Wikipedia Pages as Entry Points for Book Search

Marijn Koolen\textsuperscript{1,2}, Gabriella Kazai\textsuperscript{1}, Nick Craswell\textsuperscript{1}
\textsuperscript{1} Microsoft Research, Cambridge, UK
\textsuperscript{2} University of Amsterdam, the Netherlands

WSDM’09
Barcelona, 10 February, 2009
Outline

- Introduction
- Wikipedia Coverage
  - Wikipedia coverage of search topics
  - Wikipedia coverage of book topics
- Wikipedia as intermediary
  - Query Expansion
  - Topical Closeness
- Experiments & results
- Conclusion
Introduction

- Thanks to mass book digitisation efforts, large book collections now available online

- Books online thanks to mass-digitisation projects
  - Million Books Project, Google Book Search
Introduction

• Thanks to mass book digitisation efforts, large book collections now available online

• Books online thanks to mass-digitisation projects
  ✭ Million Books Project, Google Book Search

• Significant repository of (untapped) knowledge:
  "For hundreds of years books have been the repositories for the worlds most trusted, authoritative knowledge." – Cliff Guren, Live Search

• Access through book search (content or metadata)
Our Approach

- Use Wikipedia as an intermediary between users and books:
  - Our knowledge of the world is (for a significant part) stored in books
  - Encyclopedias summarise this world knowledge:
    "...the purpose of an encyclopedia is to collect knowledge disseminated around the globe..." — Diderot
Our Approach

- Use Wikipedia as an intermediary between users and books:
  - Our knowledge of the world is (for a significant part) stored in books
  - Encyclopedias summarise this world knowledge:
    "...the purpose of an encyclopedia is to collect knowledge disseminated around the globe..." – Diderot
Research Questions

- Many search topics have an entry in Wikipedia:
  - Can we automatically extract useful search terms from related Wikipedia pages to improve retrieval effectiveness of a book search system?
Research Questions

• Many search topics have an entry in Wikipedia:
  ✴ Can we automatically extract useful search terms from related Wikipedia pages to improve retrieval effectiveness of a book search system?

• Many book topics have corresponding Wiki pages as well

• Wikipedia has many links between related topics:
Research Questions

• Many search topics have an entry in Wikipedia:
  ✴ Can we automatically extract useful search terms from related Wikipedia pages to improve retrieval effectiveness of a book search system?

• Many book topics have corresponding Wiki pages as well

• Wikipedia has many links between related topics:
  ✴ Is the link distance between search topics and book topics in Wikipedia related to relevance and can we use this to improve retrieval effectiveness?
Wikipedia Coverage

Our approach relies on two assumptions:

1. Wikipedia covers many user search topics
2. Wikipedia covers the topics found in books
Wikipedia Coverage

- Our approach relies on two assumptions:
  1. Wikipedia covers many user search topics
  2. Wikipedia covers the topics found in books

- Two intuitions support these assumptions:
  1. Wikipedia is collectively written, on topics of interest
  2. Encyclopedias collect and summarise human knowledge

- Do we have more than just intuitions?
Wikipedia Coverage of Search Topics

- Do Wikipedia entries cover topics searched for by web users?
Wikipedia Coverage of Search Topics

- Do Wikipedia entries cover topics searched for by web users?
  - We compare queries from a Web log with Wiki page titles
  - 38.6% of 5.76 billion queries match Wikipedia page title
Wikipedia Coverage of Search Topics

- Do Wikipedia entries cover topics searched for by web users?
  - We compare queries from a Web log with Wiki page titles
  - 38.6% of 5.76 billion queries match Wikipedia page title

Query frequency distribution

(a) All  (b) Matching Wiki title  (c) All & Matching Wiki title
Wikipedia coverage of book topics

- Source: Halavais and Lackaff (2008)
  - Topics in published books and in sample of Wikipedia pages
Outline

- Introduction – Book Search

- Wikipedia Coverage
  - Wikipedia coverage of search topics
  - Wikipedia coverage of book topics

- Wikipedia as intermediary
  - Query Expansion
  - Topical Closeness

- Experiments & results

- Conclusion
Query Expansion

- Use Wiki page matching the query as rich topical description to draw terms from
  - Using INEX Book Track corpus (42,095 books)
  - Assumption: matching Wiki page is relevant
Query Expansion

• Use Wiki page matching the query as rich topical description to draw terms from
  ★ Using INEX Book Track corpus (42,095 books)
  ★ Assumption: matching Wiki page is relevant

