
<td>TF*IDF</td>
<td>0.284</td>
<td>0.021</td>
<td>0.022</td>
<td>0.109</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.136</td>
<td>0.256</td>
<td>0.219</td>
<td>0.204</td>
</tr>
</tbody>
</table>

Field-based extent selection results in higher validity
Retrieval Models

Notations:
- Query $Q = (q_1, \ldots, q_m)$
- Collection $C$ with fields $(F_1, \ldots, F_n)$
- Each document $d$ with fields $(f_1, \ldots, f_n)$

Retrieval Models:
- Document Query Likelihood (DLM)
- Mixture of Field Language Models (MFLM) [Ogilvie03]

\[
P(Q|d) = \prod_{i=1}^{m} \sum_{j=1}^{n} w_j P_{QL}(q_i|f_j)
\]  

BM25F [Robertson04]

\[
weight(q_i, d) = \sum_{f_j \in d} \frac{w_j \times tf(q_i, f_j)}{(1 - b_j) + b_j \times \frac{|f_j|}{|F_j|}}
\]
Retrieval Models

Notations:
- Query \( Q = (q_1, ..., q_m) \)
- Collection \( C \) with fields \( (F_1, ..., F_n) \)
- Each document \( d \) with fields \( (f_1, ..., f_n) \)

Retrieval Models:
- Document Query Likelihood (DLM)
- Mixture of Field Language Models (MFLM) [Ogilvie03]

\[
P(Q|d) = \prod_{i=1}^{m} \sum_{j=1}^{n} w_j P_{QL}(q_i|f_j) \tag{2}
\]

- BM25F [Robertson04]

\[
weight(q_i, d) = \sum_{f_j \in d} \frac{w_j \times tf(q_i, f_j)}{(1 - b_j) + b_j \times \frac{|f_j|}{|F_j|}} \tag{3}
\]
Retrieval Models (cont.)

PRM-S (Probabilistic Retrieval Model for Semistructured Data \[\text{[Kim09]}\])
- Probabilistically map each query word with document field
  - e.g. Meg (cast:0.8) Ryan (cast:0.7) Romance (genre:0.6)
- Combine field LMs based on mapping probability

\[
P(Q|d) = \prod_{i=1}^{m} \sum_{j=1}^{n} P_M(F_j|q_i)P_{QL}(q_i|f_j)
\]  

Improving Field-level Score Estimation of PRM-S
- Mixture of DLM and PRM-S (PRM-D)
- 2-Stage Dirichlet Smoothing (PRM-S2) \[\text{[Zhao08]}\]

2-stage smoothing provides better control at the cost of more parameters
PRM-S (Probabilistic Retrieval Model for Semistructured Data [Kim09])
- Probabilistically map each query word with document field
  - e.g. Meg (cast:0.8) Ryan (cast:0.7) Romance (genre:0.6)
- Combine field LMs based on mapping probability

\[
P(Q|d) = \prod_{i=1}^{m} \sum_{j=1}^{n} P_M(F_j|q_i)P_{QL}(q_i|f_j)
\]  
(4)

Improving Field-level Score Estimation of PRM-S
- Mixture of DLM and PRM-S (PRM-D)
- 2-Stage Dirichlet Smoothing (PRM-S2) [Zhao08]

2-stage smoothing provides better control at the cost of more parameters
Big Picture

Queries

- Manual Queries
- Generated Queries
- Validate

Documents

- E-mails
  - sender
  - receiver
  - title
  - content
  - attachment

- Webpages
  - title
  - content
  - URL

- Documents
  - title
  - content
  - author

Retrieval Models

- DLM
- BM25F
- MFLM
- PRM-S
- PRM-D
- PRM-S2
Outline

1. Introduction
2. Building Pseudo-desktop Collections
3. Retrieval Models for Desktop Search
4. Retrieval Experiments
5. Conclusions
Experimental Setting

Collections

- W3C mailing list
  - TREC queries (25 train / 125 test)
- Pseudo-desktop
  - Manual queries (50)
    - Collected by showing people documents and asking queries
  - Generated queries (100)
    - Field-based method, with average length 2

<table>
<thead>
<tr>
<th>Type</th>
<th>Jack</th>
<th>Tom</th>
<th>Kate</th>
</tr>
</thead>
<tbody>
<tr>
<td>email</td>
<td>6067 (555)</td>
<td>6930 (558)</td>
<td>1669 (935)</td>
</tr>
<tr>
<td>html</td>
<td>953 (3554)</td>
<td>950 (3098)</td>
<td>957 (3995)</td>
</tr>
<tr>
<td>pdf</td>
<td>1025 (8024)</td>
<td>1008 (8699)</td>
<td>1004 (10278)</td>
</tr>
<tr>
<td>doc</td>
<td>938 (6394)</td>
<td>984 (7374)</td>
<td>940 (7828)</td>
</tr>
<tr>
<td>ppt</td>
<td>905 (1808)</td>
<td>911 (1801)</td>
<td>729 (1859)</td>
</tr>
</tbody>
</table>
Query Examples

<table>
<thead>
<tr>
<th>TREC Manual Queries</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preliminary Report from the WSDL Attributes Task Force</td>
</tr>
<tr>
<td>XML DSig 99</td>
</tr>
<tr>
<td>MobiQuitous 2004 latest deadlines</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Pseudo-desktop Manual Queries</td>
</tr>
<tr>
<td>Martyn Jan OCLC</td>
</tr>
<tr>
<td>proof checking</td>
</tr>
<tr>
<td>syntax for RDF</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Pseudo-desktop Generated Queries</td>
</tr>
<tr>
<td>jose 03 kahan</td>
</tr>
<tr>
<td>other ua</td>
</tr>
<tr>
<td>amaya 48</td>
</tr>
</tbody>
</table>

