The Use Of Kullback-Leibler Divergence

In Opinion Retrieval

by

Kun Cen

A thesis

presented to the University of Waterloo

in fulfillment of the

thesis requirement for the degree of

Master of Applied Science

in

Management Sciences

Waterloo, Ontario, Canada, 2008

Author's declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

_____

Kun Cen

Abstract

With the huge amount of subjective contents in on-line documents, there is a clear need for an information retrieval system that supports retrieval of documents containing opinions about the topic expressed in a user's query. In recent years, blogs, a new publishing medium, have attracted a large number of people to express personal opinions covering all kinds of topics in response to the real-world events. The opinionated nature of blogs makes them a new interesting research area for opinion retrieval. Identification and extraction of subjective contents from blogs has become the subject of several research projects.

In this thesis, four novel methods are proposed to retrieve blog posts that express opinions about the given topics. The first method utilizes the *Kullback-Leibler divergence* (KLD) to weight the lexicon of subjective adjectives around query terms. Considering the distances between the query terms and subjective adjectives, the second method uses KLD scores of subjective adjectives based on distances from the query terms for document re-ranking. The third method calculates KLD scores of subjective adjectives for predefined query categories. In the fourth method, collocates, words co-occurring with query terms in the corpus, are used to construct the subjective lexicon automatically. The KLD scores of collocates are then calculated and used for document ranking.

Four groups of experiments are conducted to evaluate the proposed methods on the TREC test collections. The results of the experiments are compared with the baseline systems to determine the effectiveness of using KLD in opinion retrieval. Further studies are recommended to explore more sophisticated approaches to identify subjectivity and promising techniques to extract opinions.

Acknowledgements

First and foremost, I would like to take this opportunity to express my gratitude to my supervisor, Dr. Olga Vechtomova, for her guidance and support through the course of this study. Her advice and vision have led me through the difficulties and darkness.

I also feel thankful to Dr. Charles L.A. Clarke and Dr. Mark Smucker for their valuable comments on this thesis.

Finally, thanks to the University of Waterloo for its support. Special thanks must be given to my family for their understanding and encouragement.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Retrieving documents by a certain subject matter is basically the general goal of an Internet search engine. However, with increasing amount of subjective contents across the Web, people may need to find documents not only containing simply the query terms, but also containing other people's opinions about a certain topic. For example, consumers may want to know other people's comments about a specific product to gain more information for their purchasing decisions, while manufacturers may want to know their customers' reviews about their products, as well as other people's reviews about their competitors' products for new product development, marketing and customer relationship management. Today, there is a huge amount of subjective contents in on-line documents, such as web pages, discussion forums, and personal blogs etc. If you are considering a vacation in Italy, you might go to a search engine and enter the query "Italy travel review" or "Opinion for Italy travel". However, the fact is that only a small portion of documents expressing opinions about Italy travel may actually contain the words "review" or "opinion". Particularly, if people want to find reviews about a specific subject, they may go to specific websites for such contents, for instance, Amazon for books review, C|net or ZDnet for electronics reviews, and Rottentomatoes for movies reviews. But most returned documents may contain description or specification about the subjects rather than opinions, and going through all reviews is daunting and time-consuming. Therefore, there is a

clear need for a search engine that supports retrieval of documents containing opinions about the topic expressed in a user's query.

One of the central research problems in the area of opinion retrieval is identifying the language features and the lexicon that act as the indicators for the presence of opinion expressed about the query concept. Also, the extraction of subjective contents from a large number of on-line documents becomes a challenging problem in the retrieval of documents containing opinions about a specific topic. In recent years, there has been a large body of research focusing on topical opinion retrieval. Based on the lexicon and linguistic principles, these works use a variety of approaches, such as machine learning (Hurst and Nigam, 2004; Dave et al., 2003), adjective proximity (Skomorowski and Vechtomova, 2007), and feature terms extraction (Yi et al., 2003) to retrieve opinions from the collection of documents. These approaches will be discussed in more detail in the next chapter.

In this thesis, four novel methods are proposed to retrieve documents that express opinions about the given topic. As *Kullback-Leibler divergence* (KLD) of a term can statistically reflect the contribution of the term to discriminate relevant documents from the rest of the documents in the overall collection, KLD scoring is used in all the proposed methods. The retrieval of opinionated documents is a two-stage process. In the first stage (document retrieval), a collection of documents is retrieved in response to the original query using topic-relevance ranking method. This stage is exactly the same for all four proposed methods. And in the second stage (opinion-based re-ranking), the retrieved documents from the first stage are re-

ranked based on the KLD scores of the subjective lexicon present in the documents. The calculations for KLD scores vary in the four proposed methods.

Knowing that people frequently use adjectives to express their subjective opinions (Wiebe, Bruce and O'Hara1999; Bruce and Wiebe 2000), and that KLD measures the closeness of a term towards relevant documents, method 1 calculates KLD scores of subjective adjectives which co-occur with the query terms within the window of 30, and uses the KLD scores in query term weighting for document re-ranking. Taking into account the distance between a query term and a subjective adjective for each instance of co-occurrence, method 2 calculates KLD scores of subjective adjectives based on their distances from the query terms and assigns KLD scores to subjective adjectives according to their positions. Because of the fact that people use particular adjectives to express opinions for a specific category of topics, method 3 calculates KLD scores of subjective adjectives under predefined query categories and assigns KLD scores based on categories that the topic falls into. Further in method 4, collocates, words that co-occur with query terms in the corpus, are used to construct the subjective lexicon automatically instead of using the manually composed list of subjective adjectives. KLD scores of collocates are then calculated and used for document weighting.

Four experiments are conducted corresponding to the methods. The methods are evaluated on TREC corpora. The Blog-06 dataset is used for training our methods, while the Blog-07 datasets is used for testing. The two datasets have the same collection of documents but different topics. KLD scores are calculated based on the Blog-06 relevance judgment results using different proposed methods and then applied for document re-ranking in opinion retrieval

for Blog-07 topics. Experiment results are then compared with the Blog-07 relevance judgment file. In the experiments, a manually constructed list of 1336 subjective adjectives composed by Hatzivassiloglou and McKeown (1997) is used to identify the opinionated contents. In addition, a list of 5319 collocates around the query terms within the window of 30 are extracted to build the subjective vocabulary for method 4. The top 1000 and top 500 KLD scores of collocates are used in document weighting for re-ranking.

The rest of the thesis is organised as follows: Chapter 2 gives a theoretical background of Information Retrieval and a review of related work. Chapter 3 includes the fundamental evaluation measures and the overview of evaluation scheme applied in this study. Chapter 4 presents the detailed document ranking methods developed over the course of this thesis. Chapter 5 describes the experiment data, setup and procedures; Chapter 6 discusses the evaluation results; Chapter 7 concludes the main findings and outlines future research directions.

Chapter 2

Background and Related Work

2.1 Information Retrieval Models

In information retrieval, there are two types of tasks: *ad hoc* and *filtering*. In the ad hoc retrieval (Baeza-Yates and Ribeiro-Neto, 1999), documents in the collection remain relatively static while new queries are submitted to the system. The web search engine, Google, is a typical representation of ad hoc retrieval mode. A similar but distinct task is one in which queries remain relatively static while new documents come into the system and leave. This operational mode is termed as *filtering* (Baeza-Yates and Ribeiro-Neto, 1999). The concrete examples of filtering tasks will be the stock market and news wiring services.

In an ad hoc task, concerned with fetching information from a collection of documents by means of an input query, the effectiveness of the query plays an important role in the performance of the task. Meanwhile, the principle technique of an ad hoc retrieval task is to use a formula to rank the collection of documents and retrieve the ones with high scores.

Typically, in a filtering task, the crucial step is not the ranking of the collection, but the construction of a user profile that truly reflects the user's preferences. The most common approaches for deriving a user profile are based on iteratively collecting relevant information from the user, deriving preferences from the information, and modifying the user profile

accordingly. The information from the user may be a set of keywords the user inputs, or patterns derived from the relevance feedback cycle. Hence, the more information the system collects, the more accurate the information retrieval is.

The three classic information retrieval models proposed over years are Boolean model, Vector Space model, and Probabilistic model. The detailed description of these models will be discussed in the following sections.

2.1.1 Boolean Model

Based on the set theory and Boolean algebra, the Boolean model provides a framework, which is easy to grasp by a common user of an information retrieval system. Generally speaking, the Boolean model considers terms as either present or absent in a document. As a result, the index term weights are assumed to be binary. A query q is usually represented as Disjunctive Normal Form (DNF), linking the index terms by connectives: and, or, not.

The advantage of the Boolean model is its inherent simplicity and neat formalism. However, there are always problems with oversimplification. The drawback of the model is obviously the exact matching -- it can only predict each document to be either relevant or non-relevant, and there is no notion of a partial match to the query conditions. The exact matching is too limited that it may lead to retrieval of too few or too many documents.

2.1.2 Vector Space Model

Given the disadvantage of Boolean model, term index weighting is introduced in Vector Space model to assign non-binary weights to index term in queries and in documents. These weights are ultimately used to compute the degree of similarity between each document stored in the system and the user's query. Thus, unlike Boolean model, Vector Space model supports partial relevance.

By means of algebraic functions of vector, Vector Space model proposes to evaluate degree of similarity with regard to the query q as the correlation between the vectors of documents and queries. As documents and queries are expressed as vectors, the similarity between a document and a query can be easily calculated by cosine value of the two vectors. And then, the Vector Space model ranks documents according to their degrees of similarity to the query.

The substantial improvement in retrieval performance of Vector Space model attributes to its index term weighting scheme and partial matching strategy. Unfortunately, Vector Space model has its limitation of assuming index terms to be mutually independent.

