PageRank without hyperlinks: Structural re-ranking using links induced by language models

Oren Kurland and Lillian Lee
{kurland,llee@cs.cornell.edu}

Computer Science Department
Cornell University
High Accuracy in Ad Hoc IR

• Users want/need high accuracy (precision at top ranks)
  – HARD track (TREC), e.g. Shah&Croft ‘04
• Pseudo-feedback-based retrieval
  – Estimate a query model from an initial (short) list of retrieved documents and use it for re-ranking the corpus
• Question answering
  – Use an initial list of retrieved documents to find the required answer
A Re-Ranking Approach

Given an initial retrieved list $D_{init}$ in response to query $q$, re-rank it to obtain high precision at top ranks

- $D_{init}$ is not necessarily ranked
- one can assume documents in $D_{init}$ are somewhat similar to $q$
- it is not a pseudo-feedback approach …
  - we do not assume documents in $D_{init}$ are relevant
  - our methods can enhance the performance of pseudo-feedback-based methods
Inspiration: Web Retrieval

Common approach to web retrieval:

- Re-rank (an initial retrieved list of documents) by the degree of *centrality* (Brin&Page ‘98, Kleinberg ‘99)
- *Centrality* of a document is estimated using explicit hyperlink structure (PageRank, HITS)

Can we use the scoring by *centrality* approach for ranking non-hypertext documents?
A Possible Strategy: Structural Re-Ranking

- Use inter-document similarities to infer links between documents in $D_{init}$
- On the resultant graph (of documents and induced links) define centrality measures and use them as criteria for ranking
How to induce links?

One might suggest:

Vector Space Model (for information representation) and cosine (for similarity metric)

Erkan&Radev ‘04 : text summarization

• Cosine similarity between sentences

but …
Inducing Links

\[
\cos(d_1, d_2) = \cos(d_2, d_1)
\]
Generation Links

How can we capture the asymmetry?

The probability of \{Toronto, Sheffield, Salvador\} "generating" \{Salvador, Salvador, Salvador, Salvador\} is higher than the other way around.

\[ p_d(\bullet) \text{ a unigram LM induced from } d \]

\[
p_{\text{d}_1}\{\text{Toronto, Sheffield, Salvador}\}\{\text{Salvador, Salvador, Salvador}\} > p_{\text{d}_2}\{\text{Salvador, Salvador, Salvador}\}\{\text{Toronto, Sheffield, Salvador}\}\]

\[
\frac{1}{3} \bullet \frac{1}{3} \bullet \frac{1}{3} \quad \text{ vs. } \quad 0 \bullet 0 \bullet 1
\]
“Chase” the generators

Suggested approach:

seek for documents with LMs probable to “generate” other documents

• Generalization of the standard LM approach (Ponte&Croft 98’)
Coping with Length Bias in Generation Probabilities

Problem with unigram LMs:

"\( p(\tilde{w}) = p(w_1) \cdots p(w_n) \)" will penalize long documents

• can affect our graphs in a number of ways

\[
p_d(\tilde{w}) \triangleq \exp \left( -D \left( \tilde{p}_w^{MLE} (\cdot) || \tilde{p}_d^{smoothed} (\cdot) \right) \right)
\]

We use:

\[
= \left( p_d^{smoothed} (\tilde{w}) \right)^{\frac{1}{|\tilde{w}|}} \cdot \exp \left( H (\tilde{p}_w^{MLE} (\cdot)) \right)
\]

geometric mean \hspace{1cm} \text{entropy-based term}

Geometric mean of the MLE was used in TDT: Lavrenko et al. ‘02
Generation Graphs

For document $o \in D_{\text{init}}$:

* $TopGen(o) \overset{\triangle}{=} k$ documents $g \in D_{\text{init}}$ that yield the highest $p_g(o)$
* $g \in TopGen(o)$ is a “generator” of $o$ ($o$ is an offspring of $g$)

The complete graph $G(D_{\text{init}}, D_{\text{init}} \times D_{\text{init}})$ with edge weights:

$$\text{wt}(o \rightarrow g) = \delta[g \in TopGen(o)] \cdot p_g(o)$$

The smoothed (Brin&Page '98) complete graph $G^{[\lambda]}(D_{\text{init}}, D_{\text{init}} \times D_{\text{init}})$ with edge weights:

$$\text{wt}^{[\lambda]}(o \rightarrow g) = \frac{1 - \lambda}{|D_{\text{init}}|} + \lambda \cdot \sum_{g' \in D_{\text{init}}} \text{wt}(o \rightarrow g')$$
Inducing Centrality: \textit{Weighted Influx} Algorithm

\[
Cen_{WI}(d;G) \triangleq \sum_{o \in D_{\text{init}}} wt(o \rightarrow d)
\]