• How to select terms?
  ★ Terms from first paragraph
  ★ Terms from anchor text
  ★ Based on $tf.idf$ scores
Query Expansion

- Use Wiki page matching the query as rich topical description to draw terms from
  - Using INEX Book Track corpus (42,095 books)
  - Assumption: matching Wiki page is relevant

- How to select terms?
  - Terms from first paragraph
  - Terms from anchor text
  - Based on $tf.idf$ scores

- Initial experiments show $tf.idf$ works best
  - weight original query $N$ times as much as the $N$ added terms
Topical Closeness

• Wikipedia covers topics found in books (exit points):
  ✫ Users can traverse the link graph to related topics (→ books)
Topical Closeness

- Wikipedia covers topics found in books (exit points):
  - Users can traverse the link graph to related topics (→ books)
Topical Closeness

- Wikipedia covers topics found in books (exit points):
  - Users can traverse the link graph to related topics (→ books)

- Is link distance between search topics and book topics related to relevance?
Modelling Topical Closeness

• How can we associate books with topics in Wikipedia?
  1. use book references on Wiki pages
  2. use document similarity: book as query, rank Wiki pages
Modelling Topical Closeness

- How can we associate books with topics in Wikipedia?
  1. use book references on Wiki pages
  2. use document similarity: book as query, rank Wiki pages

- How can we measure the link distance between two topics in Wikipedia?
  - Use random walk to compute closeness scores
Book References on Wikipedia Pages

- Many Wiki pages have references to books:
  - Referenced books are relevant to the topic (?)
Book References on Wikipedia Pages

- Many Wiki pages have references to books:
  - Referenced books are relevant to the topic (?)

- Small overlap with books in INEX collection (1,362 out of 42,095):
  - Most books cited by Wiki pages published after 1970
  - Most books in INEX corpus published up to 1930
Document Similarities

- Match each book in collection against Wiki page(s) based on document similarity
Document Similarities

• Match each book in collection against Wiki page(s) based on document similarity

• We indexed Wikipedia and used books as queries
  ◆ search engine can’t handle whole book as query
  ◆ use the top 100 terms based on $tf.idf$ weights
Document Similarities

- Match each book in collection against Wiki page(s) based on document similarity

- We indexed Wikipedia and used books as queries
  - search engine can’t handle whole book as query
  - use the top 100 terms based on $tf.idf$ weights

- Associate book with top $N$ Wiki pages
  - books can have multiple topics
  - We experiment with $N = 1, 3, 5$

- All books in the INEX Book corpus can be matched
Computing Closeness

- We have linked both queries and books to Wiki pages.

- How to measure topical “closeness” in a graph?
Computing Closeness

• We have linked both queries and books to Wiki pages.

• How to measure topical “closeness” in a graph?

• Use random walk model:
  ✴ starting from page matching the query
  ✴ obtain closeness scores for all books
Computing Closeness

• We have linked both queries and books to Wiki pages.

• How to measure topical “closeness” in a graph?

• Use random walk model:
  ★ starting from page matching the query
  ★ obtain closeness scores for all books

• Are books found closer to the query topic more likely to be relevant than books further away from it?
Closeness and Probability of Relevance

- We can compute the probability of relevance (PoR) over topical closeness
Closeness and Probability of Relevance

- We can compute the probability of relevance (PoR) over topical closeness.

- Each score represents closeness between a query and a book:
  - sort scores and bin per 10,000 scores
Closeness and Probability of Relevance

- We can compute the probability of relevance (PoR) over topical closeness

- Each score represents closeness between a query and a book
  - sort scores and bin per 10,000 scores
  - count scores representing a query and a book relevant to that query
Closeness and Probability of Relevance

- We can compute the probability of relevance (PoR) over topical closeness

- Each score represents closeness between a query and a book
  - sort scores and bin per 10,000 scores
  - count scores representing a query and a book relevant to that query
  - PoR is the ratio of relevant scores in each bin
Closeness and Probability of Relevance

- We can compute the probability of relevance (PoR) over topical closeness

- Each score represents closeness between a query and a book
  - sort scores and bin per 10,000 scores
  - count scores representing a query and a book relevant to that query
  - PoR is the ratio of relevant scores in each bin

- If closeness is related to relevance, we expect PoR to go up with increasing closeness score
Closeness and Relevance

References

Doc. sim.
Closeness and Relevance

We see:

- Only at higher scores (> 0.0001) do we see a rising trend
- Document similarity seems the more stable indicator
Outline