2.1.3 Probabilistic Model

Assuming that relevant documents share similar characteristics, Probabilistic model estimates the probability of relevance of a document to the query based on information, such as term incidence in documents and queries, as well as relevance judgments given by users. Probabilistic model was proposed and extensively researched by Van Rijsbergen (Van Rijsbergen, 1975), Robertson & Sparck Jones (Robertson and Sparck Jones, 1976), Croft &

Haper (Croft and Haper, 1979), and Maron & Cooper (Robertson, Maron, and Cooper, 1982). Meanwhile, relevance feedback was proposed to use relevance judgments to further improve performance of information retrieval. The process of relevance feedback based on Probabilistic model has been heavily used by researchers, aiming to find more relevant documents using their similarities in probabilistic characteristics.

The advantage of Probabilistic models, in theory, is that documents are ranked in decreasing order in terms of the probability of relevance. The main disadvantage of Probabilistic models is the need to guess the initial separation of documents into relevant and non-relevant sets, because of the assumption of ideal answer set, saying that "given a user query, there is a set of documents contains exactly the relevant documents and no others" (Sparck Jones et al., 2000). The Robertson/Sparck Jones probabilistic model is one of the most prevalent and accepted models used in information retrieval. It will be elucidated in more detailed in the later section.

On the whole, the Boolean model is considered to be the weakest model out of the three. Practically, there is some controversy as to whether the Probabilistic model outperforms the Vector Space model. Through several different measures, it has been commonly proven that the Vector Space model is expected to outperform the Probabilistic model with general collections (Baeza-Yates and Ribeiro-Neto, 1999). However, in ad hoc retrieval task, the Robertson/Sparck Jones probabilistic model yielded better performance and the probability ranking principle showed that optimum retrieval quality can be achieved under certain assumptions (Norbert Fuhr, 1992).

2.2 Robertson / Sparck Jones' Probabilistic Model

As mentioned in last section, the Probabilistic model attempted to capture information retrieval problems within a probabilistic framework. Various experiments were conducted, evaluated and analyzed. The Robertson / Sparck Jones' Probabilistic model was proven to be one of the most prevalent and accepted models and has become the base of the Okapi information retrieval system (K. Sparck Jones, S. Walker, and S. E. Robertson, 1998). In this section, an in-depth explanation of the Robertson / Sparck Jones' Probabilistic model will be given.

2.2.1 Assumptions and Fundamental Definitions

Following the widespread convention, formal presentation of the model simply refers to the initial document descriptions as documents D, and to initial request descriptions as queries Q. For convenience, every document may be assumed to be individual and unique. Strictly speaking, relevance is rather the relevance to the query, but the relevance to the information need of the user, which is an expression of the user's request submitted for system searching. Furthermore, relevance is assumed to be binary (a document is either relevant to a query/need or non relevant), which can be attributed to a document without considering any other documents in the system (Sparck Jones et al., 2000).

By giving fundamental definition of relevance above, the general Probabilistic model is actually seeking to retrieve information ground on the probability of relevance. There is a basic question of the Probabilistic model -- "What is the probability that this document is relevant to this query?" (Baeza-Yates and Ribeiro-Neto, 1999) In other words, the basic question can be interpreted as "What is the probability that the document will be judged relevant to the

query/need?" However, the purpose of asking the basic question is to rank documents in order of their probabilistic of relevance. To answer the basic question, for each query, any number of documents has to be ranked, potentially the whole collection. Therefore, information retrieval with Probabilistic model can be treated as a ranking process. This follows the *Probability Ranking Principle* (Robertson, 1977).

The key point about the Probability Ranking Principle is that the probability of relevance to a user's need is not an end in itself, but a means to rank documents on this basis. That is, the retrieval system provides the ranking of documents in the collection, and leaves the user to examine the ranked list from the top, as far as he or she wants to go. Apparently, in the user's point of view, the kernel problem regarding information retrieval systems is the issue of predicting which documents are relevant and which are not. The decision is highly dependent on the ranking of retrieved documents by using term weighting. Therefore, ranking and weighting are critical in the process of attempting to interpret the Probabilistic model. Based on the variations of terms and documents, there are three different sources of information for term weighting: Term Incidence, Term Frequency, and Document Length. By further considering relevance information, Robertson and Sparck Jones (1976) introduced relevance weight.

2.2.2 Term Incidence and Weighting

It is obvious that terms that occur in only a few documents are often more valuable than those that occur in many, and hence are better predictors of relevance. Therefore, it is necessary to determine the contribution of a term's presence in a specific document to that document's probability of relevance from the term's overall incidence (Sparck Jones, Walker, and

Robertson, 1998). It means, the term's contribution will depend on the relation between the number of documents in which it occurs and the number of documents in the file. Thus, *Collection Frequency Weight* is defined as

$$\text{CFW} = \log \frac{N}{n_i} \tag{2.1}$$

Where n_i is the number of documents term t_i occurs, and N is the total number of documents in the collection (Sparck Jones, Walker, and Robertson, 1998). This weight was proposed by Sparck Jones in 1971, and known as *Inverse Document Frequency* (idf). It is based only on the incidence frequency and is applied in the absence of relevance feedback (Sparck Jones, Walker, and Robertson, 1998; Baeza-Yates and Ribeiro-Neto, 1999).

2.2.3 Term Frequency and Weighting

Term frequency is the term's within-document frequency that distinguishes one document containing it from another. In this case, while term's collection frequency is the same for any document, the term frequency may vary -- the more often a term occurs in a document, the more likely it is to be important for that document. So that term frequency for term i, TF_i, is simply defined to be the number of occurrences of term i in the document. Associated with the usual presence weight of term t_i, the resulting formula is

$$W(TF_i) = \frac{TF_i (k_1 + 1)}{k_1 + TF_i} w_i \tag{2.2}$$

Where k_1 determines how much weight reacts to increasing term frequency. In practice, values in the range of 1.2 - 2 were proven to be effective (Sparck Jones, Walker, and Robertson, 1998). This range implies that the effect of term frequency is highly non-linear, for example, after several occurrences of the term, the impact of additional occurrences is minimal.

This method is not only a way of bringing two separate types of information about terms and documents together, but of capturing the significance of different frequencies of terms in a single document in relation to term behavior across the collection: a document has a higher probability of relevance not simply if a term is frequent in it, but is unusually frequent given the number of documents in which it appears (Baeza-Yates and Ribeiro-Neto, 1999).

2.2.4. Document Length and Weighting

It is reasonable that a term, which occurs the same number of times in a short document and in a long document, is likely to be more valuable for the short document. Although from a linguistic point of view, wordiness attributes merely to repetition rather than greater elaboration (Sparck Jones, Walker, and Robertson, 1998), it is still sufficient to equate refinement with prolixity because the topics in retrieval system are fairly general level. Thus, it is appropriate to extend the model interpretation to normalize term frequency by document length. Robertson and Sparck Jones (1995) introduced some uniformity of scaling by relating document length to the length of an average document to ensure that a document of average length will get the same score after normalization. The simple *normalization factor* is defined as

$$\text{NF} = \frac{DL}{\text{AVDL}} \tag{2.3}$$

By adding a tuning constant b, the *mixed normalization factor* would be

$$\text{NF} = ((1\text{-b}) + \text{b}* \frac{DL}{AVDL}) \tag{2.4}$$

Considering the term frequency function mentioned above, after normalization, it becomes

$$W(TF_i) = \frac{TF_i(k_1 + 1)}{k_1 * ((1-b) + b\frac{DL}{AVDL}) + TF_i} w_i \tag{2.5}$$

Replacing $k_1 * ((1-b) + b\frac{DL}{AVDL})$ with K, Formula 2.5 can be simplified as

$$W(TF_i) = \frac{TF_i(k_1 + 1)}{K + TF_i} w_i \tag{2.6}$$

The above formula is known as BM25, which is the best-match weighting function implemented in Okapi (Sparck Jones, Walker, and Robertson, 1998). A value of around b = 0.75 is often used for better results (Sparck Jones, Walker, and Robertson, 1998).

2.2.5 Relevance Information and Relevance Weights

In addition to the attributes of the query terms and documents, such as term incidence, term frequency, and document length, information about whether the documents, in which a term is present, are already actually known to be relevant or non-relevant to the user will lead to more accurate estimation of probability of relevance. That is, the probability of relevance can be estimated based on the known relevance information. Robertson and Sparck Jones (1976) introduced the *term incidence contingency table*, shown in Table 2.1.

	Relevant	Non-relevant	Total
Containing the term	r	n – r	N
Not containing the term	R – r	N – n – R + r	N – n
Total	R	N – R	N

Table 2.1: Term incidence contingency table

With the above relevance information, P_i, the probability of presence of term i in the relevant document set, and $\overline{P_i}$, the probability of presence of term i in the non-relevant document set, can be estimated as

$$p = \frac{r}{R} \qquad \text{and} \qquad \overline{p} = \frac{n-r}{N-R} \tag{2.7}$$

and the *Term Presence Weighting* formula can be rewritten as

$$w = \log \frac{r(N-n-R+r)}{(R-r)(n-r)} \tag{2.8}$$

Where N - the number of documents in the collection,

R - the number of relevant documents in the collection,

n - the number of documents containing term i,

r - the number of relevant documents containing term i

In the absence of relevance information, $\overline{P}$ can be estimated from the proportion of items in the collection that contains the term, that is n/N , based on the assumption that in the context of the entire collection N, the number of relevant documents R is likely to be small (K. Sparck Jones, S. Walker, and S. E. Robertson, 1998).

The small values of the central cells in Table 2.1 result in an infinite weight in Formula 2.8. To avoid this problem, Robertson and Sparck Jones (1997) modified the formula by adding 0.5 to all the central cells. The *Relevance Feedback Weighting* formula is developed to Formula 2.9.

$$RW = \log \frac{(r+0.5)(N-n-R+0.5)}{(R-r+0.5)(n-r+0.5)} \tag{2.9}$$

This weighting scheme gives relatively higher weight to query terms that have a high relevant document incidence and low additional non-relevant document incidence (K. Sparck Jones, S. Walker, and S. E. Robertson, 1998). For example, with the IDF as 4.6 and term incidence as 10046, for the term “coffee”, if the formula is applied by using two values of r, 17 and 2, the one using r=17 will result in RW = 6.3 which is greater than the one using r=2 with RW = 2.6.

