A Generation Model to Unify Topic Relevance and Lexicon-based Sentiment for Opinion Retrieval

Min Zhang
State key lab of Intelligent Tech.& Sys,
Dept. of Computer Science,
Tsinghua University, Beijing, 100084, China
86-10-6279-2595
z-m@tsinghua.edu.cn

Xingyao Ye
School of Software
Tsinghua University
Beijing, 100084, China
86-10-5153-1413
yexy04@mails.tsinghua.edu.cn

ABSTRACT
Opinion retrieval is a task of growing interest in social life and academic research, which is to find relevant and opinionate documents according to a user's query. One of the key issues is how to combine a document's opinionate score (the ranking score of to what extent it is subjective or objective) and topic relevance score. Current solutions to document ranking in opinion retrieval are generally ad-hoc linear combination, which is short of theoretical foundation and careful analysis. In this paper, we focus on lexicon-based opinion retrieval. A novel generation model that unifies *topic-relevance* and *opinion generation* by a *quadratic combination* is proposed in this paper. With this model, the relevance-based ranking serves as the weighting factor of the lexicon-based sentiment ranking function, which is essentially different from the popular heuristic linear combination approaches. The effect of different sentiment dictionaries is also discussed. Experimental results on TREC blog datasets show the significant effectiveness of the proposed unified model. Improvements of 28.1% and 40.3% have been obtained in terms of MAP and p@10 respectively. The conclusion is not limited to blog environment. Besides the unified generation model, another contribution is that our work demonstrates that in the opinion retrieval task, a Bayesian approach to combining multiple ranking functions is superior to using a linear combination. It is also applicable to other result re-ranking applications in similar scenario.

Categories and Subject Descriptors
H.3.3 [**Information Search and Retrieval**]: Retrieval Models

General Terms: Algorithms, Experimentation, Theory

Keywords
Generation model, topic relevance, sentiment analysis, opinion retrieval, opinion generation model

1. INTRODUCTION
In recent years, there is a growing interest in finding out people's opinions from web data. In many cases, obtaining subjective attitudes towards some object, person or event is often a stronger request than getting encyclopedia-like descriptions. General opinion retrieval is an important issue in practical activities such as product survey, political opinion polls, advertisement analysis, etc. Some researchers have observed this underrepresented need of information and made attempts towards efficient detection, extraction and summarization of opinions from web data [7, 8, 15]. However, much of the work focused on presenting a comprehensive and detailed analysis of the sentiments expressed in the text, without studying how well each source document can meet the need of the user. In addition, this branch of work seek solutions to a specific data domain, such as product/movie review websites [7,15] and weblogs [8], so they make use of many field-dependent features such as different aspects of a product, which are not present for other types of text data.

The rising prospects of research and implementation on opinion search are opened up by the explosive amount of user-centric data available recently. People have been writing about their lives and thoughts more freely than ever on personal blogs, virtual communities and special interest forums. Driven by this trend and its intriguing research values, TREC started a special track on blog data in 2006 with a main task of retrieving personal opinions towards various topics, and it has been the track that has the most participants in 2007.

But how to combine opinion score (the ranking score of to what extent it is subjective or objective) with relevance score is a key problem in research. In previous work, there are many examples that the existing methods of document opinion ranking provide no improvements over mere topic-relevance ranking. [12] Things come better in 2007. But there's still an interesting observation that the topic-relevance result outperforms most opinion-based approaches [26]. Ad-hoc solutions have been adopted to combine relevance ranking and the opinion detection result, causing performance to suffer from lack of adequate theoretical support.

In this paper, we focus on the problem of searching opinions over general topics with the aim of presenting a ranked list of documents containing personal opinions towards the given query. We start from the general statistics-based information retrieval, following the idea of taking relevance estimation problem as query generation and document generation. Then considering the opinion retrieval background, we induct the new constrain of sentiment expression into the model. With probabilistic derivation,

SIGIR'08, July 20–24, 2008, Singapore

* Supported by the Chinese National Key Foundation Research & Development Plan (2004CB318108), Natural Science Foundation (60621062, 60503064, 60736044) and National 863 High Technology Project (2006AA01Z141.

we come to a novel generation model that unifies the *topic-relevance* model and the *opinion generation* model by a *quadratic combination*. It is essentially different from the linear interpolation between the document's relevant score and its opinion score, which is popularly used in such tasks. With this proposed model, the relevance-based ranking criterion now serves as the weighting factor for the lexicon-based sentiment ranking function. Experimental results show the significant effectiveness of the proposed unified model. It is reasonable since the relevance score is a reliable indicator of whether opinions, if any, expressed in the document is indeed towards the wanted object. This notion is a novel characteristic of our model because in previous work, the opinion score is always calculated independently to the topic-relevance degree. Furthermore, this process can be viewed as a result re-ranking. Our work demonstrates that in IR and sentiment analysis, a Bayesian approach to combining multiple ranking functions is superior to using a linear combination. It is also applicable to other result re-ranking applications in similar scenario. This opinionate document ranking problem is of fundamental benefits to all opinion-related research issues, in that it can provide high quality results for further feature extraction and user behavior learning.

Although the experiments in this paper are conducted on TREC (Text REtreival Conference) blog 06 and 07 data sets, no characteristic of blog data has been used, such as feature extraction, blog spamming filtering, processing on blog feed and comments, etc. In addition, the lexicons used in this work are all domain-independent ones. Hence the conclusion is not limited to blog environment and the proposed approach is applicable to all opinion retrieval tasks on different kinds of resource.

