Query-biased Partitioning for Selective Search
Outline

• **Background – Selective Search**
• Proposed Methods
  • Query-driven clustering initialization
  • Query-biased similarity metric
• Experiments & Analysis
• Conclusions
Selective Search

• Traditional Distributed Search
  • A document corpus => small random shards
  • Searched all shards in parallel
  • Merge results
  • Exhaustive Search

• Selective Search
  • A document corpus => topical shards.
  • The query is run against only a few shards.
  • Goal: same search accuracy as exhaustive search, but much faster
Selective Search Pipeline

1. Sample
2. Clustering K-means with random seeds
3. Project Project remaining documents

Corpus \rightarrow PARTITIONING \rightarrow Topical Shards

RESOURCE SELECTION
- Small-Document: Rank-S (KTCH, CIKM’12)
- Big-Document: Taily (ADH, SIGIR’13)
- Supervised: LeToR (DKC, submitted)
Error Analysis (DKC, SIGIR’15)

1. System Variance: from the clustering process.
2. Lower search effectiveness (MAP) than exhaustive if using a real resource selection algorithm.

Rank-S: Real Resource Selection  RBR: Oracle Resource Selection
Why does resource selection select the wrong shards?

- **Problem:** The topics generated by the content-based partitioning do not match the topics searched by the users.

- **Example:** Query **Obama family tree**

![Diagram of Obama family tree]

- **People Names’ shard:** Ann, Ben, Charles, Peter, Michelle, Jonathan...
- **U.S. Politics’ shard:** Immigration, federal, president, police, Obama, Washington...
Content-based Partitioning: Topic Mismatch

• How to group together documents that satisfy the same user intent?
  • Query Logs!
• This work investigates aligning document partitioning with topics from the query logs.
Outline

• Background
• Proposed Methods
  • Query-driven clustering initialization
  • Query-biased similarity metric
• Experiments & Analysis
• Conclusions
QInit: Query-driven Clustering Initialization

Randomly Sample \( K \) Topics
Documents as Seeds

Document Clustering
QInit: Query-driven Clustering Initialization

• Start the document clustering process with query-log topics.
Term Weighting in QInit

\[ w_q(t) = \log(tf_{t,Q} + 1) \times \log\left(\frac{|D|}{df_{t,D}} + 1\right) \]

- **Query-log TF**
  - \( tf_{t,Q} \): Term frequency in the query log
  - Promotes the importance of terms frequently used by users in search
  - Log function: term distribution in the query log is very skewed

- **Collection IDF**
  - \( df_{t,D} \): # of documents that contain the term in the collection.
  - Demotes terms that are too common in the corpus
Examples of shard generated with QInit

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Top weighted Terms in Initial Seed</th>
<th>Relevant Queries</th>
</tr>
</thead>
<tbody>
<tr>
<td>CW09-B</td>
<td>wine, tea, coffee, smoking, alcohol, drink</td>
<td>Starbucks, quit smoking</td>
</tr>
<tr>
<td></td>
<td>animal, cock, bird, wild, egg, cat</td>
<td>dinosaurs, Arizona game and fish, moths,...</td>
</tr>
<tr>
<td>Gov2</td>
<td>tax, revenue, loans, business, bank, taxation</td>
<td>reverse mortgages, timeshare resales, ...</td>
</tr>
<tr>
<td></td>
<td>diabetes, autism, obesity, arthritis, hypertension, celiac</td>
<td>aspirin cancer prevention, embryonic stem cells, ...</td>
</tr>
</tbody>
</table>
Outline

• Background
• Proposed Methods
  • Query-driven clustering initialization
  • Query-biased similarity metric
• Experiments & Analysis
• Conclusions
QKLD: Query-biased Similarity Metric

• Bias the clustering (partitioning) towards important query log terms.

SNIPPET 1: Family of Obama
Barack Obama was raised by his mother, Stanley Ann Dunham, called Ann, and grandparents Madelyn and Stanley Dunham.

SNIPPET 2: Lena Dunham
Dunham was born in New York City. Her father, Carroll Dunham, is a painter, and her mother, Laurie Simmons, is an artist and photographer.

SNIPPET 3: Obama’s Education Law
President Barack Obama signed into law legislation that replaces the landmark No Child Left Behind education law of 2002.
QKLD: Query-biased Similarity Metric

• Bias the clustering (partitioning) towards important query log terms.

SNIPPET 1: Family of Obama
Barack Obama was raised by his mother, Stanley Ann Dunham, called Ann, and grandparents Madelyn and Stanley Dunham.