2.3 Topical Opinion and Blogs

2.3.1 Topical Opinion

People searching for information on the Internet may have more complex information needs than simply finding documents related to a certain keywords. They may not only be interested in documents that are about a certain topic, but also documents that contain other people's opinions. With the rapid expansion of e-commerce, in order to enhance customer satisfaction, it has become a common practice for the merchants to enable their customers to review and express opinions on the products. Similarly, people can express opinions about other people, such as celebrities and politicians. With more and more users becoming comfortable with expressing their opinions on the Internet, an increasing number of people are writing opinionated content. Consequently, the number of documents with opinionated content on the Internet grows rapidly, and they are spread across different locations, such as news groups, e-commerce websites, individual websites and blogs (or web logs). People looking for opinions on a certain subject have to go to specific websites that might contain such content. For instance, Amazon for reviews of books or Vote Survey for politicians reviews. Alternatively, people have to add subjective words or phrases to their query, such as "reviews" or "opinion". However, only a small portion of documents that express opinions on a topic may contain words such as "reviews" or "opinion".

Therefore, there is an obvious demand for a domain-independent search engine that would support ad hoc retrieval of documents containing opinions on the topic expressed in the user's query. The topical opinions retrieved can contribute to tracking consumer-generated content, brand monitoring, and, more generally, media analysis.

2.3.2 Blogs

Blogs (or web logs) have recently emerged as a new publishing medium. Unlike the traditional media such as newspaper, which only involves a limited number of authors and readers, blogs can reach a wider audience, and with the accessible and easy to use blog-writing software, blogs can be written by anyone with Internet access. Thus, the journal-like blogs have attracted a large number of authors and readers, creating a large subset of the World Wide Web that evolves and responds to real-world events. With the feature that Blogs are created by the authors purely for self-expression, but not intended for any sizable audience, blogs contain a great volume of personal opinions covering all kinds of topics from global events to daily personal life. The opinionated nature of blogs makes blogs a new interesting research area.

2.3.3 Related Work in Topical Opinion Retrieval

There exists a large body of research focusing on sentiment analysis and orientation classification. Based on the lexicon and linguistic principle, these works use a variety of approaches, such as machine learning (Hurst and Nigam, 2004; Dave et al., 2003), adjective proximity (Skomorowski and Vechtomova, 2007), and feature terms extraction (Yi et al., 2003) to retrieve opinion from the collection of documents.

Given that semantically similar words can be identified automatically on the basis of distributional properties and linguistic cues (Brown et al., 1992; Pereira et al., 1993; Hatzivassiloglou and McKeown, 1993), identifying the semantic orientation of words would allow a system to further refine the retrieved semantic similarity relationships. Hatzivassiloglou and McKeown (1997) presented an approach to automatically retrieve semantic orientation

information of adjectives by utilizing indirect information collected from a large corpus (the collection of documents). Because the method relies on the corpus, information extracted is domain-dependent. If the corpus changes, it can automatically adapt to the new domain (Hatzivassiloglou and McKeown, 1997). The result yields 78% accuracy on the sparsest test set up, and 92.37% when a higher number of links is present. These results are significant statistically when compared with the baseline method of randomly assigning orientations to adjectives, or always predicting the most frequent (for types) category (Hatzivassiloglou and McKeown, 1997).

Hurst and Nigam (2004) proposed a lightweight but robust approach to reliably extract polar sentences about a specific topic from a corpus of data containing both relevant and irrelevant text. Hurst and Nigam first assumed that "any sentence that is both polar and topical is polar about the topic in question" (Hurst and Nigam, 2004). And then they approximated the topicality judgment with a statistical machine learning classifier and the polarity judgment with shallow NLP techniques. With the sound underlying assumption, this method results in high-precision identifying the blurry sentences. In the industrial setting, the marriage of topical statement retrieval and the polarity detection of both the document and part of the document have greatly improved the performance of opinion retrieval.

Yi et al. (2003) proposed sentiment analysis to extract positive and negative opinions about specific features of a topic using natural language processing techniques. The sentiment analysis consists of three elements according to its process flow: a topic specific feature term extraction, sentiment extraction and (subject, sentiment) association by relationship analysis.

Their method first determines candidate feature terms based on structural heuristics then narrows the selection using the mixture language model and the log-likelihood ratio (Yi, 2003). A pattern-dependent comparison is then made to a sentiment lexicon gathered from a variety of linguistic resources. This method is run on the review article datasets about digital camera and music review. The result is compared with the collocation algorithm and the best performing algorithm in ReviewSeer, which is by far the best opinion classifier. The result shows that the precision, recall and accuracy of SA are 87%, 56% and 85.6%, while the accuracy of the best algorithm of ReviewSeer is 88.4%.

Dave et al. (2003) proposed and evaluated a number of algorithms for selecting features for document classification by positive and negative sentiment using machine learning approaches (Dave et al., 2003). The classifier was used to identify and classify review sentences from the web, obtaining as good as 76% in accuracy for the review classification task.

Hu and Liu (2004) focused their works on mining opinion and product features that the reviewers have commented on. They developed a method of identifying frequent features of a specific review item, and finding opinion words from reviews by extracting adjectives most proximate to the terms representing frequent features. Their experiment is conducted on customer reviews of five electronic products from Amazon.com and C|net.com with the average recall of 80% and the average precision of 72%.

Skomorowski and Vechtomova (2007) proposed a lightweight method for ad hoc retrieval of documents, which contain subjective content on the topic of the query. In the method,

documents are ranked by the likelihood each document expresses an opinion on a query term, approximated as the likelihood any occurrence of the query term is modified by a subjective adjective (Skomorowski and Vechtomova, 2007). A domain-independent user-based evaluation of the proposed methods was conducted, and the result shows statistically significant gains over Google ranking as the baseline.

2.4 Query expansion

The problem of word mismatch is fundamental to information retrieval. Simply stated, it means that people often use different words to describe concepts in their queries than authors use to describe the same concepts in documents. In the Probabilistic model, attributes of terms, such as term frequency and term incidence, are used to predict the document relevance to the query topic. However, when a query term is not present in the document, its contribution to the document score is zero. Query expansion is created to solve the problem. Query expansion methods find additional discriminative terms and add them to the original query. If the additional terms are related to the query topic, it is likely that the retrieval performance would be improved.

In the methods proposed in this thesis, different approaches were used to extract words (terms) from the collection of judged relevant documents. And then, the KLD scores were calculated based on the terms' occurrences in the two relevance judgment sets, the set of all judged documents and the set of the opinionated documents. However, these extended terms were not added to the original query to form a new query as query expansion does. With the same goal to improve the retrieval performance, these terms ranked by their KLD scores were used to re-rank

the retrieved documents. More details about the methods will be described in Chapter 4 Methodology.

2.5 Term Proximity

Document retrieval functions, such as BM25, have been shown to be highly effective in ad-hoc information retrieval tasks. However, the Probabilistic model is based on term independence assumption that the occurrences of query terms contained in a document are independent of each other. It becomes one of its shortcomings that the proximity of query terms within a document is not taken into account and accordingly same score is given to a document regardless whether the query terms appear close to each other or far apart (Büttcher et al., 2006). Terms in a document are actually dependent on each other because of syntactic and semantic relationships. Because of the fact that query terms appear relatively closer to each other in a relevant document (Büttcher et al., 2006), the dependency of terms, which co-occur in proximity of a certain distance, is helpful in document ranking using the distance information.

Hawking and Thistlewaite (1995) obtained document relevant scores by using proximity relationships by multiplying proximity information score to each occurrence of proximity relationships. The proximity score was defined as $1/\sqrt{S_i - 1}$, where S_i is the span of the instance i of a proximity relationship.

With the proximity information, the longer the proximity relationships span, the smaller the document relevance scores. The results improved the performance by 10% compared with the

commonly used $tf \times idf$ (ft – term frequency within a document; idf – inverse document frequency) document ranking (Hawking and Thistlewaite, 1995).

Rasolofo and Savoy (2003) further added term proximity scores to the Okapi weights. Term proximity is considered as the product of term minimum weight of the co-occurred query terms and the inverse of their distance (Rasolofo and Savoy, 2003). The result showed that with the increase of the distance between the query terms, the importance of the co-occurrence decreased. This method improved the retrieval performance, especially for the collection of fewer documents (Rasolofo and Savoy, 2003).

Previous studies have shown that using proximity information in document ranking can improve retrieval performance. These studies assumed that the closer a set of query terms is, the likely they could indicate the document relevance. Based on the belief that semantically related words are always co-occurring in proximity and the proven performance improvement in document ranking using term proximity, proximity of query term instance to subjective adjectives was used in this study to calculate the document weights.

2.6 Ad hoc Opinion Retrieval

Skomorowski and Vechtomova (2007) proposed a domain-independent method for ad hoc retrieval of documents containing opinions about a given query topic. This approach ranked documents according to the likelihood each document expresses an opinion on a query term, approximating it as the likelihood that the query term occurrences were modified by subjective adjectives (Skomorowski and Vechtomova, 2007).

Assuming that users always want to find opinions about a single entity and such an entity is typically expressed as a noun or noun phrase, adjectives modifying the entity were treated as the indicators of opinion about the entity. The manually constructed list of subjective adjectives by Hatzivassiloglou and McKeown (1997) was used to calculate the probability of a noun at a certain distance from an adjective being the target of that adjective. Instead of applying syntactic parsing at search time to determine whether the query term instance was the target of a subjective adjective in a document, which was computationally expensive, probabilities of subjective adjectives modifying nouns at certain distances were pre-computed. Skomorowski and Vechtomova (2007) introduced a method to calculate the probabilities by using a parsed training corpus with marked adjective targets. As identification of adjective targets in the training corpus and the calculation of probabilities were done before the search time, at the search time, the system only needed to determine the distance between the instance of query term and the nearest subjective adjective, and look up the probability that the adjective modifies a noun at this distance (Skomorowski and Vechtomova, 2007).

