A Generation Model To Unify Topic Relevance and Lexicon-based Sentiment For Opinion Retrieval

Min Zhang, Xingyao Ye
Tsinghua University, Beijing, China
z-m@tsinghua.edu.cn
July 22, 2008
Introduction

- A growing interest in finding out people’s opinions from web data
  - Product survey
  - Advertisement analysis …
  - Political opinion polls
  - TREC started a special track on blog data in 2006 – blog opinion retrieval
    - It has been the track that has the most participants in 2007
Related Work

- The popular opinion identification approaches
  - Text classification [Tong 01, Liu 05, Zhang 06, etc]
  - Lexicon-based sentiment analysis [Liao 06, Mishne 06, Yang 06, Oard 06, Macdonald 07]
  - Opinion retrieval

- Opinion retrieval: To find the sentimental relevant documents according to a user’s query

- Topicality and polarity are first fused together to form the notion of opinion retrieval by Hurst and Nigam [Hurst 04]
  - Emphasize on how to judge the existence of opinions

- First generation model on opinion ranking using the cross entropy of topics and sentiments [Eguchi 06]

\[ \alpha \sum_v R_t(v) \log P_t(v) + (1 - \alpha) \sum_v R_s(v) \log P_s(v) \]
Related Work (cont.)

- One of the key problems: How to combine opinion score with relevance score of each document for ranking
- Ad hoc solutions of combining relevance ranking and opinion detection result
  - 2 steps: rank with relevance, then re-rank with sentiment score
  - Generally linear combination by experience
  - TREC blog 06 observation [Ounis 06]
    - Existing methods to sentimental document ranking provide no improvements over mere topic-relevance ranking
  - TREC blog 07
    - Better result, but still an interesting observation that the topic-relevance result outperforms most opinion-based approaches
Generation Model For Opinion Retrieval

- The Generation Model
  - To find both sentimental and relevant documents with ranks

- Topic Relevance Ranking

- Opinion Generation Model and Ranking

- Ranking function of generation model for opinion retrieval
The Proposed Generation Model

- In existing probabilistic-based IR models, two ways to factor the “relevance” probability [Lafferty 03]
  - query generation and document generation

- Document generation model: how well the document \( d \) “fits” the particular query \( q \), estimate posterior probability \( p(d \mid q) \)

\[
p(d \mid q) \propto p(q \mid d)p(d)
\]

- When assuming a uniform document prior, the ranking function is reduced to the likelihood of generating the expected query terms from the document.
The Proposed Generation Model (Cont.)

- In opinion retrieval, \( p(d \mid q, s) \)
- In this work, discuss lexicon-based sentiment analysis
  - Assume
    - The latent variable \( s \) is estimated with a pre-constructed bag-of-word sentiment thesaurus
    - All sentiment words \( s_i \) are uniformly distributed.
  - Then
    \[
    p(d \mid q, s) = \sum_i p(d \mid q, s_i)p(s_i, s) \\
    = \frac{1}{|s|} \sum_i p(d \mid q, s_i) \cdot \# \text{ of words in sentiment thesaurus} \\
    \propto \frac{1}{|s|} \sum_i p(q, s_i \mid d)p(d) \\
    = \frac{1}{|s|} \sum_i p(s_i \mid d, q)p(q \mid d)p(d)
    \]

given query \( q \), how probably a document \( d \) generates a sentiment word \( s_i \)
The Proposed Generation Model (Cont.)

- The final generation model
  \[ p(d|q,s) = I_{Op}(d,q,s)I_{rel}(d,q). \]

- \( I_{Op}(d,q,s) \overset{\text{def}}{=} \frac{1}{|S|} \sum_i p(s_i|d,q) \), *opinion generation model* to sentiment analysis

- \( I_{rel}(d,q) \overset{\text{def}}{=} p(q|d)p(d) \) *document generation model* to estimate topic relevance

- Essentially it presents a quadratic relationship between document sentiment and topic relevance

- v.s. In previous work, linear combination

  \[ \text{Rel}_\text{Op}_\text{Score} = (1-\lambda)\text{Senti}_\text{Score} + \lambda \text{Rel}_\text{Score} \]
Generation Model For Opinion Retrieval

- The Generation Model
  - To find both sentimental and relevant documents

- Topic Relevance Ranking

- Opinion Generation Model and Ranking

- Ranking function of generation model for opinion retrieval
Topic Relevance Ranking Model $I_{rel}(d,q)$