The rest of the paper is organized as follows. We first review previous work in section 2. In section 3, we present our generation model for opinion retrieval that unifies topic relevance model and sentiment-based opinion generation. Details for estimating model parameters are also discussed in the section. After introducing experiment settings in section 4, we test our generation model with comparative experiments in section 5, together with some further discussions. Finally, we summarize the paper and suggest avenues for future work in section 6.

2. RELATED WORK

There has long been interest in either the topics discussed or the opinions expressed in web documents. A popular approach to opinion identification is text classification [7, 15, 22]. Typically, a sentence classifier is learned from both opinionate and neutral web pages available using language features such as local phrases [15] and domain-specific adjective-noun patterns [7]. In order to calculate an opinion score, the classification result is then combined with topic-relevance score using binary operator [12].

Another line of research on opinionate documents comes from natural language processing and deals with pure text without constraints on the source of opinionate data. The work in general treats opinion detection as a text classification problem and use linguistic features to determine the presence and the polarity of opinions [13, 17, 22]. Nevertheless, they either neglect the problem of retrieving valuable documents [13, 17], or adopt an intuitive solution to ranking that is in a way out of their opinion detection [22].

It is the first in Hurst and Nigam's work [4] that topicality and polarity are first fused together to form the notion of opinion retrieval, i.e. to find opinions *about* a given topic. However in that work, the emphasis is on how to judge the presence of such opinions and no ranking strategy is put forward. The first opinion ranking formula is introduced by Eguchi and Lavrenko [2] as the cross entropy of topics and sentiments under a generation model. The instantiation of this formula, however, does not perform very well in the following TREC opinion retrieval experiments. No encouraging result has been obtained.

Opinion search systems that perform well empirically generally adopt a two-stage approach [12]. Topic-relevance search is carried out first by using relevance ranking (e.g. TF*IDF ranking or language modeling). Then heuristic opinion detection is used to re-rank the documents. One major method to identify opinionate content is by matching the documents with a sentiment word dictionary and calculating term frequency [6, 10, 11, 19]. The matching process is often performed multi-times for different dictionaries and different restrictions on matching. Dictionaries are constructed according to existing lexical categories [6, 10, 19] or the word distribution over the dataset [10, 11, 19]. Matching constraints often concern with the distance between topic terms and opinion terms, which can be thought of as a sliding window. Some require the two types of words to be in the same sentence [10], others set the maximum word allowed between them [19]. After the opinion score is calculated, an effective ranking formula is needed to combine multiple sources of information. Most existing approaches use a linear combination of relevance score and opinion score [6, 10, 19]. A typical example is shown below.

$$\alpha * Score_{rel} + \beta * Score_{opn} \qquad (1)$$

where α and β are combination parameters, which are often tuned by hand or learned to optimize a target metric such as binary preference [10]. Other alternatives include demoting the ranking of neutral documents [11].

Domain specific information has always been studied by researchers. Mishne [22, 23] proposed three simple heuristics with improved opinion retrieval performance by using blog-specific properties. Other works make use of many field-dependent features such as different aspects of a product or movie [7, 15], which are not present for other types of text data. TREC blog track is also an important research and experimental platform for opinion retrieval. The major goal is to explore the information seeking behavior in the blogosphere, with an emphasis on spam detection, blog structure analysis, etc. Hence submitted work often goes to great lengths to exploit the non-textual nature of a blog post [10, 12]. This approach makes strong assumptions on the problem domain and is difficult to generalize.

3. GENERATION MODEL FOR OPINION RETRIEVAL

3.1 A New Generation Model

The opinion retrieval task aims to find the documents that contain relevant opinions according to a user's query. In existing probabilistic-based IR models, relevance is modeled with a binary random variable to estimate "What is the probability that *this* document is relevant to *this* query". There are two different ways to factor the relevance probability, i.e. *query generation* and *document generation* [5].

In order to rank the document by their relevance, the posterior probability $p(d|q)$ is generally estimated, which captures how well

the document d "fits" the particular query q. According to Bayes formula,

$$p(d \mid q) \propto p(q \mid d)p(d) \qquad (2)$$

where $p(d)$ is the prior probability that a document d is relevant to any query, and $p(q|d)$ denotes the probability of query q being "generated" by d. When assuming a uniform document prior, the ranking function is reduced to the likelihood of generating the expected query terms from the document.

However, when explicitly searching for opinions, users' information need is now restricted to only an opinionate subset of the relevant documents. This subset is characterized by sentiment expressions s towards topic q. Thus the ranking estimation for opinion retrieval changes to $p(d|q,s)$.

In this paper, for simplicity, when we discuss the lexicon-based sentiment analysis, the latent variable s is assumed to be a pre-constructed bag-of-word sentiment thesaurus, and all sentiment words s_i are uniformly distributed. Then the prior probability that the document d contains relevant opinions to query q is given by

$$\begin{aligned} p(d \mid q, s) &= \sum_i p(d \mid q, s_i) p(s_i, s) \\ &= \frac{1}{|S|} \sum_i p(d \mid q, s_i) \\ &\propto \frac{1}{|S|} \sum_i p(q, s_i \mid d) p(d) \\ &= \frac{1}{|S|} \sum_i p(s_i \mid d, q) p(q \mid d) p(d) \end{aligned} \qquad (3)$$

where $|s|$ is the number of words in sentiment thesaurus s.