To calculate the probability that an adjective modifies a noun at a certain distance, a training corpus with marked adjectives and their targets were needed. SNoW Shallow Paser (Li and Roth, 2001) was used to determine the part-of-speech (POS) of words and boundaries of noun phrases. Skomorowski and Vechtomova (2007) defined a noun to be the target of an adjective when it was the head of the noun phrase that the adjective modified. As the last noun in the noun phrase was assumed to be the head, the probability Pi that a noun was the target of an adjective at distance i was calculated by Formula 2.10 as follows:

$$P_i = \frac{T_i}{K_i} \tag{2.10}$$

Where: T_i - the total number of nouns which are targets of any subjective adjective separated by distance i; K_i - the total number of nouns separated by distance i from a subjective adjective.

Based on Formula 2.10, probabilities for positions of +/-10 words away from an adjective were calculated using the AQUAINT corpus. The results showed that the position immediately following a subjective adjective, i.e. “lucky man”, had the highest probability of containing the target of the adjective. Due to the cases where the target would be the head of a longer noun phrase, the position with the next highest probability was one word away following the adjective, i.e. “excellent basketball players”. The predicative use of adjectives, i.e. “sky is beautiful” made position -2 the third highest probability.

With the probabilities of a noun being modified by adjectives at certain distances, documents were ranked by the likelihood they express opinions on the query terms, which was a document score calculated by “the aggregate probability that the documents refer to occurrences of the query term based on the pre-computed probabilities” (Skomorowski and Vechtomova, 2007).

To evaluate the system, a user-based evaluation with 33 users was conducted. Google search engine was used to measure the number of relevant documents retrieved based on the users’ judgments of each document as “query relevance”, “relevance to a related topic”, or “containing

no opinion". The results showed that the developed method significantly (paired t-test, $P<0.05$) improved performance of topical opinion retrieval over the baseline.

Chapter 3

Evaluation

3.1 Retrieval Evaluation Measures

The most directed functional performance analysis in information retrieval evaluation is to evaluate how precise the answer set is (Turpin and Hersh, 2001). This type of evaluation is known as retrieval performance evaluation. The evaluation is usually based on a test reference collection and on an evaluation measure (Baeza-Yates and Ribeiro-Neto, 1999). The test reference collection consists of a collection of documents, a set of queries, and a set of relevant documents provided by specialists for each query. Given a retrieval strategy S, for each query, the evaluation measure quantifies the similarity between the set of documents retrieved by S and the set of judged relevant documents. This provides an estimation of the goodness of the retrieval strategy S.

3.1.2 Recall and Precision Measures

Given a query Q and its set R of relevant documents, by using an evaluation strategy, a set A is generated as the answer set for the query Q. Let |R| be the number of relevant documents in R, |A| be the number of retrieved documents in A, and |Ra| be the number of documents in the intersection of the sets R and A. *Recall* is defined as the fraction of relevant documents which

have been retrieved, and *Precision* is defined as the fraction of retrieved documents which are relevant. The formulae are showed as follow:

$$\text{Recall} = \frac{|Ra|}{|R|} \tag{3.1}$$

$$\text{Precision} = \frac{|Ra|}{|A|} \tag{3.2}$$

Recall is usually used to measure the effectiveness of a retrieval system (Baeza-Yates and Ribeiro-Neto, 1999). For example, if there were 10 relevant documents in total, strategy A that retrieves 8 relevant documents (Recall = 0.8) is considered to be more effective than strategy B that only retrieves 5 relevant documents (Recall = 0.5). Precision is usually used to measure the efficiency of a retrieval system (Baeza-Yates and Ribeiro-Neto, 1999). A retrieval system with higher precision could fetch relevant documents to users faster than a retrieval system with lower precision.

3.1.2 Standard Evaluation Measures

After the relevance judgments have been made, the performance of each run is evaluated by the comparison between the experiment results and relevance judgments results according to standard measures. The most frequently used measures are precision at various document cutoff points, such as precision at 10 retrieved documents (P@10), precision after R relevant documents are retrieved (R-Prec), Mean Average Precision (MAP), and Binary Preference (bpref).

3.1.2.1 Precision at 10 (P (10))

Precision at different document cutoff values n is the mean precision values when n relevant documents are retrieved. It is calculated according to Formula 3.3.

$$\Pr ec(r) = \frac{n(r)}{r} \tag{3.3}$$

Where n(r) – the number of relevant documents retrieved up to and including rank r.

P (10) counts the number of relevant documents in the top 10 documents in the ranked list returned for a topic. This measure is closely correlated with users' satisfaction in tasks, as high precision in top 10 documents can undoubtedly increase users' satisfaction with system (Voorhees and Buckley, 2004). However, P (10) has a much larger margin of error associated with it than other measures. Therefore, it is not a powerful discriminator among retrieval methods and averages poorly.

3.1.2.2 R-Precision (R-prec)

R-Precision is the precision after *R* documents are retrieved where *R* is the number of relevant documents for the given topic. It has a much smaller margin of error than P (10), though a larger error than MAP. Baeza-Yates and Ribeiro-Neto (1999) claimed that it is useful to observe behaviors of a strategy for each individual query in an experiment.

3.1.2.3 Mean Average Precision (MAP)

To evaluate the retrieval performance for strategies that run for several distinct queries, *Average Precision*, the average of the precision for a query at each recall level, is calculated according to Formula 3.4.

$$AvgP = \sum_{i=1}^{N} \frac{\Pr ec(r)}{N} \tag{3.4}$$

Where N – the number of relevant documents retrieved for the topic; r – the rank of the retrieved relevant documents; Prec(r) – the number of relevant documents retrieved up to and including rank r calculated as in Formula 3.3

Assuming the precision for relevant documents that are not retrieved to be zero, average precision is basically the mean of precision scores obtained after each relevant document is retrieved (Voorhees and Buckley, 2000). Average precision is based on much more information than either P (10) or R-precision, therefore it is a more powerful and more stable measure. *Mean Average Precision (MAP)* is calculated as the mean of all average precision values for a set of queries used in the evaluation.

3.1.2.4 Binary Preference (bpref)

Since the scores for P (10), R-precision, and MAP are completely determined by the ranks of the relevant documents in the result set, these measures make no distinction in pooled collections between documents that are explicitly judged as non-relevant and documents that are assumed to be non-relevant because they are not yet judged (Buckley and Voorhees, 2004).

Therefore, *binary preference (bpref)* was introduced to measure the effectiveness of a system on the basis of judged documents by depending only on the absolute number of relevant and/or judged non-relevant documents. It is called "bpref" because it uses binary relevance judgments to define the preference relation, in which any relevant document is preferred over any non-relevant document for a given topic (Buckley and Voorhees, 2004).

Consequently, bpref measures the average number of times non-relevant documents are retrieved before relevant documents. For a topic with R relevant documents, bpref is calculated according to Formula 3.5.

$$bpref = \frac{1}{R}\sum_{r} 1 - \frac{\left| n \quad ranked \quad higher \quad than \quad r \right|}{R} \tag{3.5}$$

Where r is a relevant document and n is a member of the first R judged non-relevant documents retrieved by the system. For example, a bpref of 80% means that 20% of non-relevant documents are ranked above relevant documents, while a bpref of 100% means that all relevant documents appear before all non-relevant documents.

3.2 TREC and Blog Track

Text Retrieval Conference (TREC), co-sponsored by the National Institute of Standards and Technology (NIST) and U.S. Department of Defense, is an annual international conference for researchers in information retrieval areas to report their latest works and exchange ideas. For each track of TREC, NIST provides a collection of documents and a set of user queries (topics). The participating research groups run their own retrieval systems on the document collection

for the given query topics, and return to NIST a list of the retrieved top-ranked documents. NIST hires evaluators to judge the relevance of the top retrieved documents, and evaluates the results. TREC provides the infrastructure necessary for large-scale evaluation of text retrieval methodologies. In every annual cycle, TREC consists of a set of tracks (retrieval tasks), such as High Accuracy Retrieval from Documents (HARD), Question Answering, Enterprise, Blog etc.

3.2.1 TREC Blog Track

TREC began the Blog track from 2006. It aimed to explore the information seeking behaviors in the blogs. For this purpose, a new large-scale test collection, namely the TREC Blog06 collection, has been created and used for 2006 – 2008 TREC tasks. In the first pilot run of the blog track, there were two tasks, the main task (opinion retrieval) and an open task. The opinion retrieval task focuses on a specific feature - the opinionated nature of many blogs (Iadh Ounis et al., 2006). The open task was introduced to allow participants the opportunity to influence the determination of a suitable second task for other features of blogs.

3.2.2 Blog-06 Test Collection

The collection of blogs, called Blog-06, was created by the University of Glasgow. It included a selection of "top blogs" provided by Nielsen BuzzMetrics (Nielsen BuzzMetrics, 2005) and supplemented by the University of Amsterdam. The University of Glasgow monitored the resulting 100,649 blog feeds over a period of 11 weeks from December 2005 to February 2006. During the period, XML feeds and the corresponding permalink documents were fetched and saved. The total number of permalink documents used in the TREC 2006 Blog Track is over 3.2 million. Furthermore, the blogs in different topic categories accessible to the TREC assessors

were covered. Topics included news, sports, politics, health, etc. Table 3.1 shows the detailed statistics of the final Blog-06 test collection.