- The Binary Independent Retrieval (BIR) model is one of the most famous ones in this branch
  - Heuristic ranking function BM25

$$ScoreI_{rel}(d, q) := \sum_{w \in q \cap d} \left( \ln \frac{N - df(w) + 0.5}{df(w) + 0.5} \times \frac{(k_1 + 1)c(w,d)}{k_1((1-b) + b\frac{|d|}{avdl} + c(w,d))} \times \frac{(k_3 + 1)c(w,q)}{k_3 + c(w,q)} \right)$$

- $c(w,d)$ is the count of word $w$ in the document $d$,
- $c(w,q)$ is the count of word $w$ in the document $q$,
- $N$ is the total number of documents in the collection,
- $df(w)$ is the number of documents that contain word $w$
- $|d|$ is the length of document $d$,
- $avdl$ is the average document length,
- $k_1$ (1.0 to 2.0), $b$ (usually 0.75) and $k_3$ (0 to 1000) are constants.
Generation Model For Opinion Retrieval

- The Generation Model
  - To find both sentimental and relevant documents
- Topic Relevance Ranking
- Opinion Generation Model and Ranking
- Ranking function of generation model for opinion retrieval
Opinion Generation Model $I_{op}(d,q,s)$

- $I_{op}(d,q,s)$ focus on the problem that given query $q$, how probably a document $d$ generates a sentiment expression $s$.

\[ I_{op}(d,q,s) \overset{\text{def}}{=} \frac{1}{|S|} \sum_i p(s_i | d, q) \propto \sum_i p(s_i | d, q) \]

- This opinion generation model is on the branch of query generation

- Different from general query generation model
  - $|S|$ is quite large ($\sim$ thousands)
    (v.s. in general models, # of terms in the query is usually small)
  - Sparseness problem $\rightarrow$ smoothing
Opinion Generation Model
– Parameter Estimation (smoothing)

- \[ p(s_i|d,q) = \begin{cases} 
  p_{\text{seen}}(s_i|d,q) & \text{if } s_i \text{ is seen} \\
  p_{\text{unseen}}(s_i|d,q) & \text{otherwise} 
\end{cases} \]

- By Zhai & Lafferty’s study, **Jelinek-Mercer smoothing** is more effective when the “queries” are long and more verbose.
  - In this proposed opinion generation model, the “queries” are sentiment words
    \[ p_s(s_i|d,q) = (1 - \lambda) p_m|l(s_i|d,q) + \lambda p(s_i|c,q), \quad \alpha_d = \lambda. \]

\( p_m|l(s_i|d,q) \): the maximum likelihood estimation of \( p(s_i|d,q) \)
Opinion Generation Model
– Parameter Estimation (smoothing)

- Recall \( I_{op}(d, q, s) \overset{\text{def}}{=} \frac{1}{|s|} \sum_i p(s_i | d, q) \propto \sum_i p(s_i | d, q) \)

\[ \sum_i p(s_i | d, q) \]
\[ = \sum_{s_i \in d} p(s_i | d, q) + \sum_{s_i \notin d} p(s_i | d, q) \]
\[ = \sum_{s_i \in d} p_s(s_i | d, q) + \sum_{s_i \notin d} \alpha_d p(s_i | c, q) \]
\[ = \sum_{s_i \in d} [(1 - \lambda) p_{ml}(s_i | d, q) + \lambda p(s_i | c, q)] + \sum_{s_i \notin d} \lambda p(s_i | c, q) \]
\[ = \sum_{s_i \in d} (1 - \lambda) p_{ml}(s_i | d, q) + \lambda \sum_i p(s_i | c, q) \]
\[ = \sum_{s_i \in d} (1 - \lambda) p_{ml}(s_i | d, q) + \lambda \]

- We use the co-occurrence of \( s_i \) and \( q \) inside \( d \) within a window \( W \) as the ranking measure of \( p_{ml}(s_i | d, q) \), then

\[
\text{Score}_{I_{op}}(d, q, s) := \sum_{s_i \in d} (1 - \lambda) \frac{co(s_i, q|w)}{c(q, d)|W|} + \lambda
\]

\( co(s_i, q|w) \): the frequency of \( s_i \) which is co-occurred with \( q \) within \( W \)
\( c(q, d) \): the query term frequency in the document, \( |W| \): window size
Generation Model For Opinion Retrieval

- The Generation Model
  - To find both sentimental and relevant documents
- Topic Relevance Ranking
- Opinion Generation Model and Ranking
- Ranking function of generation model for opinion retrieval
Ranking function of generation model for opinion retrieval