SNIPPET 2: Lena Dunham
Dunham was born in New York City. Her father, Carroll Dunham, is a painter, and her mother, Laurie Simmons, is an artist and photographer.

SNIPPET 3: Obama’s Education Law
President Barack Obama signed into law legislation that replaces the landmark No Child Left Behind education law of 2002.
QKLD: Query-biased Similarity Metric

• Previous state-of-art similarity metric in selective search

\[ \text{sim}_{KLD}(d, c) = \sum_{t \in d \cap c} \text{s}_{KLD}(\vec{d}_t, \vec{c}_t) \]

\[ \text{s}_{KLD}(\vec{d}_t, \vec{c}_t) = p_c(t) \log \frac{p_d(t)}{\lambda p_B(t)} + p_d(t) \log \frac{p_c(t)}{\lambda p_B(t)} \]

• Re-weight each term’s similarity contribution by their importance in the query log

\[ \text{sim}_{QKLD}(\vec{d}, \vec{c}) = \sum_{t \in d \cap c} (w_q(t) + b) \times \text{s}_{KLD}(\vec{d}_t, \vec{c}_t) \]

* Previous state-of-art similarity metric in selective search
* Re-weight each term’s similarity contribution by their importance in the query log

QTF * IDF:

• \( b > 0 \), smoothing parameter. Balance query log and document content
Outline

- Background
- Proposed Methods
  - Query-driven clustering initialization
  - Query-biased similarity metric
- Experiments & Analysis
- Conclusions
Datasets

<table>
<thead>
<tr>
<th>Document Collection</th>
<th>ClueWeb09-B</th>
<th>Gov2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Documents</td>
<td>50,220K</td>
<td>25,205K</td>
</tr>
<tr>
<td>Test Query Set (with manual relevance judgements)</td>
<td>200 queries (TREC)</td>
<td>150 queries (TREC)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Query Logs</th>
<th>AOL-ALL</th>
<th>AOL-Gov2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Queries</td>
<td>24,189,556</td>
<td>540,285</td>
</tr>
<tr>
<td>Queries after filtering</td>
<td>13,950,463</td>
<td>403,610</td>
</tr>
<tr>
<td>%Terms (w/o numbers)</td>
<td>978,714</td>
<td>69,482</td>
</tr>
<tr>
<td>%Terms after filtering</td>
<td>80,963</td>
<td>14,018</td>
</tr>
</tbody>
</table>

- Word Embeddings: 300-d Google word2vec trained on the corpora.
- Gov2 results are not shown in this presentation. Similar to ClueWeb09-B.
Baseline & Proposed Methods

• Partitioning methods:
  • KLD-Rand (baseline)
  • QKLD-Rand
  • KLD-QInit
  • QKLD-Qinit

• K (Number of clusters):
  • CW09-B: 100, Gov2: 150
  • Split big shards with another level of clustering

• Partition 10 times -> 10 different system instances
  • rule out random effects
  • evaluate system variance
Experiments

• **Effectiveness**: How does our method affect the clustering? Can it improve search effectiveness?

• **Robustness**: Is the method robust to query logs?

• **Efficiency**: Does it change the efficiency of the system?
Experiment 1: Clustering Analysis

• Are the query’s relevant documents concentrated in a few shards?
  • Easier for resource selection algorithm to find the right shard
  • Higher recall with fewer shards searched

• Metric: coverage

• Coverage of query q:
  • sorting shards by the number of relevance documents they contain
  • The percentage of relevance documents covered by the first t% shards

\[
\text{coverage}_t(q) = \frac{\sum_{i=1}^{\text{floor}(N \times t\%)} R_{s_i}^q}{R^q}
\]

• Coverage of the query set:

\[
\text{coverage}_t = \frac{\sum_{q=1}^{\mid Q \mid} \text{coverage}_t(q)}{\mid Q \mid}
\]
Experiment 1: Clustering Analysis (Cont.)

* indicates statistically significant difference with KLD-Rand

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>Percentage of Shards (t)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1%</td>
</tr>
<tr>
<td>CW09-B</td>
<td>KLD-Rand</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>KLD-QInit</td>
<td>0.60</td>
</tr>
<tr>
<td></td>
<td>QKLD-Rand</td>
<td>0.65*</td>
</tr>
<tr>
<td></td>
<td>QKLD-QInit</td>
<td><strong>0.67</strong>*</td>
</tr>
</tbody>
</table>

- Similarity metric: QKLD > KLD
- Initialization: QInit > Rand when combined with QKLD
- Best: QKLD-QInit
Experiment 2: Search Effectiveness

• CW09-B Results:
  • *: statistically significant difference with KLD-Rand; **: statistically significant difference with QKLD-Rand

<table>
<thead>
<tr>
<th>Metric</th>
<th>Mean</th>
<th>Standard Deviation (*10^3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Exhaustive</td>
<td>KLD-Rand</td>
</tr>
<tr>
<td>P@10</td>
<td>0.253</td>
<td>0.275</td>
</tr>
<tr>
<td>NDCG@100</td>
<td>0.286</td>
<td>0.254</td>
</tr>
<tr>
<td>MAP@1000</td>
<td>0.186</td>
<td>0.155</td>
</tr>
</tbody>
</table>
Experiment 2: Search Effectiveness (Cont.)