Quantity	Value
Number of Unique Blogs	100,649
RSS	62%
Atom	38%
First Feed Crawl	06/12/2005
Last Feed Crawl	21/02/2006
Number of Feeds Fetches	753,681
Number of Permalinks	3,215,171
Number of Homepages	324,880
Total Compressed Size	25GB
Total Uncompressed Size	148GB
Feeds (Uncompressed)	38.6GB
Permalinks (Uncompressed)	88.8GB
Homepages (Uncompressed)	20.8GB

Table 3.1: Details of the Blog-06 test collection and its corresponding statistics

In addition, given the particular severity of spam in the blogs, a selection of assumed spam blogs was added to the collection to ensure that blog track participants had a realistic research setting (Iadh Ounis et al., 2006). This created a simulated "real world" environment for evaluation.

3.2.3 Opinion Retrieval Task

In the TREC 2006 blog track, the opinion retrieval task is to search for blog posts that express opinions about a given target. The target can be a "traditional" named entity, i.e. a name of an organization or a person, a concept, i.e. a type of technology, a product name, or an event (Iadh Ounis et al., 2006). Given the described topics, the retrieval task can be summarized as "What do people think about X", where X is the target. In this case, the topic of the post is not necessary to be the same as the target, but an opinion about the target has to be present in the post or one of the comments to the post (Iadh Ounis et al., 2006). To be more specific, the

opinion retrieval task is to identify and rank the blog posts expressing an opinion regarding a given topic.

In the opinion retrieval, only retrieving blog posts that contain the query terms is far from correct. It must uncover "explicit expression of opinions or the public sentiment towards the given topics showing a personal attitude, and exclude the ones that give formal definitions of the topics" (Iadh Ounis et al., 2006). Thus, to retrieve blog posts that contain opinions about the topics becomes the major challenge in opinion retrieval task.

3.2.4 Topics

A TREC topic is "a natural language statement of the information need" (D. Harman, 1994). The Blog-06 topics were selected by NIST from a donated collection of queries sent to commercial blog search engines during the time Blog-06 test collection was being collected (Iadh Ounis et al., 2006). The final 50 topics used in the opinion retrieval task followed the structure used in previous TREC test collections. Each topic consists of fixed segments, such as topic number, title, description, and narrative. An example of a topic is shown in Figure 3.1.

```
<top>
  <num> Number: 871

  <title> cindy sheehan

  <desc> Description:
  What has been the reaction to Cindy Sheehan and the
  demonstrations she has been involved in?

  <narr> Narrative:
  Any favorable or unfavorable opinions of Cindy Sheehan are
  relevant. Reactions to the anti-war demonstrations she has
  organized or participated in are also relevant.
</top>
```

Figure 3.1: Blog track 2006, opinion retrieval task, topic #871

The title fields of the topics directly came from the literal queries from the search query logs file (Iadh Ounis et al., 2006). Based on the titles, NIST assessors developed interpretations of what the searchers were originally looking for when the queries were submitted. And then, the assessors searched the Blog-06 test collection to see if blog posts with relevant opinions appeared in the collection and recorded the explanations of queries in the description and narrative fields of the topics (Macdonald and Ounis, 2006).

3.2.5 TREC Evaluation

Given the results file and a standard set of judged results, TREC_EVAL is a program designed to evaluate TREC results using the standard NIST evaluation procedures. TREC_EVAL has become the primary method for evaluating retrieval results (Voorhees and Buckley, 2000). There are two major input parameters in TREC_EVAL – the experiment results file and the TREC judgment results file. The experiment results file follows the format of "Topic_id Q0 docno rank sim runtag" delimited by spaces, where Topic_id is the three digit topic number, (i.e. "851"); Q0 is an integer constant (i.e. "0"); docno is the permalink document number that can be found between <DOCNO> and </DOCNO> tags; rank is the rank at which the system returned the document, and is required to be an integer from 1 to 1000; sim is the system's similarity score; run_id is the run's identifier string.

TREC_EVAL calculates the standard measures of system effectiveness by comparing the experiment results file and the relevance judgments file. It outputs values for different evaluation measures, such as MAP, GMAP, R-Prec, bprec, Precision at n documents (n=5, 10, 15 …) etc for each topic, as well as averaged results for all topics. Figure 3.2 shows an example

of the TREC_EVAL output for 50 queries. Therefore, the results of TREC_EVAL become the indicator for the opinion retrieval performance.

num_q	all	50
num_ret	all	39782
num_rel	all	12187
num_rel_ret	all	8218
map	all	0.3805
gm_ap	all	0.2938
R-prec	all	0.4097
bpref	all	0.4361
recip_rank	all	0.7962
ircl_prn.0.00	all	0.8894
ircl_prn.0.10	all	0.7084
ircl_prn.0.20	all	0.6156
ircl_prn.0.30	all	0.5707
ircl_prn.0.40	all	0.4936
ircl_prn.0.50	all	0.4092
ircl_prn.0.60	all	0.3136
ircl_prn.0.70	all	0.2254
ircl_prn.0.80	all	0.1511
ircl_prn.0.90	all	0.0963
ircl_prn.1.00	all	0.0213
P5	all	0.6760
P10	all	0.6900
P15	all	0.6760
P20	all	0.6410
P30	all	0.6013
P100	all	0.4932
P200	all	0.4151
P500	all	0.2670
P1000	all	0.1644

Figure 3.2: An example of TREC_EVAL output

In the example of TREC_EVAL, it calculates the total number of queries (num_q), total number of documents retrieved over all queries (num_ret), total number of relevant documents over all queries (num_rel), total number of relevant documents retrieved over all queries (num_rel_ret), MAP, Average Precision with Geometric Mean (gm_ap), R-prec, bpref, reciprical rank of top relevant document (recip_rank), Interpolated Recall - Precision Averages at k recall (ircl_prn.k, k = 0.00, 0.10, 0.20, …), and Precision at n documents (P@n, n=5, 10, 15 …).

3.2.6 TREC Approach Evaluation Vs. User-based Evaluation

Retrieval performance evaluation using TREC approach evaluation is a laboratory methodology. That means, the results are repeatable if the test collection, including topics, documents and relevance judgments, remain exactly the same. TREC approach evaluation tries to simulate the activity of real users in the controlled environment (Turpin and Hersh, 2001). With the fixed topics, documents and relevance judgments, researchers can mimic the "real world" retrieval system for large-scale test collections, but don't need to worry about any human factors involved. Because of the stability of TREC approach evaluation, researchers can also tune the parameters in the retrieval system, monitor the changes in evaluation measures, and compare the effects of various parameters. Unlike user-based evaluation, in which users judge the set of documents one by one, TREC approach evaluation is a batch process, which compares thousands of retrieved documents with the relevance judgments file collectively as a whole and produces evaluation measures result by only one operation. Accordingly, the TREC approach evaluation can be completed in a short period of time. Therefore, TREC approach evaluation is always selected as the primary method for retrieval results evaluation (Voorhees and Buckley, 2000).

However, real world searching is more complex and human factors have to be taken into account. How real users judge a particular document for a specific topic varies because of their individual perspectives. Each user judgment for a specific document is unique. User-based evaluation methodology, which relies on the participation of real users in the experiment, is appropriate for designated tasks without a suitable test collection. Though user-based evaluation yields more realistic data about the system, only a relatively small number of users performing

limited number of tasks can be studied. Because of a large amount of time spent in recruiting users, conducting experiments, and analyzing results, it is more expensive than the TREC approach. Also, the results are unrepeatable and unstable because human factors are uncontrollable and most experiments are entirely unreplicable. A user may judge a document relevant now, but non-relevant two hours later, because of subtle sentiment changes. There are also many uncontrollable variables involved in opinion retrieval in practice.

3.3 Blog Track Assessment Procedure

There are 50 provided topics in Blog-06 Blog Track. Participants were allowed to submit up to five runs, including a compulsory automatic run using the title-only field of the topic. The track organizers encouraged participants to submit manual runs, since they are valuable for improving the quality of the test collection (Iadh Ounis et al., 2006). Each submitted run consisted of the top 1,000 opinionated documents retrieved for each topic. The retrieval units were the documents from the permanent links, to which Blog posts and comments related. Permalink is the essential element in opinion retrieval task, as most Blogs change regularly, and without a permanent link, the posts would be impossible to find later.

After the submission, NIST organized the assessments for the opinion retrieval task. The relevance judgment of a document for a topic was only made by one assessor. Given a topic and a blog post, assessors were asked to judge the content of the blog post, which includes the content of the post itself and the contents of all comments to the post (Iadh Ounis et al., 2006). That is, if the relevant content was in the contents of blog post or in the contents of one of the comments, then the permalink is declared to be relevant.

Before assessing the blog post, NIST gave all assessors a working definition of subjective or opinionated content associated with a number of examples to assure they understood that a post has a subjective content. By the definition from NIST, a post has a subjective content if it contains an explicit expression of opinion or sentiment about the target, showing a personal attitude of the writer, rather than attempting to provide a formal definition about the target (Iadh Ounis et al., 2006). Take the NIST sample example of "Skype" for instance. Figure 3.3 shows an excerpt from an opinionated blog post, which contains explicit expression of opinion.

Skype 2.0 eats its young

The elaborate press release and WSJ review while impressive don't help mask the fact that, Skype is short on new ground breaking ideas. Personalization via avatars and ring-tones. . . big new idea? Not really. Phil Wolff over on Skype Journal puts it nicely when he writes, "If you've been using Skype, the Beta version of Skype 2.0 for Windows won't give you a new Wow! experience."

Figure 3.3: An excerpt from an opinionated Blog post

And Figure 3.4 shows an excerpt from an unopinionated post, which only gives functionalities of Skype without any personal attitude.

Skype Launches Skype 2.0 Features Skype Video

Skype released the beta version of Skype 2.0, the newest version of its software that allows anyone with an Internet connection to make free Internet calls. The software is designed for greater ease of use, integrated video calling, and . . .