Other details

- All parameters tuned using TREC training queries
- Reciprocal Rank was used as the evaluation measure
Result - Manual Queries

Table: Retrieval performance for TREC email collection

<table>
<thead>
<tr>
<th>Collection</th>
<th>DLM</th>
<th>MFLM</th>
<th>BM25F</th>
<th>PRM-S</th>
<th>PRM-D</th>
<th>PRM-S2</th>
</tr>
</thead>
<tbody>
<tr>
<td>TREC</td>
<td>0.538</td>
<td>0.559</td>
<td>0.594</td>
<td>0.617</td>
<td>0.619</td>
<td><strong>0.630</strong></td>
</tr>
</tbody>
</table>

- **PRM-S < PRM-D < PRM-S2**
- **PRM-S2** outperforms the best TREC submission (0.621)

Table: Retrieval performance for pseudo-desktop email collections

<table>
<thead>
<tr>
<th>Collection</th>
<th>DLM</th>
<th>MFLM</th>
<th>BM25F</th>
<th>PRM-S</th>
<th>PRM-D</th>
<th>PRM-S2</th>
</tr>
</thead>
<tbody>
<tr>
<td>PD-Jack</td>
<td>0.378</td>
<td>0.235</td>
<td>0.229</td>
<td>0.334</td>
<td><strong>0.389</strong></td>
<td>0.356</td>
</tr>
<tr>
<td>PD-Tom</td>
<td>0.403</td>
<td>0.312</td>
<td>0.311</td>
<td>0.422</td>
<td><strong>0.457</strong></td>
<td>0.438</td>
</tr>
<tr>
<td>PD-Kate</td>
<td><strong>0.482</strong></td>
<td>0.307</td>
<td>0.401</td>
<td>0.413</td>
<td>0.463</td>
<td>0.455</td>
</tr>
</tbody>
</table>

- **PRM-S < PRM-S2 < PRM-D**
Result - Generated Queries in Pseudo-desktop

<table>
<thead>
<tr>
<th>Type</th>
<th>User</th>
<th>DLM</th>
<th>PRM-S</th>
<th>PRM-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>email</td>
<td>Jack</td>
<td>0.213</td>
<td>0.285</td>
<td>0.275</td>
</tr>
<tr>
<td></td>
<td>Tom</td>
<td>0.197</td>
<td>0.244</td>
<td>0.243</td>
</tr>
<tr>
<td></td>
<td>Kate</td>
<td>0.329</td>
<td>0.438</td>
<td>0.413</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.246</td>
<td>0.323</td>
<td>0.310</td>
</tr>
<tr>
<td>html</td>
<td>Jack</td>
<td>0.418</td>
<td>0.428</td>
<td>0.440</td>
</tr>
<tr>
<td></td>
<td>Tom</td>
<td>0.428</td>
<td>0.381</td>
<td>0.409</td>
</tr>
<tr>
<td></td>
<td>Kate</td>
<td>0.422</td>
<td>0.377</td>
<td>0.410</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.423</td>
<td>0.395</td>
<td>0.419</td>
</tr>
<tr>
<td>pdf</td>
<td>Jack</td>
<td>0.417</td>
<td>0.378</td>
<td>0.410</td>
</tr>
<tr>
<td></td>
<td>Tom</td>
<td>0.411</td>
<td>0.359</td>
<td>0.392</td>
</tr>
<tr>
<td></td>
<td>Kate</td>
<td>0.408</td>
<td>0.420</td>
<td>0.453</td>
</tr>
<tr>
<td>Average</td>
<td></td>
<td>0.412</td>
<td>0.385</td>
<td>0.419</td>
</tr>
</tbody>
</table>

Result
- DLM is better for html, while PRM-S is better for email
- PRM-D shows consistently high performance
Analysis: What Makes PRM-S Better?

The accuracy of mapping probability

The field from which query terms are chosen

<table>
<thead>
<tr>
<th>Type</th>
<th>Field</th>
<th>DLM</th>
<th>PRM-S</th>
<th>PRM-D</th>
</tr>
</thead>
<tbody>
<tr>
<td>email</td>
<td>title</td>
<td>0.251</td>
<td>0.389</td>
<td>0.342</td>
</tr>
<tr>
<td></td>
<td>content</td>
<td>0.339</td>
<td>0.267</td>
<td>0.327</td>
</tr>
<tr>
<td>html</td>
<td>title</td>
<td>0.461</td>
<td>0.533</td>
<td>0.547</td>
</tr>
<tr>
<td></td>
<td>content</td>
<td>0.514</td>
<td>0.278</td>
<td>0.339</td>
</tr>
</tbody>
</table>
Outline

1. Introduction
2. Building Pseudo-desktop Collections
3. Retrieval Models for Desktop Search
4. Retrieval Experiments
5. Conclusions
Conclusions & Future Works

Conclusions

- Field-based query generation is more valid
  - Insight into querying behavior of known-item search
- PRM-S is effective and PRM-D / PRM-S2 improves it further
  - Different collection requires different retrieval model
- Generated queries are useful to analyze the performance
  - Enables completely controlled experiments

Future Works

- Pseudo-desktop
  - More realistic query generation and evaluation method
  - Further validation by user study
- Retrieval Model
  - Merging the results from each sub-collection
Conclusions & Future Works

Conclusions

- Field-based query generation is more valid
  - Insight into querying behavior of known-item search
- PRM-S is effective and PRM-D / PRM-S2 improves it further
  - Different collection requires different retrieval model
- Generated queries are useful to analyze the performance
  - Enables completely controlled experiments

Future Works

- Pseudo-desktop
  - More realistic query generation and evaluation method
  - Further validation by user study
- Retrieval Model
  - Merging the results from each sub-collection