When Referring to Equation 2, it is easy to find that Eq.3 is combined with two factors: the last part $p(q|d)p(d)$ gives the estimation of topic relevance, and the remaining shows that given query q, how probably a document d *generates* a sentiment word s_i. Then Equation 3 is rewritten as:

$$\begin{aligned} &p(d \mid q, s) = I_{op}(d, q, s) I_{rel}(d, q), \quad where \\ &I_{op}(d, q, s) \equiv \frac{1}{|S|} \sum_i p(s_i \mid d, q), \quad I_{rel}(d, q) \equiv p(q \mid d) p(d) \end{aligned} \qquad (4)$$

This is the generation model for opinion retrieval. In this model, $I_{rel}(d,q)$ is the *document generation* probability to estimate topic relevance, and $I_{op}(d,q,s)$ is the opinion generation probability to sentiment analysis.

Essentially it presents a quadratic relationship between document sentiment and topic relevance, which is naturally induced from the opinion generation process and is proven more effective in our experiments than the popular linear interpolation used in previous work, e.g.

$$p(d \mid q, s) \overset{rank}{=} (1-\lambda) p(s \mid d, q) + \lambda p(q \mid d) p(d) \qquad (5)$$

where λ is the linear combination weight.

This result is reasonable since the relevance score is a reliable indicator of whether opinions, if any, expressed in the document is indeed towards the wanted object. This notion is a novel characteristic of our framework in that previous work calculated $p(d|q,s)$ independent of the topic-relevance degree.

In the following two sections, we will discuss the two sub-models in the generation opinion retrieval model respectively.

3.2 Topic Relevance Ranking

In the topic relevance model, $I_{rel}(d,q)$ is based on the notion of document generation. A classic probabilistic model, the Binary Independent Retrieval (BIR) model [5], is one of the most famous ones in this branch. The heuristic ranking function BM25 and its variants have been successfully applied in many IR experiments, including TREC (Text Retrieval Conference) evaluation.

Hence in this paper, we adopt this BIR-based document generation model, by which the topic relevance score $ScoreI_{rel}(d,q)$ given by the ranking function presented in [25] can be shown as:

$$\begin{aligned} ScoreI_{rel}(d, q) = \sum_{w \in q \cap d} (\ln \frac{N - df(w) + 0.5}{df(w) + 0.5} \times \\ \frac{(k_1 + 1) \times c(w, d)}{k_1 (1 - b) + b \frac{|d|}{avdl} + c(w, d)} \times \frac{(k_3 + 1) \times c(w, q)}{k_3 + c(w, q)}) \end{aligned} \qquad (6)$$

where $c(w,d)$ is the count of word w in the document d,

$c(w,q)$ is the count of word w in the query 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(from 1.0 to 2.0),b (usually 0.75) and k_3 (from 0 to 1000) are constants.

3.3 Opinion Generation Model Parameter Estimation

In the opinion generation model, $I_{op}(d,q,s)$ focus on the problem that given query q, how probably a document d *generates* a sentiment expression s. This model is on the branch of *query generation*, in which language model has been shown quite effective in information retrieval during recently years.

The sentiment expressions s is a latent variable in our framework which is not inputted in the query but expected to appear in search results. In this work, we assume s to be a bag-of-word sentiment thesaurus, and sentiment words s is uniformly distributed. Hence

$$I_{op}(d, q, s) \equiv \frac{1}{|S|} \sum_i p(s_i \mid d, q) \propto \sum_i p(s_i \mid d, q) \qquad (7)$$

Different from query generation-based language model in IR, where the number of query terms ($|q|$) is usually small (less than 100, and in most cases be 1 or 2), in our opinion generation model, the number of sentiment words (i.e. $|s|$) is large (generally several thousand), and the sparseness problem is prominent. Hence smoothing has turned out to play an important role for parameter estimation in this proposed model.

$$p(s_i \mid d, q) = \begin{cases} p_{seen}(s_i \mid d, q) \\ p_{unseen}(s_i \mid d, q) \end{cases} = \begin{cases} p_S(s_i \mid d, q) & if\ s_i\ is\ seen \\ \alpha_d\, p(s_i \mid C, q) & otherwise \end{cases} \qquad (8)$$

where $p_S(s_i|d, q)$ is the smoothed probability of a word s_i seen in the document d given query q, α_d is a coefficient controlling the probability mass assigned to unseen words, $p(s_i|C, q)$ is the collection language model given query q.

This unigram model can be estimated using any existing method. As iluustrated in Zhai & Lafferty's study [20], Jelinek-Mercer smoothing is much more effective than the other two when the

"queries" are long and more verbose. In this proposed opinion generation model, the "queries" are sentiment words. Therefore, under this similar scenario, we use the MLE estimation, smoothed by Jelinek-Mercer method. According to Jelinek-Mercer smoothing,

$$p_s(s_i|d,q) = (1-\lambda)\, p_{ml}(s_i|d,q) + \lambda\, p(s_i|C,q), \qquad \alpha_d = \lambda$$

where λ is the smoothing parameter, and $p_{ml}(s_i|d,q)$ is the maximum likelihood estimation of $p(s_i|d,q)$. Then use this smoothing to Equation 7 and Equation 8, we get the estimation:

$$\begin{aligned}
&\sum_i p(s_i \mid d,q) \\
&= \sum_{S_i \in d} p(s_i \mid d,q) + \sum_{S_i \notin d} p(s_i \mid d,q) \\
&= \sum_{S_i \in d} p_S(s_i \mid d,q) + \sum_{S_i \notin d} \alpha_d\, p(s_i \mid C,q) \\
&= \sum_{S_i \in d} [(1-\lambda)\, p_{ml}(s_i \mid d,q) + \lambda p(s_i \mid C,q)] + \sum_{S_i \notin d} \lambda p(s_i \mid C,q) \\
&= \sum_{S_i \in d} (1-\lambda)\, p_{ml}(s_i \mid d,q) + \lambda \sum_i p(s_i \mid C,q) \\
&= \sum_{S_i \in d} (1-\lambda)\, p_{ml}(s_i \mid d,q) + \lambda
\end{aligned} \tag{9}$$