Figure 3.4: An excerpt from an unopinionated Blog post

Based on the assessors' knowledge, the assessments were asked to determine the documents by a 5-point scale as follows:

-1 *Not judged*. The content of the post was not examined due to offensive URL or header (such documents do exist in the collection due to spam).

0 *Not relevant*. The post and its comments were examined, and do not contain any information about the target, or refer to it only in passing.

1 *Relevant*. The post or its comments contain information about the target, but do not express an opinion towards it.

2 *Relevant - Negative Opinionated*. It contains an explicit expression of opinion about the target, showing some personal attitude of the writer(s), and the opinion expressed is explicitly negative about, or against, the target.

3 *Relevant - Mixed Opinionated*. Same as (2), but contains both positive and negative opinions. Posts with ambiguous, mixed, or unclear opinions were also considered as this scale.

4 *Relevant - Positive Opinionated*. Same as (2), but the opinion expressed is explicitly positive about, or supporting, the target.

To make sure the assessors have a clear definition for each category, a number of examples were given to illustrate the various evaluation labels above. After the assessment was complete, the relevance judgments file was formed.

Chapter 4

Methodology

4.1 Introduction (Hypotheses)

The basic research question of the project is whether the use of *Kullback-Leibler Divergence* (KLD) can improve performance of opinion retrieval. To answer this question, we first aimed at investigating different ways of using KLD to weight subjective adjectives around the query terms within the window of 30 words, as subjective adjectives have been proven to have positive correlation with subjectivity (Wiebe, Bruce and O'Hara1999; Bruce and Wiebe 2000). We used a manually constructed list of 1336 adjectives composed by Hatzivassiloglou and McKeown (1997) to identify the opinionated contents. In addition, to further extend the subjective lexicon from manually constructed list to automatic one, collocates of query term, which are words around the query term within the window of 30 words (30 words either side of the query term instance), were also considered. Over the course of the project four main hypotheses were examined and tested.

Initially, assuming that KLD scores of subjective terms can statistically reflect the "goodness" of discrimination of terms towards the opinionated documents, we hypothesized that document ranking based on KLD scores of subjective adjectives can lead to performance improvement in opinion retrieval. The formal hypothesis statement is as follows:

Hypothesis 1: Document ranking using KLD scores of subjective adjectives results in performance improvement in the opinion retrieval task compared to baseline systems.

Based on the premise that topical subjectivity is mainly expressed by adjectives and KLD weighting function can determine terms that have made relevant documents divergent to the rest of the documents in the overall collection, we computed KLD scores of subjective adjectives by applying KLD weighting function to the subjective adjectives. As some adjectives are more likely to appear in relevant documents than non-relevant documents, KLD scores of subjective adjectives would be helpful to retrieve opinionated relevant documents. The detailed description of proposed method to explore and test this hypothesis is given in Section 4.5.1.

Next, assuming KLD scores of subjective adjectives associated with position information could add distance information to KLD scores of subjective adjectives, we hypothesized that KLD scores based on different distances in the proximity of query term instance to each subjective adjective within the window of 30 may be helpful for opinion retrieval. The formal hypothesis statement is as follows:

Hypothesis 2: Document ranking using KLD scores of subjective adjectives calculated for each distance from a query term results in performance improvement in the opinion task compared to baseline systems.

As distances between the query term and subjective adjectives around it within the window of 30 may be indicative of the "likelihood" of subjective adjectives expressing opinion about the query term, by adding the position information to the original KLD scores, we hope to achieve better performance in opinion retrieval. Section 4.5.2 presents the detailed method proposed for this hypothesis.

Furthermore, we moved on to investigate whether KLD scores of subjective adjectives computed according to different topical categories can lead to performance improvement in opinion retrieval. As people use a variety of wordings to express opinions regarding a particular topic, we hypothesized that KLD scores of subjective adjectives corresponding to different topic categories may benefit opinion retrieval. The formal hypothesis statement is as follow:

Hypothesis 3: Document ranking using KLD scores of subjective adjectives calculated for each topic category results in performance improvement in the opinion retrieval task compared to baseline systems.

By assuming that particular words are used to express opinions in a specific topic category more often than others, KLD scores of subjective adjectives were computed according to manually characterized and selected topic categories. The experiments were conducted separately corresponding to topic categories. The detailed description of the proposed method is discussed in Section 4.5.3.

In addition, besides the manually constructed lexicon (a dictionary of terms used for different subjects), i.e. the fixed list of 1336 adjectives composed by Hatzivassiloglou and McKeown (1997), we took into account subjective lexicon, which can be automatically extracted from the collection of documents. Collocates of query terms were used to construct the lexicon instead of the manually constructed list of 1336 adjectives. We hypothesized that KLD-weighted collocates may lead to performance improvements in opinion retrieval tasks. The formal hypothesis statement is as follows:

Hypothesis 4: Document ranking using KLD scores of collocates of query terms results in performance improvement in the opinion retrieval task compared to baseline systems.

Collocates of query terms were automatically selected from words around the query terms within the window of 30. We hope that the proximity of query terms to collocates extracted from relevant documents could assist in retrieving more relevant documents contain subjective information about the query topic.

4.2 Baseline Methods Selection

4.2.1 BM25 & Parameters

Robertson et al. (1994) first introduced the BM25 weighting formula and demonstrated that it worked very well in Okapi at TREC-3 (Robertson et al., 1994). Since then, BM25 has become a popular choice for scoring document relevance based on term frequency, document length, and

other collection statistics. Over the last decade, BM25 has consistently produced good results in TREC evaluations (Fan et al., 2004) and has been widely used in many research experiments. Therefore, it is recognized as the "strong" baseline in retrieval evaluation (Roussinov and Fan, 2006). Due to its effectiveness in prior TREC evaluations, BM25 is used as one of the baselines in this study.

The BM25 weighting formula consists of two tuning parameters: k_1 and b, where k_1 determines how much term weight reacts to increasing term frequency and b is the constant tuning factor, controlling the effect of document length on term weight. In practice, $k_1$values in the range of 1.2 - 2 are proven to be effective, and experiments in previous TRECs suggest the value of b around 0.75 gives better results.

However, both parameters are query- and collection-dependent (Sparck Jones, et al., 1998). The best values of b and k_1 may fluctuate corresponding to the test collections. The smaller values for b may work effectively for particular queries (Robertson and Walker, 1999). Bennett et al. (2008) also pointed out that if relevance information were available in advance, it would be possible to tune the BM25 function to further increase retrieval evaluations, such as MAP and P@10.

Believing that values of b and k_1 giving the best retrieval performance are query- and collection- dependent, a preliminary experiment was run to tune the two parameters. Based on the Blog-06 documents collection and query terms, we retrieved the top 1000 documents that were ranked by Okapi BM25 weighting function, with b = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.75 and

k_1= 0.5, 0.75, 1, 1.2, 1.5, 1.75, 2, 2.5. TREC_EVAL was then used to evaluate the retrieved documents, comparing the Blog-06 relevant judgment documents. Retrieval measures, such as MAP, R-Prec, bpref, and P@10 were then used to evaluate the results, which used different combinations of b and k_1 for the experiment. Table A.1 in Appendix A shows the topical relevance evaluation results with different values of b and k_1 using measures of MAP, R-Prec, bpref, and P@10.

Because MAP is the most significant measure among the four, the highest score in MAP should indicate the best value of b and k_1 for BM25 in the Blog-06 test collection. Based on the result, it shows that b = 0.1 and k_1= 1.75 gave the highest score in MAP. As shown in Table A.1 in Appendix A, b = 0.1 and k_1= 1.75 also scored the highest in R-Prec and bpref.

To evaluate the opinion retrieval performance, TREC_EVAL was also used to evaluate the retrieved opinionated documents, comparing the Blog-06 relevant judgment documents. Table A.2 in Appendix A shows the opinion relevance evaluation results with different values of b and k_1 using measures of MAP, R-Prec, bpref, and P@10. Similarly, MAP had the highest score with b = 0.1 and k_1= 1.75. Based on the tuning experiment, we decided to use b = 0.1 and k_1= 1.75 in this study, because they are more appropriate for Blog-06 documents collection than the original documents collections used to evaluate BM25, or previous TREC documents collections.

4.2.2 Proximity-based Method

With the goal to rank documents containing opinions about the concept expressed in the query, Vechtomova (2007) developed a novel method to calculate document scores based on the proximity of the query term instance to subjective adjectives within the window surrounding the query term.

This method used a two-staged process to retrieve opinions from blogs. The first stage used BM25 to retrieve the top 1000 documents corresponding to the query term for initial document retrieval. With the hypothesis that subjective adjectives occur within a fixed-size windows around query term instances in a document was useful feature for finding opinion about the query term, in the second stage, the set of top 1000 documents retrieved from the first stage was re-ranked using the opinion-finding method, in which a list of 1336 subjective adjectives manually constructed by Hatzivassiloglou and McKeown (1997) was used to calculate the *weighted term frequency (wf)*. The wf was calculated by the Formula 4.1 and 4.2.

$$c(t_i) = \begin{cases} 1 + \sum_{j=1}^{|A|} \frac{1}{distance(t_i, a)^P} & if\ |A| > 0 \\ 1 & otherwise \end{cases} \quad (4.1)$$

Where: $c(t_i)$ – the contribution of the i^{th} instance of the query term t occurring in the document; $distance(t_i,a_j)$ – distance in number of non-stop words between the i^{th} instance of the query term t and subjective adjective a; p–constant, moderating the effect of the distance between t_i and a; A – number of subjective adjectives within the span S before and after t_i.

$$wf_t = \sum_{i=1}^{N} c(t_i) \quad (4.2)$$

Where: N – the number of instances of query term t in the document. After wf was calculated for a query term, its term weight in the document was calculated in the same way as in the BM25 formula, using wf instead of tf. The term weight was calculated by the Formula 4.3.

$$TW_t = \frac{(k_1 + 1) \times wf_t}{k_1 \times NF + wf_t} \times idf_t \qquad (4.3)$$

Where: k_1– the term frequency normalization factor, which moderates the contribution of the weight of frequent terms. NF – the same formula as in BM25. The value of k_1= 1.2 is based on the standard BM25 optimal setting based on TREC results.