○ The final ranking function of the proposed generation model

\[
p(d \mid q,s) = \text{Score}_{I_{op}}(d,q,s) \times \text{Score}_{I_{rel}}(d,q)
\]

\[
= (\sum_{S_i \in d} (1 - \lambda) \frac{\text{co}(s_i,q \mid W)}{\text{c}(q,d) \cdot |W|} + \lambda) \times \text{Score}_{I_{rel}}(d,q)
\]

\[
p(d \mid q,s) = \text{Score}_{I_{rel}}(d,q) \quad \lambda = 1
\]

○ To reduce the impact of unbalance between #(sentiment words) and #(query terms) \rightarrow \text{logarithm normalization}

\[
p(d \mid q,s) = [(1 - \lambda) \log(\sum_{S_i \in d} \frac{\text{co}(s_i,q \mid W)}{\text{c}(q,d) \cdot |W|} + 1) + \lambda] \times \text{Score}_{I_{rel}}(d,q)
\]
Experimental Setup – Data Set

- **Data set**
  - TREC blog 06 and TREC blog 07 data
    - Permalinks, homepages and feeds
  - 100,649 blogs during 2.5 months
  - Only use permalinks in this work
  - 50 + 50 topics
  - Short queries (only <title> field)

- **Strategy**: find top 1000 relevant documents, then re-rank the list with proposed model
Experimental Setup – Models

- General linear combination (Shown as *Linear Comb.*)
  \[ \text{Rel}_Op\_Score = (1 - \lambda) \text{Senti}_\_Score + \lambda \text{Rel}_\_Score \]

- Our proposed generation model with Jelinek-Mercer smoothing (Shown as *generation model*)
  \[ p(d \mid q, s) = \text{ScoreI}_{op}(d, q, s) \times \text{ScoreI}_{rel}(d, q) \]
  \[ = \left( \sum_{S_i \in d} (1 - \lambda) \frac{\text{co}(s_i, q \mid W)}{c(q, d) \cdot |W|} + \lambda \right) \times \text{ScoreI}_{rel}(d, q) \]

- Our proposed generation model with Jelinek-Mercer smoothing and logarithm normalization (Shown as *Generation, log*)
  \[ p(d \mid q, s) = \left( (1 - \lambda) \log \left( \sum_{S_i \in d} \frac{\text{co}(s_i, q \mid W)}{c(q, d) \cdot |W|} + 1 \right) + \lambda \right) \times \text{ScoreI}_{rel}(d, q) \]
## Experimental Setup – Sentimental Lexicons

<table>
<thead>
<tr>
<th>Thesaurus Name</th>
<th>Size</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HowNet</td>
<td>4621</td>
</tr>
<tr>
<td>2</td>
<td>WordNet</td>
<td>7426</td>
</tr>
<tr>
<td>3</td>
<td>Intersection</td>
<td>1413</td>
</tr>
<tr>
<td>4</td>
<td>Union</td>
<td>10634</td>
</tr>
<tr>
<td>5</td>
<td>General Inquirer</td>
<td>3642</td>
</tr>
<tr>
<td>6</td>
<td>SentiWordNet</td>
<td>3133</td>
</tr>
</tbody>
</table>
Experimental Results And Discussion

- Effectiveness of Sentimental Lexicons
- Selection of Window Size
- Opinion Retrieval Model Comparison
- Per-topic Analysis
- Case Study
1. Effectiveness of Sentimental Lexicons

Figure 3 MAP - λ curve for different sentiment thesaurus. (Blog 06 Data)
All the following experiments use sentiWordNet
2. Selection of Window Size

Figure 4. MAP v.s. window size under different $\lambda$ (Blog06)
3. Opinion Retrieval Model Comparison

Figure 5. MAP-\(\lambda\) curve for different opinion ranking formulas
3. Opinion Retrieval Model Comparison (Cont.)

Figure 6. Precision-recall curves before and after opinion re-ranking of top 1000 relevant documents
3. Opinion Retrieval Model Comparison (Cont.)

![Graph showing the effect of opinion re-ranking (Hownet)](image)

- **TREC06 Before Re-ranking**
- **TREC06 After Re-ranking**
- **TREC07 Before Re-ranking**
- **TREC07 After Re-ranking**

The graph illustrates the precision and recall of retrieval models before and after re-ranking for the TREC06 and TREC07 datasets.
## 3. Opinion Retrieval Model Comparison (Cont.)