We use the co-occurrence of sentiment word s and query word q inside document d within a window W as the ranking measure of $p_{ml}(s_i|d,q)$. Hence the sentiment score of a document d given by the opinion generation model is:

$$ScoreI_{op}(d,q,s) = \sum_{S_i \in d} (1-\lambda) \frac{co(s_i, q \mid W)}{c(q,d) \cdot |W|} + \lambda \tag{10}$$

Where $co(s_i,q|W)$ is the frequency of sentiment word s_i which is co-occurred with query q within window W, $c(q,d)$ is the query term frequency in the document.

3.4 Ranking function of generation model for opinion retrieval

Taking the topic-relevance rank (Equation 6) and opinion-generation rank (Equation 11), we get the overall ranking function for the unified generation model:

$$\begin{aligned}
p(d \mid q,s) &\overset{rank}{=} ScoreI_{op}(d,q,s) \times ScoreI_{rel}(d,q) \\
&= (\sum_{S_i \in d} (1-\lambda) \frac{co(s_i, q \mid W)}{c(q,d) \cdot |W|} + \lambda) \times ScoreI_{rel}(d,q) \\
&\overset{rank}{=} \begin{cases} (1 + \lambda' TF_{CO(s,q,W)}) \times ScoreI_{rel}(d,q) & if\ \lambda \neq 0 \\ ScoreI_{rel}(d,q) & if\ \lambda = 0 \end{cases} \\
where \quad \lambda' &= \frac{1-\lambda}{\lambda}, \quad TF_{CO(s,q,W)} = \sum_{S_i \in d} \frac{co(s_i, q \mid W)}{c(q,d) \cdot |W|}
\end{aligned} \tag{11}$$

Notice that this ranking function is not the precise quantitative estimation of $p(d|q,s)$, because proportion factor $1/|S|$ in opinion-generation rank is ignored. But this factor has no affect to document ranking and hence this approximation is order-preserving.

In this ranking function, we directly use the co-occurrence frequency as the factor to estimate the generation probability $p_{ml}(s_i|d,q)$. But as mentioned in section 3.3, generally, the number of query terms are relative small, such as 1 or 2, but the size of sentiment thesaurus is really large, e.g. over several thousand or even tens of thousands. In order to reduce this impact of unbalance, the logarithm normalization is taken on opinion ranking. By this way, the ranking function turns out to be:

$$\begin{aligned}
p(d \mid q,s) &\overset{rank}{=} \begin{cases} [1 + \lambda' \log(TF_{CO(s,q,W)} + 1)] \times ScoreI_{rel}(d,q) & if\ \lambda \neq 0 \\ ScoreI_{rel}(d,q) & if\ \lambda = 0 \end{cases} \\
where \quad \lambda' &= \frac{1-\lambda}{\lambda}, \quad TF_{CO(s,q,W)} = \sum_{S_i \in d} \frac{co(s_i, q \mid W)}{c(q,d) \cdot |W|}
\end{aligned} \tag{12}$$

The experimental analysis on this logarithm relationship will be made in section 5.3, which shows the effectiveness of this normalization.

4. EXPERIMENTAL SETUP

4.1 Data set

We test our opinion retrieval model on the TREC Blog06 and Blog07 corpus [12, 26], which is the most authoritative opinion retrieval dataset available up to date.

The corpus is collected from 100,649 blogs during a period of two and a half months. We focus on retrieving permalinks from this dataset since human evaluation result is only available for these documents. There are 50 topics (Topic 851~900) from the TREC 2006 blog opinion retrieval task, and 50 topics (Topic 901~950) from TREC blog 2007. Query terms are extracted from the title field using porter stemming and standard stop words removal.

Generally, queries from blog 06 are used for parameter comparison study, including selection of sentiment thesaurus, window size, and the effectiveness of different models. And queries of blog 07 are used as the testing set, where all the parameters have been tuned in blog 06 data and no modification is made.

4.2 Evaluation

To make the experiments applicable to real word applications and comparable to TREC evaluations, only short queries are used.

The evaluation metrics used are general IR measures, i.e. mean average precision (MAP), R-Precision (R-prec), and precision at top 10 results (p@10). Totally three approaches have been comparative studied in our experiments.

(1) General linear combination (Shown as *Linear Comb*.)

$$p(d \mid q,s) \overset{rank}{=} (1-\lambda) ScoreI_{op}(d,q,s) + \lambda ScoreI_{rel}(d,q)$$

where the $ScoreI_{op}(d,q,s)$ and $ScoreI_{rel}(d,q)$ are computed using the same way as that in the Equation 11.

(2) Our proposed generation model with Jelinek-Mercer smoothing (Shown as *Generation Model*). See Equation 11.

(3) Our proposed generation model with Jelinek-Mercer smoothing and logarithm normalization (Shown as *Generation, log*). See Equation 12.