- Effects on Recall and Precision

Relative gains over baseline at different document rankings

<table>
<thead>
<tr>
<th>Gain of NDCG@Rank</th>
<th>CW09-B</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>QKLD -Rand</td>
</tr>
<tr>
<td>10</td>
<td>3.77%</td>
</tr>
<tr>
<td>30</td>
<td>5.49%</td>
</tr>
<tr>
<td>100</td>
<td>7.70%</td>
</tr>
<tr>
<td>500</td>
<td>9.44%</td>
</tr>
<tr>
<td>1000</td>
<td>9.87%</td>
</tr>
</tbody>
</table>

- Proposed methods improved recall

- Selective search rarely hurt Precision, sometimes even better
  - Filtering out false-positives

- Recall is harder to improve
  - Searching fewer shards will miss relevant documents in other shards
  - Important in re-ranking schema
Experiment 3: Search Robustness
Query Log Influences

• Robustness: Does temporal gaps between training queries and testing queries affect the proposed methods?
Experiment 3: Search Robustness Query Log Influences (Cont.)

• Temporal Mismatch:
  • Training: AOL query logs (2006)
  • Testing: TREC queries (Gov2: 2004-2006; CW09-B: 2009-2012)

• Compare 2 temporal conditions

<table>
<thead>
<tr>
<th>Condition</th>
<th>Training query log</th>
<th>Testing query set</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unaligned</td>
<td>AOL, first 2 months</td>
<td>TREC</td>
</tr>
<tr>
<td>Aligned</td>
<td>AOL, first 2 months</td>
<td>AOL, last 1 months</td>
</tr>
</tbody>
</table>

• Evaluation: overlap between exhaustive search and selective search (the high the better)

\[
overlap_k = \frac{|D_{k}^{exh} \cap D_{k}^{sel}|}{k}
\]
Experiment 3: Search Robustness Query Log Influences (Cont.)

- Unaligned in general had higher overlap
  - TREC queries, less noisy

- Same trend:
  - QKLD-QInit > QKLD-Rand > KLD-Rand

- Same relative gain over baseline: difference is not statistically significant
- Not Sensitive to the temporal mismatch.
Experiment 4: Search Efficiency

• Efficiency: Whether query-biased partitioning changes selective search efficiency?
**Experiment 4: Search Efficiency**

<table>
<thead>
<tr>
<th></th>
<th>CW09-B</th>
<th>Gov2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$C_{RES}$</td>
<td>$C_{LAT}$</td>
</tr>
<tr>
<td>Exhaustive Search</td>
<td>5.24</td>
<td>0.33</td>
</tr>
<tr>
<td>KLD-Rand</td>
<td>0.53</td>
<td>0.24</td>
</tr>
<tr>
<td>QKLD-Rand</td>
<td>0.52</td>
<td>0.24</td>
</tr>
<tr>
<td>QKLD-QInit</td>
<td>0.52</td>
<td>0.23</td>
</tr>
</tbody>
</table>

**Metrics:**

- Total resource usage $C_{RES}$:
  \[ C_{RES}(q) = |D_{CSI}^q| + \sum_{i=1}^{T} |D_{S_i}^q| \]  
- Query Latency $C_{LAT}$:
  \[ C_{LAT}(q) = |D_{CSI}^q| + \max_{i=1}^{T_q} |D_{S_i}^q| \]

**Query-biased partitioning does NOT increase search cost.**
Outline

• Background
• Proposed Methods
  • Query-driven clustering initialization
  • Query-biased similarity metric
• Experiments & Analysis
• Conclusion
Conclusion

• Proposed a query-biased partitioning strategy for selective search
  • Previous clustering: un-supervised.
  • Use query-logs as a weak supervision for the clustering.

• Evaluation & Analysis:
  • Improves search effectiveness and reduce variance:
    • Concentrates relevant documents together
  • Not sensitive to temporal difference between training & testing queries
    • Queries change over time, but the general topics are stable.
    • Do not need a perfect query log!
Thank You!

Q&A