In this proximity-based method, the results suggested that the presence of subjective adjectives close to any word from the query was a useful indicator of opinions expressed about the query concept (Vechtomova, 2007). To determine whether to treat a query as a phrase or single term in initial documents retrieval (the first stage) and opinion-based re-ranking (the second stage), experiments with different types of queries constructed from the topic titles, i.e. phrases and single terms, in different stages were conducted. The results demonstrated that better performance was obtained by using phrases in the initial documents retrieval (the first stage) and using single terms in the opinion-based re-ranking (the second stage). Take the query "mashup camp" (topic 925) as an example. Using "mashup camp" as a phrase in the first stage can bundle "mashup" and "camp" together to avoid unrelated documents containing two single terms, "mashup" and "camp". While in the second stage, treating "mashup" and "camp" as two

single terms is likely to be more effective than a phrase, because people may use the head of a phrase to refer to the whole phrase in the later context. For instance, having said that "the mashup camp is very exciting", people may use a single term "camp", which is the head of the phrase "mashup camp", to refer to this concept in the later development.

4.3 Subjective Adjectives

To determine whether a document contains opinions about a given topic, recognition of topical subjectivity in the document is needed. Topical subjectivity, by definition, must contain subjective contents towards the given topic. We assume that users normally want to find opinions about a single entity, such as a name of product, person, company, location or activity, etc. Such an entity is typically expressed as a noun, a noun phrase or a gerund (a noun derived from a verb, with -ing suffix). Because adjectives act as directed noun modifiers in noun phrases or in the context of nouns (Baker, 2003) and adjectives are positively correlated with subjectivity (Wiebe, Bruce and O'Hara1999; Bruce and Wiebe 2000), adjectives, to some extent, can be treated as the indicators of subjective contents towards the noun concept. Based on a statistical analysis of the assigned semantic classifications, Bruce and Wiebe (2000) provided support to show that adjectives are statistically significantly and positively correlated with subjectivity. Also, Wiebe (2000) showed that the presence of adjectives is useful for predicting subjectivity. Though the role played by adjectives may vary widely between languages, adjectives are considered as one of the major ways of expressing value judgment in English (Dixon and Aikhenvald, 2004).

In English, the most frequent usages of adjectives are attributive or predicative (Greenbaum, 1996). Attributive usage is where a noun is immediately modified, typically premodified (i.e., “a beautiful portrait”), while predicative usage links the adjective to the subject with a copular verb such as "be" (i.e. "the portrait is beautiful"). The less frequent usages of adjectives include objective complements of verbs, such as "make" and "let" (i.e. "made the portrait beautiful"), resultative secondary predicates (i.e. "paint the portrait beautiful"), and degree phrases (i.e. “as beautiful as the portrait” (Rijkhoek, 1998; Baker, 2003). As shown in the example, “beaufiful” clearly expresses positive opinion from the author about the “portrait” with all the usages as an adjective.

Lexicons of subjective adjectives can be created manually or automatically by various approaches. Yi et al. (2003) extracted 2500 subjective adjectives using machine learning techniques automatically. Based on information extracted from conjunctions between adjectives in a large corpus, Hatzivassiloglou and McKeown (1997) manually composed a list of 1336 subjective adjectives, including 657 positive and 679 negative adjectives. The list of 1336 adjectives created by Hatzivassiloglou and McKeown was used in this study. There also exist many automatic methods for subjective adjective mining that can be used instead (e.g. Wiebe, 2000; Turney, 2002).

4.4 Kullback-Leibler Divergence

There exist many approaches to discriminate relevant from non-relevant documents based on the distribution of appropriate terms, terms that appear more in relevant documents than in the whole collection. Based on the assumption that the difference between the distribution of terms

in a collection of relevant documents and the distribution of the same terms in the whole document collection is an indicator of semantic difference (Carpineto et al., 2003), appropriate terms are more likely to appear in relevant documents than non-relevant documents. The discrimination between good and bad terms in query expansion based on distribution analysis has been extensively investigated (van Rijsbergen, 1977; van Rijsbergen et al., 1978)

Kullback-Leibler (KL) divergence weighting function based on the appropriateness of a term by distribution analysis was proven to be a computationally simple and theoretically justified approach to weight candidate terms (Carpineto et al., 2003). Initially, KL divergence was a measure in statistics to quantify how close a probability distribution P is to a model (or candidate) distribution Q (Cover and Thomas, 1991). On top of the KL divergence measure, Carpineto et al. (2003) proposed a probabilistic method to assign scores to feedback terms based on their distribution in the pseudo relevant documents (R) and in the documents in the whole collection (C). Using the terms retrieved from the pseudo-relevant documents, the *Kullback-Leibler divergence (KLD) score* for each term was computed by the term-scoring function shown in Formula 4.5:

$$\text{KLDscore}(t) = [P_R(t)] * \log[P_R(t)/P_C(t)] \quad (4.5)$$

Where: $P_R(t)$ - the probability of the term t in the relevant (pseudo-relevant) document set; $P_C(t)$ - the probability of the term t in the whole collection.

The KLD weighting function was able to determine terms that have made the relevant documents divergent to the rest of the documents in the overall collection. A good discriminative term must have a great contribution to the difference between relevant documents and the rest of the collection. KLD scores were introduced to statistically reflect the "goodness" of discrimination of a term towards the collection of documents. A smaller KLD score of a term means that occurrence probability of the term in relevant document is closer to the occurrence probability of the term in the whole collection, that is, the term is not a discriminative one to distinguish relevant documents from the whole collection. While a larger KLD score for a term means that occurrence probability of the term in relevant document is either much bigger or smaller than that of the term in the whole collection, , showing that the term is discriminative. Apparently, in most cases, the KLD scores are fairly small, as most terms are indistinctive in a large collection of documents.

In previous studies in information retrieval, KLD scoring and variants of the KLD weighting function have been used for selecting topic models for queries (Xu and Croft, 1999), for identifying relevant sentences (Kolla and Vechtomova, 2007) and for modeling term weight as deviation from randomness (Amati and van Rijsbergen, 2002).

In our methods, we computed the KLD scores of subjective adjectives in the top 1000 initially retrieved documents for each topic in the training dataset (Blog-06 documents collection). We then normalized the KLD scores of subjective adjectives, ranked the list of subjective adjectives by their normalized KLD scores and selected the adjectives with positive normalized KLD

scores for term weighting. A detailed description of the proposed methods will be given in Section 4.5.

4.5 Proposed Methods

In this study, we propose four approaches to find documents that contain opinions about a specific topic represented by query term(s). These approaches are two-stage processes. In the first stage (initial documents retrieval), a collection of top 1000 documents was retrieved in response to the original query using topic-relevance ranking method. This stage was exactly the same for all four approaches. And in the second stage (opinion-based re-ranking), the retrieved documents from the first stage were re-ranked based on the aggregate of KLD scores of subjective adjectives occurrences. The KLD scores were calculated based on the collection of pseudo-relevant documents using different proposed methods. The detailed description of these methods will be discussed respectively in Sections 4.5.1, 4.5.2, 4.5.3, and 4.5.4. The calculations for KLD scores are different for the proposed methods.

For all methods, in the first stage (documents retrieval), the top 1000 documents were retrieved using BM25 in response to the original query terms. Based on the documents relevance results comparison between single terms and phrases done by Vechtomova (2007), the best performance was obtained using phrases to retrieve the initial document set, and single terms for the subjectivity-based re-ranking. Therefore, in this study, query terms enclosed in quotes by the user were treated as phrases in the first stage, for example, “winter olympics” was treated as one single query term instead of “winter”, “olympics”, as well as “Steve jobs” instead of

"Steve", "jobs". In the second stage, the top 1000 retrieved documents from the first stage were re-ranked using the following proposed methods.

4.5.1 Method 1 – KLD Scores of Subjective Adjectives

Knowing that people frequently use adjectives to express their subjective opinions, and that KLD measures the closeness of a term towards relevant documents, KLD scores of subjective adjectives can statistically reflect the "goodness" of the adjectives' discrimination towards the opinionated documents. As some adjectives are more likely to appear in relevant documents than non-relevant documents, KLD scores of subjective adjectives would be helpful to retrieve opinionated relevant documents. For instance, the word "good" is likely to appear more in relevant documents than non-relevant documents, while the occurrence of the word "definitive" is likely to be the same in relevant or non-relevant documents, because people use "good" more often to express their opinions about a specific topic, while "definitive" is not a discriminative one to distinguish between relevant and non-relevant documents. Calculation of KLD scores for "good" and "definitive" shows that score of "good" is $1.48*10^{-4}$ (ranked as #1), while score of "definitive" is $4.36*10^{-8}$ (ranked as #892). This motivates our hypothesis that document ranking based on KLD scores of subjective adjectives can lead to performance improvement in opinion retrieval.