4.3 Selection of Sentimental Lexicon

For lexicon-based opinion detection methods, the selection of opinion thesaurus plays an important role. There are several online public dictionaries from the area of linguistics, such as WordNet [18] and General Inquirer [14]. We follow the general way [6] to select a small seed sentiment words list of WordNet, and then incrementally enlarge the list with synonyms and antonyms.

Another option is to rely on a self-constructed dictionary. Wilson et al [17] manually selected 8821 words as their sentiment lexicon and it has been used in some other works. Esuli and Sebastiani [3]

scored each word in WordNet regarding its positive, negative and neutral indications to obtain a SentiWordNet lexicon. Words with positive or negative score above a threshold in SentiWordNet are used by some participants of the TREC opinion retrieval task.

Furthermore, we seek help from other languages. HowNet [1] is a knowledge database of the Chinese language, and some of the words in the dictionary have properties of positive or negative. We use the English translation of those sentiment words provided by HowNet.

For comparison, sentimental words from HowNet, WordNet, General Inquirer and SentiWordNet are used as lexicons respectively. Table 1 shows the detail information on the lists.

Table 1. Sentiment thesauruses used in our experiments

	Thesaurus Name	Size	Description
1	HowNet	4621	English translation of positive/negative Chinese words
2	WordNet	7426	Selected words from WordNet
3	Intersection	1413	Words appeared in both 1 and 2
4	Union	10634	Words appeared in either 1or 2
5	General Inquirer	3642	Words in the positive and negative category
6	SentiWordNet	3133	Words with a positive or negative score above 0.6

5. EXPERIMENTAL RESULTS AND DISCUSSION

5.1 Effectiveness of Sentimental Lexicons

The retrieval performance under different sentiment thesauruses is presented in Figure 1. The cross-language HowNet dictionary performs better than all other candidates and is quite insensitive to the smoothing parameter. SentiWordNet and the Intersection thesauri perform next and close to each other. General Inquirer does not perform well and has the worst result.

There might be two reasons that lead to the better performance of using the words from HowNet than using that from WordNet. First, the list generated from WordNet might be lack of diversity since the words come from a limited initial seeds and only synonyms and antonyms are taken into consideration. Second, the English translations of the Chinese sentiment words are annotated by non-native speakers; hence most of them are common and popular terms, which are generally used in the Web environment.

Since the performance of SentiWordNet and HowNet are with no big difference when λ is higher, and SentiWordNet is open in the Internet, we choose SentiWordNet as the sentiment thesaurus in the following experiments to make the experiments much easier to repeat by other researchers.

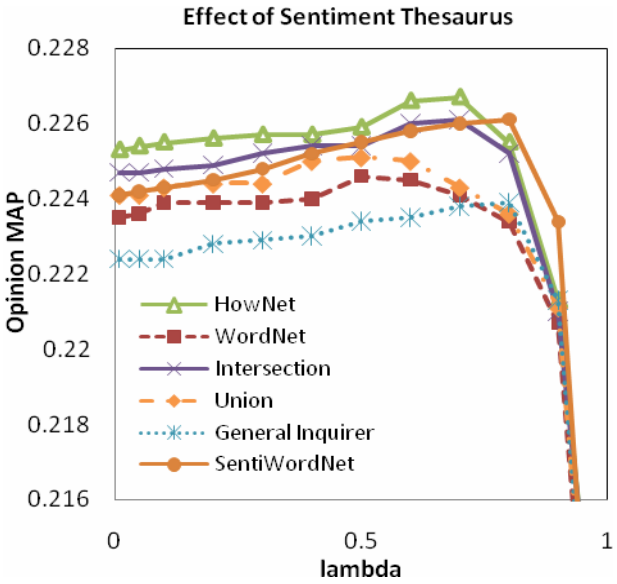

Figure 1 MAP-λ curves with different thesaurus. (Blog 06)

5.2 Selection of Window Size

It is intuitive that opinion modifiers are less likely to be related to an object far away from it than those close to it in the text. Thus during the opinion term matching process, a proximity window is often used to restrict the valid distance between the sentiment words and topic words. However, no one is sure about how close the two types of words should be to each other and this threshold is often set by hand with various indications. In previous work, window sizes that represent the length of direct modification (e.g. 3 [11]), a sentence [10, 22] (e.g. 10~20), a paragraph (e.g. 30~50 [11]), or the whole document [6] have been used.

We test the retrieval performance under these settings respectively to illustrate how this factor could influence the opinion retrieval ability of our model. The result is given in Figure 2.

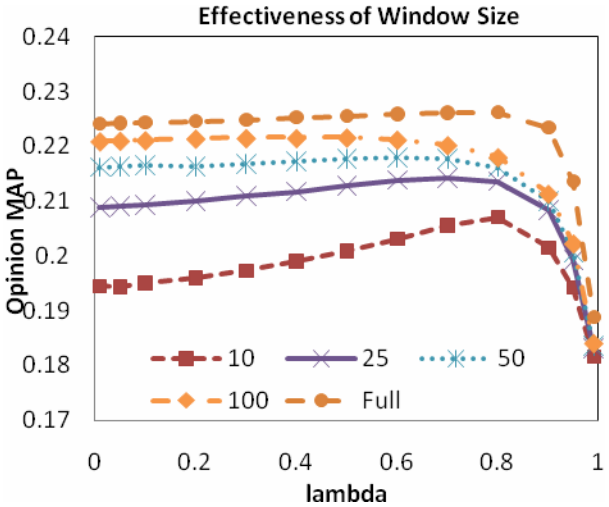

Figure 2. MAP v.s. window size with different λ. (Blog06)

It is clear that the larger the window is, the better the performance is. And this tendency is invariant to different levels of smoothing. The result is reasonable since the distance between a query term and a sentiment word is generally used to demonstrate the opinion relevance to the topic, which has already been taken into consideration in this unified model by the quadratic combination of topic relevance. And in the Web documents, the opinion words may not always been located near the topic words.