- Introduction

- Wikipedia Coverage
  - Wikipedia coverage of search topics
  - Wikipedia coverage of book topics

- Wikipedia as Intermediary
  - Query expansion
  - Topical Closeness

- Experiments & Results

- Conclusion
Experiments

- Books indexed using Lemur/Indri

- INEX 2007 Book corpus
  - 42,095 books
  - 250 topics with relevance judgements from Live Search
  - On average, 15.56 judgements per query

- 176 queries match title of a Wiki page (70.4%)
## Query Expansion

<table>
<thead>
<tr>
<th>Run id</th>
<th># judged</th>
<th>MAP</th>
<th>Bpref</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>rel.</td>
<td>non-rel.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>baseline</td>
<td>1666</td>
<td>808</td>
<td>0.3771</td>
<td>0.6131</td>
</tr>
<tr>
<td>$N = 5$</td>
<td>1666</td>
<td>808</td>
<td>0.3725</td>
<td><strong>0.6205</strong></td>
</tr>
<tr>
<td>$N = 10$</td>
<td>1671</td>
<td>808</td>
<td><strong>0.3874</strong></td>
<td>0.6168</td>
</tr>
<tr>
<td>$N = 20$</td>
<td>1667</td>
<td>807</td>
<td>0.3837*</td>
<td>0.6149</td>
</tr>
<tr>
<td>$N = 50$</td>
<td>1666</td>
<td>806</td>
<td>0.3780</td>
<td>0.6136</td>
</tr>
<tr>
<td>$N = 100$</td>
<td>1666</td>
<td>807</td>
<td>0.3780*</td>
<td>0.6133</td>
</tr>
</tbody>
</table>
Query Expansion

<table>
<thead>
<tr>
<th>Run id</th>
<th># jugded rel.</th>
<th># jugded non-rel.</th>
<th>MAP</th>
<th>Bpref</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>1666</td>
<td>808</td>
<td>0.3771</td>
<td>0.6131</td>
<td>0.3040</td>
</tr>
<tr>
<td>$N = 5$</td>
<td>1666</td>
<td>808</td>
<td>0.3725</td>
<td></td>
<td>0.6205</td>
</tr>
<tr>
<td>$N = 10$</td>
<td>1671</td>
<td>808</td>
<td>0.3874*</td>
<td></td>
<td>0.6168</td>
</tr>
<tr>
<td>$N = 20$</td>
<td>1667</td>
<td>807</td>
<td>0.3837*</td>
<td></td>
<td>0.6149</td>
</tr>
<tr>
<td>$N = 50$</td>
<td>1666</td>
<td>806</td>
<td>0.3780</td>
<td></td>
<td>0.6136</td>
</tr>
<tr>
<td>$N = 100$</td>
<td>1666</td>
<td>807</td>
<td>0.3780*</td>
<td></td>
<td>0.6133</td>
</tr>
</tbody>
</table>

• We see:
  ✴ QE has almost no effect on number of relevant documents
Query Expansion

<table>
<thead>
<tr>
<th>Run id</th>
<th># jugged</th>
<th>MAP</th>
<th>Bpref</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>rel.</td>
<td>non-rel.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>baseline</td>
<td>1666</td>
<td>808</td>
<td>0.3771</td>
<td>0.6131</td>
</tr>
<tr>
<td>$N = 5$</td>
<td>1666</td>
<td>808</td>
<td>0.3725</td>
<td>0.6205</td>
</tr>
<tr>
<td>$N = 10$</td>
<td>1671</td>
<td>808</td>
<td><strong>0.3874</strong>*</td>
<td>0.6168</td>
</tr>
<tr>
<td>$N = 20$</td>
<td>1667</td>
<td>807</td>
<td>0.3837*</td>
<td>0.6149</td>
</tr>
<tr>
<td>$N = 50$</td>
<td>1666</td>
<td>806</td>
<td>0.3780</td>
<td>0.6136</td>
</tr>
<tr>
<td>$N = 100$</td>
<td>1666</td>
<td>807</td>
<td>0.3780*</td>
<td>0.6133</td>
</tr>
</tbody>
</table>

- We see:
  - QE has almost no effect on number of relevant documents
  - Improvements are small but significant for MAP and P10
  - Impact drops with increasing $N$ (consequence of term weighting)
## Topical Closeness