<table>
<thead>
<tr>
<th>Data Set</th>
<th>Method</th>
<th>MAP</th>
<th>R-Prec</th>
<th>P@10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blog 06</td>
<td>Best run at blog 06</td>
<td>0.2052</td>
<td>0.2881</td>
<td>0.468</td>
</tr>
<tr>
<td></td>
<td>Best title-run at blog 06</td>
<td>0.1885</td>
<td>0.2771</td>
<td>0.512</td>
</tr>
<tr>
<td></td>
<td>Our Relevance Baseline(title-run)</td>
<td>0.1758</td>
<td>0.2619</td>
<td>0.350</td>
</tr>
<tr>
<td></td>
<td>Our Unified Model</td>
<td><strong>0.2257</strong></td>
<td><strong>0.3038</strong></td>
<td>0.507</td>
</tr>
<tr>
<td>Blog 07</td>
<td>Greatest improvement at blog 07</td>
<td>15.9%</td>
<td>8.6%</td>
<td>21.6%</td>
</tr>
<tr>
<td></td>
<td>Our Relevance Baseline(title-run)</td>
<td>0.2632</td>
<td>0.3249</td>
<td>0.432</td>
</tr>
<tr>
<td></td>
<td>Our Unified Model *</td>
<td>0.3371</td>
<td>0.3896</td>
<td>0.606</td>
</tr>
<tr>
<td></td>
<td>improvement</td>
<td><strong>28.1%</strong></td>
<td><strong>19.9%</strong></td>
<td><strong>40.3%</strong></td>
</tr>
</tbody>
</table>

*: on Blog 07 data, use the same parameters as those on Blog 06 data, \( \lambda = 0.6 \), window = full, thesaurus: SentiWordNet
4. per-topic analysis (MAP Gain)
4. per-topic analysis (p@10 Gain)
5. Case Study

Details of the best re-ranked topics examples

<table>
<thead>
<tr>
<th>Topic</th>
<th>Title</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TREC 06 - 895</td>
<td>Oprah</td>
<td>Find opinions about Oprah Winfrey's TV show</td>
</tr>
<tr>
<td></td>
<td>MAP</td>
<td>Prec@10</td>
</tr>
<tr>
<td>Before re-ranking</td>
<td>0.0687</td>
<td>0.2000</td>
</tr>
<tr>
<td>After re-ranking</td>
<td>0.2721</td>
<td>0.8000</td>
</tr>
<tr>
<td>Topic</td>
<td>Title</td>
<td>Description</td>
</tr>
<tr>
<td>TREC 07 - 946</td>
<td>tivo</td>
<td>Find opinions about TIVO brand digital video recorders</td>
</tr>
<tr>
<td></td>
<td>MAP</td>
<td>Prec@10</td>
</tr>
<tr>
<td>Before re-ranking</td>
<td>0.2779</td>
<td>0.1000</td>
</tr>
<tr>
<td>After re-ranking</td>
<td>0.4991</td>
<td>1.0000</td>
</tr>
</tbody>
</table>
TiVo’s custom wireless G USB adapter has arrived. As previously reported the adapter is compatible with Series 2 models and off-loads some of the network processing, by utilizing a full MAC chipset...

Sells for $50 ……

I understand it’s not the adapter but the TiVo software, but I don’t care. **TiVo could fix that pretty easy but** chooses not to at the expense of our security. I run my TiVo wireless. Mainly because...

I guess it’s a good thing indirectly, less incentive to watch TV. Personally I think that it’s nice to be able to organise TV life around your life

**Not having WPA is a fault of the TiVo software**, not the device (AFAIK this is not a “smart” device). I run all my Tivos with Netgear WG111s... **I’ll also be a bit miffed** if they turn out to get decent transfer speed with these things as Tivo just sold me the 111s as their fastest transfer method, ……
Tivo

Monday, August 15, 2005

The Internet's tivo software Resource    Search Google:    Other tivo software Resources: Hacking the TiVo TiVo is a trademark of TiVo Inc. This site or software on this site is in no way affiliated with or endorsed by TiVo Inc. TiVo Community Forum - powered by vBulletin This also includes hacks that remove ads from TiVo software. 17385 2434... USATODAY.com - TiVo investors give standing ovation to Comcas..

posted by mdb @   &nbsp;

0 Comments:

Ranked 5 → Ranked 306
Conclusion

- Proposed a formal generation opinion retrieval model
  - Topic relevance & sentimental scores are integrated with quadratic comb.

- Opinion generation ranking functions are derived
  - Using the language modeling approach with smoothing
  - With logarithm normalization paradigm

- Discussed the roles of the sentiment lexicon and the matching window.

- It is a general model for opinion retrieval
  - Domain-independent lexicons
  - No assumption has been made on the nature of blog-structured text
Future work

- Automatically constructing collection-based sentiment lexicons
- Understanding the nature of opinion expressing behavior on the Web
- Go beyond document re-ranking
  - Opinion-oriented index
  - Use linguistic information
  - .......
Thanks for your attention!
Questions & comments?