Therefore, we set the full document as the default window size in the following experiments.

5.3 Opinion Retrieval Model Comparison

Three opinion ranking formulas are tested in our experiment. Their performance is compared in Figure 3.

We can see that the *generation model* is more effective than *linear combination* especially when mild smoothing is performed. As the value of λ goes up, desired documents with only a few opinion terms are deprived of the discriminative ability contained in their opinion expressions, as this part of the probability is discounted to the whole document collection. *Generation log model* overcomes this problem and gives the best retrieval performance under all values of λ. This demonstrates the usefulness of our log-smoothing approach in the setting of opinion search. In addition, all three ranking schemes perform equivalent to or better than the best run at TREC 2006 owing to the careful selection of sentiment thesaurus and window size as discussed above.

To further demonstrate the effectiveness of our opinion retrieval model, a comparison of opinion MAP with previous work is given in Table 2. Performance improvement after opinion re-ranking is shown in Figure 4 in precision-recall curves.

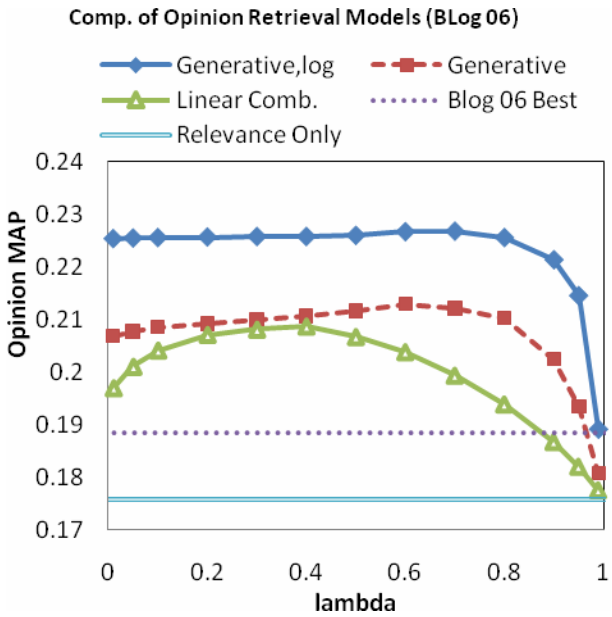

Figure 3. MAP-λ curve for different opinion ranking formulas.

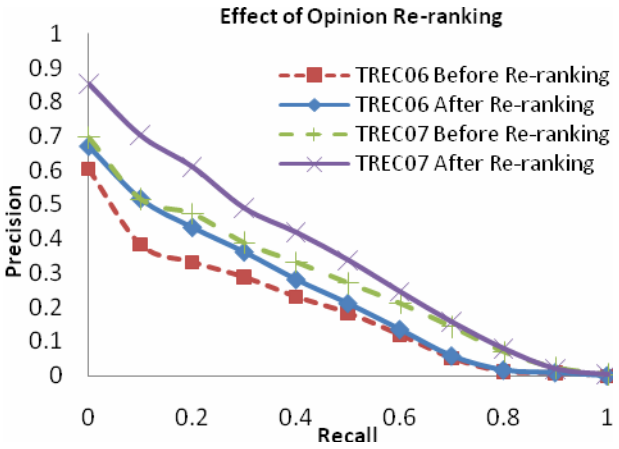

Figure 4. Precision-recall curves before and after opinion re-ranking of top 1000 relevant documents.

Table 2. Comparison of opinion retrieval performance

Data Set	Method	MAP	R-Prec	P@10
Blog 06	Best run at blog 06	0.2052	0.2881	0.468
	Best title-run at blog 06	0.1885	0.2771	**0.512**
	Our Relevance Baseline	0.1758	0.2619	0.350
	Our Unified Model	**0.2257**	**0.3038**	0.507
Blog 07	Most improvement at blog 07	15.9%	8.6%	21.6%
	Our Relevance Baseline	0.2632	0.3249	0.432
	Our Unified Model *	0.3371	0.3896	0.606
	improvement	**28.1%**	**19.9%**	**40.3%**

***: on Blog 07 data, use the same parameters as those on Blog 06 data. λ=0.6, window=full, thesaurus: SentiWordNet.**
All our approaches use title–only run.

In Figure 5, per topic gain in opinion MAP and p@10 are visualized on blog 07 data set. Notice that no characteristic of blog data has been used in this work, such as feature extraction, blog spamming filtering, processing on blog feed and comments, etc. In terms of MAP, 16 of the 50 topics receive improvement of more than 50%, while only 5 topics result in minor performance loss. Few topics that benefit the most from opinion re-ranking, such as topic 912 (144%) and topic 928 (135%), are those where only a few documents with relevant opinions are retrieved and ranked lowly in the first stage. Only 4 topics' performances decrease a little (less than 40%). In terms of p@10, even more significant results are given. Three topics get more than 200% improvement, such as topic 946 (+900%), and only 6 topics get a little drop on performance.

Table 3 gives detailed descriptions of two topics in blog06 and blog07. We can see our re-ranking procedure successfully re-scores almost all the target documents into the top 100 results. This proves our formula to be highly accurate in discriminating a few subjective texts from a large amount of factual descriptions.