<table>
<thead>
<tr>
<th>Run id</th>
<th>MAP</th>
<th>Bpref</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>0.3771</td>
<td>0.6131</td>
<td>0.3040</td>
</tr>
<tr>
<td>References</td>
<td>0.3769</td>
<td>0.6150</td>
<td>0.3051</td>
</tr>
<tr>
<td>Doc.Sim.1</td>
<td>0.3604</td>
<td>0.6010</td>
<td>0.2983</td>
</tr>
<tr>
<td>Doc.Sim.3</td>
<td>0.3790</td>
<td>0.6245*</td>
<td><strong>0.3091</strong>*</td>
</tr>
<tr>
<td>Doc.Sim.5</td>
<td><strong>0.3823</strong>*</td>
<td><strong>0.6251</strong>*</td>
<td>0.3080*</td>
</tr>
</tbody>
</table>

- Final RSV is Indri score + sigmoid trans. of closeness score
Topical Closeness

<table>
<thead>
<tr>
<th>Run id</th>
<th>MAP</th>
<th>Bpref</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>0.3771</td>
<td>0.6131</td>
<td>0.3040</td>
</tr>
<tr>
<td>References</td>
<td>0.3769</td>
<td>0.6150</td>
<td>0.3051</td>
</tr>
<tr>
<td>Doc.Sim.1</td>
<td>0.3604</td>
<td>0.6010</td>
<td>0.2983</td>
</tr>
<tr>
<td>Doc.Sim.3</td>
<td>0.3790</td>
<td>0.6245</td>
<td>0.3091</td>
</tr>
<tr>
<td>Doc.Sim.5</td>
<td><strong>0.3823</strong></td>
<td><strong>0.6251</strong></td>
<td><strong>0.3080</strong></td>
</tr>
</tbody>
</table>

- Final RSV is Indri score + sigmoid trans. of closeness score
  - References have small impact (because of small overlap)
  - Doc.Sim. using top 1 Wiki page hurts performance
## Topical Closeness

<table>
<thead>
<tr>
<th>Run id</th>
<th>MAP</th>
<th>Bpref</th>
<th>P10</th>
</tr>
</thead>
<tbody>
<tr>
<td>baseline</td>
<td>0.3771</td>
<td>0.6131</td>
<td>0.3040</td>
</tr>
<tr>
<td>References</td>
<td>0.3769</td>
<td>0.6150</td>
<td>0.3051</td>
</tr>
<tr>
<td>Doc.Sim.1</td>
<td>0.3604</td>
<td>0.6010</td>
<td>0.2983</td>
</tr>
<tr>
<td>Doc.Sim.3</td>
<td>0.3790</td>
<td>0.6245*</td>
<td>0.3091*</td>
</tr>
<tr>
<td>Doc.Sim.5</td>
<td>0.3823*</td>
<td>0.6251*</td>
<td>0.3080*</td>
</tr>
</tbody>
</table>

- Final RSV is Indri score + sigmoid trans. of closeness score
  - References have small impact (because of small overlap)
  - Doc.Sim. using top 1 Wiki page hurts performance
  - Doc.Sim. using multiple Wiki pages improves all measures (significantly for $N = 5$)
Conclusions (1/2)

• Wikipedia as intermediary between user and book collections:
  ✳ Wikipedia covers many search topics and books topics
Conclusions (1/2)

- Wikipedia as intermediary between user and book collections:
  - Wikipedia covers many search topics and books topics
- Can we automatically extract useful terms from related Wikipedia pages to improve retrieval effectiveness?
  - Yes, QE using $tf.idf$ term selection from single entry point leads to small improvements
  - Problem: entry point might not be relevant
Conclusions (2/2)

- Is the link distance between query pages and book pages related to relevance and can we use this to improve retrieval effectiveness?
  - link distance shows weak relation to relevance
  - document similarity using multiple Wiki pages can significantly improve performance
Conclusions (2/2)

- Is the link distance between query pages and book pages related to relevance and can we use this to improve retrieval effectiveness?
  - link distance shows weak relation to relevance
  - document similarity using multiple Wiki pages can significantly improve performance

- Low number of judgements might not properly reflect effectiveness of chosen methods
  - Last year’s (2008) INEX Book Track topics have deeper pools
  - judgements are about to be released
Future Work

- Use different book representations to find best matching Wikipedia pages:
  - Use collocations, latent semantic indexing
Future Work

- Use different book representations to find best matching Wikipedia pages:
  - Use collocations, latent semantic indexing

- There are uninformative links (Abraham Lincoln $\rightarrow$ 2006)
  - leads to noise closeness scores
  - filter uninformative links
  - weight links by measuring document similarity between two topics
Thank You!