Document in \( D_{\text{init}} \) is \textit{central} if it’s a “good” generator of many other documents in \( D_{\text{init}} \):

- Weighted in-degree analog of the “\textit{journal impact factor}” (Garfield ‘72)
Inducing Centrality: Recursive \textit{Weighted Influx} Algorithm

\[
Cen_{RWI}(d; G^{[\lambda]}) \triangleq \sum_{o \in D_{init}} wt^{[\lambda]}(o \rightarrow d) \cdot Cen_{RWI}(o; G^{[\lambda]})
\]

\[
\text{s.t. } \sum_{d \in D_{init}} Cen_{RWI}(d; G^{[\lambda]}) = 1
\]

- Value quality over quantity (Pinski&Narin ‘76)
- Smoothed graph $G^{[\lambda]}$: “right” kind of Markov chain, power method converges
- The \textit{Recursive Weighted Influx} algorithm is a weighted analog of PageRank (Brin&Page ‘98)

$\text{Cen}_{WI}(Y) = \text{Cen}_{WI}(A)$

$\text{Cen}_{RWI}(Y) > \text{Cen}_{RWI}(A)$
Experimental Setting

- Evaluation measure: prec@5 (see paper for prec@10, mrr)
- $D_{init}$ contains 50 documents retrieved by a LM baseline
  - optimized for (non-interpolated) avg. precision at 1000
- Our algorithms:
  - Optimize $k$ (number of top-generators) and $\lambda$ (smoothing parameter)
Ranking by Centrality

Can we do better?
The Language Modeling Framework

Score by: $Cen(d; G) \cdot p_d(q)$  cf. $p(d) \cdot p(q \mid d)$

- doc “prior”  initial ranking

Algorithms: (Recursive) *Weighted Influx*+LM

Lafferty & Zhai ’01: “with hypertext, [a document prior] might be the distribution calculated using the ‘PageRank’ scheme”
LM Framework with Centrality Scores as “priors”

**prec @ 5**

- **AP**
  - init rank: 51
  - W-Influx+LM: 56
  - RW-Influx+LM: 60
- **TREC8**
  - init rank: 46
  - W-Influx+LM: 51
  - RW-Influx+LM: 56
- **WSJ**
  - init rank: 41
  - W-Influx+LM: 51
  - RW-Influx+LM: 56
- **AP89**
  - init rank: 31
  - W-Influx+LM: 26
  - RW-Influx+LM: 30
Comparing Centrality Measures

Best

- Structural re-ranking: Recursive Weighted Influx
- Structural re-ranking: Weighted Influx
- entropy, log(length), log(#unique terms)
- uniform distribution (initial ranking)

Worst

- length, #unique terms
Cosine vs. LM Probabilities
Additional Issues

- Uniform edge weights:
  - outperform the initial ranking
  - slightly less effective than using generation weights

- Performance not strongly correlated to percentage of relevant documents in $D_{init}$

- Effect of the number of docs in $D_{init}$
  - 50, 100: improvements over init ranking
  - 1000: performance degradation compared to 50 and 100
  - entire corpus (ranking vs. re-ranking): no improvements over initial ranking

- Inducing centrality with HITS (Kleinberg ’98)
  - authority score more effective than hub score
  - outperform initial ranking
  - performance in several cases is inferior to that of the (Recursive) Weighted Influx algorithms
Related Work

– Using PageRank/HITS with no hyperlinks
  • Erkan&Radev ‘04: text summarization
  • Mihalcea and Tarau ‘04: text summarization, keyword extraction
  • Toutanova et al. ‘04: prepositional phrase attachment
– Modeling document prior (centrality) in the LM framework for more specific domains
  • Miller et al. ‘99: document source
  • Kraaij et al. ‘02: hyperlink in-degree, URL form
  • Li&Croft ‘03: time-based LMs
– Query-dependent clustering
  • Willet ‘85, Hearst&Pederson ‘96, Leuski ‘02, Tombros et al. ‘02, Liu&Croft ‘04
Summary

• Novel approach to re-ranking using centrality measures defined on generation graphs; our algorithms:
  – exhibit substantially better performance than that of the initial ranking
  – provide better estimate for document’s centrality than previously proposed measures
  – work both in the vector space and in the LM simplex
    • LM is better
  – are not strongly dependent on the percentage of relevant documents in the initial list