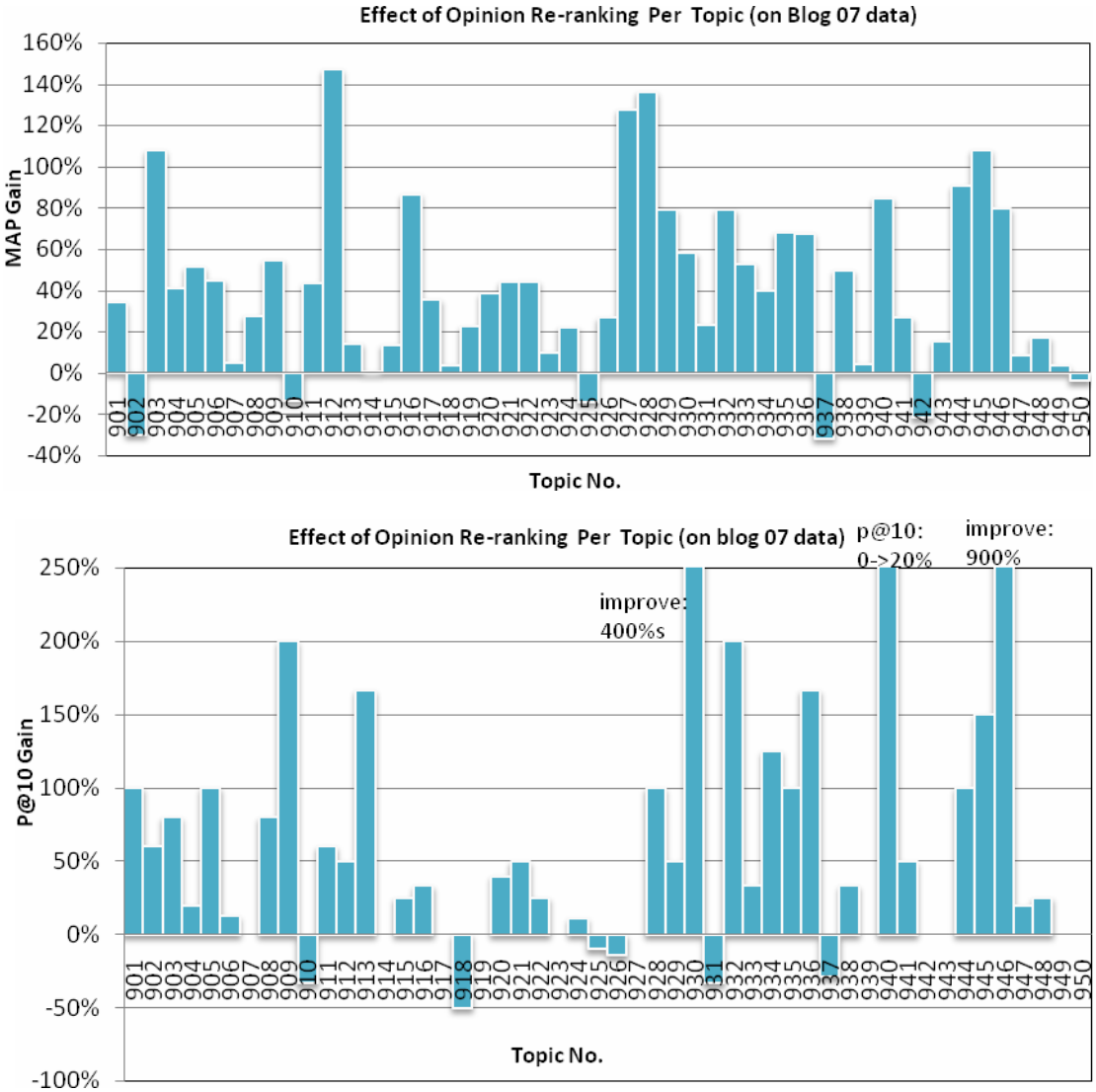

Figure 5. Per-topic analysis: Performance improvement over 50 topics after re-ranking on Blog 07 data.

(a)MAP improvement, (b) p@10 improvement

(in (b), the three topics whose improvement is much higher than the figure upper-bound have been annotated individually.)

Table 3. Details of the best re-ranked topics examples

Topic	**Title**		**Description**		
TREC 06 - 895	Oprah		Find opinions about Oprah Winfrey's TV show		
	MAP	**Prec@10**	**Prec@30**	**Prec@100**	**Prec@1000**
Before re-ranking	0.0687	0.2000	0.0333	0.1200	0.0640
After re-ranking	0.2721	0.8000	0.5000	0.3400	0.0640
Topic	**Title**		**Description**		
TREC 07 - 946	tivo		Find opinions about TIVO brand digital video recorders		
	MAP	**Prec@10**	**Prec@30**	**Prec@100**	**Prec@1000**
Before re-ranking	0.2779	0.1000	0.3333	0.3900	0.2650
After re-ranking	0.4991	1.0000	0.9667	0.8300	0.2650

6. CONCLUSION AND FUTURE WORK

In this work we deal with the problem of opinion search towards general topics. Contrary to previous approaches that view facts retrieval and opinion detection as two distinct parts to be linearly combined, we proposed a formal probabilistic generation model to unify the topic relevance score and opinion score. A couple of opinion re-ranking formulas are derived using the language modeling approach with smoothing, together with logarithm normalization paradigm. Furthermore, the effectiveness of different sentiment lexicons and variant distances between sentiment words and query terms are compared and discussed empirically. Experiment shows that bigger windows are better than smaller windows. According to the experiments, the proposed model yields much better results on TREC Blog06 and Blog07 dataset.

The novelty of our work lies in a probabilistic generation model for opinion retrieval, which is general in motivation and flexible in practice. This work derives a unified model from the quadratic relation between opinion analysis and topic relevance, which is essentially different from general linear combination. Furthermore, in this work, we do not make any assumption on the nature of blog-structured text. Therefore this approach is expected to be generalized to all kinds of resources for opinion retrieval task.

Future directions on opinion retrieval may go beyond merely document re-ranking. An opinion-oriented index, as well as deeper analysis on the structural information of opinion resources such as blogs and forums could be helpful in understanding the nature of opinion expressing behavior on web. Another interesting topic is to automatically construct a collection-based sentiment lexicon, which has been a hot research topic [26], and to induct this lexicon into our generation model.

7. REFERENCES

[1] Dong, Z. HowNet. http://www.HowNet.org

[2] Eguchi, K. and Lavrenko, V. Sentiment Retrieval using Generative Models. In Proceedings of Empirical Methods on Natural Language Processing (EMNLP) 2006, 345-354.

[3] Esuli, A. and Sebastiani, F. Determining the semantic orientation of terms through gloss classification. In Proceedings of CIKM 2005, 617-624.

[4] Hurst, M. and Nigam, K. Retrieving Topical Sentiments from Online Document Collections. Document Recognition and Retrieval XI. 27--34. 2004.

[5] Lafferty, J. and Zhai, C. Probabilistic relevance models based on document and query generation. Language Modeling and Information Retrieval, Kluwer International Series on Information Retrieval, Vol. 13, 2003.

[6] Liao, X., Cao, D., Tan, S., Liu, Y., Ding, G., and Cheng X. Combining Language Model with Sentiment Analysis for Opinion Retrieval of Blog-Post. Online Proceedings of Text Retrieval Conference (TREC) 2006. http://trec.nist.gov/

[7] Liu, B., Hu, M., and Cheng, J. Opinion observer: analyzing and comparing opinions on the Web. WWW 2005: 342-351

[8] Mei, Q., Ling, X., Wondra, M., Su, H., and Zhai, C. Topic sentiment mixture: modeling facets and opinions in weblogs. WWW 2007: 171-180

[9] Metzler, D., Strohman T., Turtle H., and Croft, W.B. Indri at TREC 2004: Terabyte Track. Online Proceedings of 2004 Text REtrieval Conference (TREC 2004), 2004

[10] Mishne, G. Multiple Ranking Strategies for Opinion Retrieval in Blogs. Online Proceedings of TREC, 2006.

[11] Oard, D., Elsayed, T., Wang, J., and Wu, Y. TREC-2006 at Maryland: Blog, Enterprise, Legal and QA Tracks. Online Proceedings of TREC, 2006. http://trec.nist.gov/

[12] Ounis, I., de Rijke, M., Macdonald, C., Mishne, G., and Soboroff, I. Overview of the TREC 2006 Blog Track. In Proceedings of TREC 2006, 15–27. http://trec.nist.gov/

[13] Pang, B., et al, Thumbs up? Sentiment Classification Using Machine Learning Techniques. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP) 2002, 79-86.

[14] Stone, P., Dunphy, D., Smith, M., and Ogilvie, D. The General Inquirer: A Computer Approach to Content Analysis. MIT Press, Cambridge, 1966.

[15] Tong, R. 2001. An Operational System for Detecting and Tracking Opinions in on-line discussion. SIGIR Workshop on Operational Text Classification. 2001. 1-6.

[16] Turtle, H. and Croft, W.B. Evaluation of an Inference Network-Based Retrieval Model. ACM Transactions on Information System, in 9(3),187-222, 1991.

[17] Wilson, T., Wiebe, J., and Hoffmann, P. Recognizing Contextual Polarity in Phrase-Level Sentiment Analysis. In Proceedings of HLT/EMNLP 2005. 347-354.

[18] WordNet. http://wordnet.princeton.edu/

[19] Yang, K., Yu, N., Valerio, A., Zhang, H. WIDIT in TREC-2006 Blog track. Online Proceedings of TREC, 2006. http://trec.nist.gov/

[20] Zhai, C. and Lafferty, J. A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems (ACM TOIS), Vol. 22, No. 2, 179-214.2004.

[21] Zhai, C. A Brief Review of Information Retrieval Models, Technical report, Dept. of Computer Science, UIUC, 2007

[22] Zhang, W. and Yu, C. UIC at TREC 2006 Blog Track. Online Proceedings of TREC, 2006. http://trec.nist.gov/

[23] Mishne, G. and Glance, N. Leave a Reply: An analysis of Weblog Comments. In WWE 2006 (WWW 2006 Workshop on Weblogging Ecosystem), 2006.

[24] Mishne, G. Using blog properties to improve retrieval, In Proceedings of the International Conference on Weblogs and. Social Media (ICSWM) 2007.

[25] Singhal, A. Modern information retrieval: A brief overview. Bulletin of the IEEE Computer Society Technical committee on Data Engineering, 24(4):35-43, 2001.

[26] Macdonald, C. and Ounis, I. Overview of the TREC-2007 Blog Track. Online Proceedings of the 16th Text Retrieval Conference (TREC2007). http://trec.nist.gov/pubs/trec16/t16_proceedings.html