In this method, a list of 1336 subjective adjectives manually composed by Hatzivassiloglou and McKeown (1997) was used for identifying opinionated contents. We calculated the KLD scores of subjective adjectives based on the Blog-06 relevance judgment file. According to the 5-point scale for the relevance judgment file: -1 – *not judged*; 0 – *non-relevant*; 1 – *relevant*; 2 –

relevant, negative opinion; 3 – *relevant, mixed positive and negative opinion*; 4– *relevant, positive opinion*, the relevant opinionated documents, which have been categorized as 2, 3, 4 in the relevance judgment file, were collected into a set called {O} and the documents, which have been judged as relevant or non-relevant and categorized as 0, 1, 2, 3, 4 in the relevance judgment file, were collected into a set called {All}. Then we calculated the occurrences of subjective adjectives around query term instances within a fixed-sized window of 30 for the two sets of documents, {O} and {All}. To be more specific, the pseudo-code for the calculation is as follows:

```
For each subjective adjective A, A∈the list of 1336 adjectives {
    For each query term t {
        For the distance of 30 words before and after the query term q {
            Calculate the occurrences of A in the set {O}, Freq{O}(A)
            Calculate the occurrences of A in the set {All}, Freq{All}(A)
        }
    }
}
```

Figure 4.1: Pseudo-code for KLD calculation

Moreover, regardless of the fixed-size span before and after the query terms, the total number of words in the windows around query term occurrences in the sets of {O} and {All} was also calculated respectively as totalFreq{O} and totalFreq{All}. The total number of query terms times 60 gave us an approximation of the total number of words around the query terms within the window of 30. Based on the KLD term-scoring function showed in Formula 4.5, the KLD score for subjective adjective A was calculated in Formula 4.6.

$$KLDScore(A) = [P_{\{O\}}(A)] * \log[P_{\{O\}}(A)/ P_{\{All\}}(A)] \quad (4.6)$$

Where $P_{\{O\}}(A) = Freq\{O\}(A) / totalFreq\{O\}$ and $P_{\{All\}}(A) = Freq\{All\}(A) / totalFreq\{All\}$.

As KLD scores are normally very close to zero, the scores are too small to make significant impact on the term weighting. Thus, we normalized the KLD scores to make them more sensitive in the term weighting function. To normalize KLD scores, we simply divided all KLD scores by the maximum KLD score for all 1336 subjective adjectives as shown in Formula 4.7.

$$KLD(A) = \frac{KLD\,Score(A)}{maxKLD} \quad (4.7)$$

This made the normalized KLD scores spread within $(-\infty, 1]$. In all the proposed methods, we utilized the normalized KLD scores for document re-ranking.

With a list of KLD scores of subjective adjectives, document re-ranking was performed by locating all instances of subjective adjectives around the query terms in the document and aggregating the KLD scores for subjective adjectives occurrences. To be more precise, in each document retrieved in response to original query terms, all instances of query terms and subjective adjectives were identified. For each occurrence of query term, we determined whether there were any subjective adjectives before or after the query term in the window of 30. If so, the KLD scores of subjective adjectives were added to $c(t_i)$ to calculate *weighted term frequency* (wf) value as specified in Formula 4.8.

$$c(t_i) = \begin{cases} 1 + \sum_{j=1}^{|J|} KLD(A_j) & if\ |J| > 0 \\ 0 & otherwise \end{cases} \quad (4.8)$$

Where: $c(t_i)$ – contribution of the ith instance of the query term t to wf; $KLD(A_j)$ – KLD score of the subjective lexical unit A_j, $|J|$ – the number of subjective lexical units occurring in the window of 30 words around t_i.

$$wf_t = \sum_{i=1}^{N} c(t_i) \quad (4.9)$$

Where: N – the number of instances of the query term t in the document.

After wf was calculated for a query term, it was used in the standard BM25 formula (Sparck Jones, et al., 1998), which is given in Formula 2.5, in the same way as tf to calculate the *Term Weight* (*TW*) as shown in Formula 4.10.

$$TW_t = \frac{(k_1 + 1) \times wf_t}{k_1 \times ((1-b) + b\frac{DL}{AVDL}) + wf_t} \times idf_t \quad (4.10)$$

All other parameters used in Formula 4.10 were exactly the same as in the BM25 document ranking function. Then the document *Matching Score* (MS) was calculated as the sum of weights of all query terms found in the document according to Formula 4.11.

$$MS = \sum_{t=1}^{|Q|} TW_t \quad (4.11)$$

Where: |Q| is the number of query terms occurring in the document. Then the top 1000 documents retrieved from the first stage were re-ranked based on the MS.

4.5.2 Method 2– Position Information

Motivated by research done by Skomorowski (2006), which computed the probability that an adjective modifies a noun at different distances between [-10, 10] using a training corpus, and calculated the document score as the sum of such probabilities of adjectives, we hypothesized that KLD scores according to different distances in the proximity of query term instance to each subjective adjective within the window of 30 may be helpful for opinion retrieval. By integrating different distances with KLD scores, document re-ranking based on KLD scores of subjective adjectives at different distances can capture both the position information and the divergence of subjective adjectives in relevant documents to the whole collection of documents. For example, for a sentence "I like the laptop because it is very good" in a document, the probability that "good" modifies "laptop" based on probability table computed by Skomorowski (2006) is 0.0156, which is at distance -5. However, as "good" is a heavily used opinionated adjective and is discriminative to distinguish between relevant or non-relevant documents, it should be more significant than other words in opinion retrieval though it is five words away from the query term "laptop" and contribute more to the document re-ranking. In this case, the KLD scores with position information would be helpful to boost the term weight of "good" in the document re-ranking.

In this method, the training set of relevant documents for KLD scores calculation was the same as in method 1. The only difference was that position information was added when calculating

the occurrences of subjective adjectives around query terms instances within a fixed-sized window of 30 for the two sets of documents, {O} and {All}. The pseudo-code for the calculation is as follows:

```
For each subjective adjective A, A∈ the list of 1336 adjectives {
    For each distance n, n∈ [-30, 30] {
        For each query term t {
            Calculate the occurrences of A in the set {O} at distance n, Freq{O}n(A)
            Calculate the occurrences of A in the set {All} at distance n, Freq{All}n(A)
        }
    }
}
```

Figure 4.2: Pseudo-code for KLD at distance n calculation

Similarly as proposed method 1, totalFreq{O} and totalFreq{All} were calculated correspondingly for each distance. The total number of words around the query terms in the window of 30 was approximated by multiplying the total number of query terms occurrences by 60. Based on the KLD term-scoring function shown in Formula 4.5, the KLD score for subjective adjective A at distance n was calculated according to Formula 4.12.

$$\text{KLDScore}_n(A) = [P_{\{O\}n}(A)] * \log[P_{\{O\}n}(A) / P_{\{All\}n}(A)] \quad (4.12)$$

Where $P_{\{O\}n}(A) = \text{Freq}\{O\}_n(A) / \text{totalFreq}\{O\}$ and

$P_{\{All\}n}(A) = \text{Freq}\{All\}_n(A) / \text{totalFreq}\{All\}$.

Then, the normalized KLD scores were calculated as Formula 4.13. The maxKLD_n was the maximum KLD score for all 1336 subjective adjectives at the distance n.

$$\text{KLD}_n\ (\text{A}) = \frac{\text{KLD Score}_n\,(\text{A})}{\text{maxKLD}_n} \tag{4.13}$$

For document re-ranking, we first determined the distance between an instance of the query term and the subjective adjectives around it within the windows of 30, and then looked up the KLD scores associated with the subjective adjectives at those distances. $c(t_i)$ was calculated according to Formula 4.14, replacing KLD(A) with KLD_n (A) in Formula 4.8.

$$c(t_i) = \begin{cases} 1 + \sum_{j=1}^{|J|} KLD_n(A_j) & if\,|J| > 0 \\ 0 & otherwise \end{cases} \tag{4.14}$$

Where: $\text{KLD}_n\ (A_j)$ – KLD score of the subjective lexical unit A_j at distance n, |J| – the number of subjective lexical units occurring in the window of 30 words around t_i. $c(t_i)$, the contribution of the ith instance of the query term t, was used to calculate wf value as shown in Formula 4.9. Instead of tf, wf was used in the standard BM25 formula (Sparck Jones, et al., 1998), in the same way as in Formula 4.10. Similarly, the matching scores of documents were calculated exactly in the same way as in Formula 4.11.

4.5.3 Method 3 - Topic Categories

Inspired by the fact that wording is dependent on the object that one is expressing opinions about, we further extend to investigation whether KLD scores of subjective adjectives computed according to different topical categories can lead to performance improvement in opinion retrieval. For the list of 1336 subjective adjectives used in the document re-ranking, their occurrences in relevant documents were assumed to be the same across all the topics in the former proposed methods. However, people only use a fixed subset of subjective adjectives to express opinions regarding a particular category of topics. In this case, the KLD scores of subjective adjectives should vary from different categories of topics. For instance, "guilty" would be used more frequently in topics expressing opinions about a person, but seldom in topics about a product, while "compatible" would be used more often in the topics related to products, but rarely in topics related to a person. Therefore, assuming that particular words are used to express opinions in a specific category more often than others, KLD scores of subjective adjectives corresponding to different topic categories may benefit document re-ranking.

There are many Named Entity Recognition (NER) tools that can classify atomic elements (undividable words) into predefined categories automatically, such as Named Entity WordNet (Toral et al., 2008) and DRAMNERI (Toral, 2005). The precision of automatic categorization tools can reach 70%. In this research, to attain higher precision, we manually categorized topics into predefined categories - person, event, product, organization, media/art (TV show/ film/ book/ song/ album) and miscellaneous. Topics under each category for Blog-06 and Blog-07 are presented in Appendix B.

The calculation of KLD scores of subjective adjectives was then conducted under query categories. In more detail, we calculated the occurrences of subjective adjectives around query term instances in the window of 30 for the two sets {O} and {All} for each topic in the same way as proposed in method 1. The pseudo-code for the calculation of Freq{O}(A) and Freq{All}(A) was the same as shown in Figure 4.1. And then we summed up the Freq{O}(A) and Freq{All}(A) for topics in the same category. So that, for a subjective adjective A within the query category QC, the occurrences of A 30 words before and after the query terms for {O} and {All} were $\text{Freq}\{O\}_{QC}(A)$ and $\text{Freq}\{All\}_{QC}(A)$, where $\text{Freq}\{O\}_{QC}(A) = \sum \text{Freq}\{O\}_{T}(A)$ and $\text{Freq}\{All\}_{QC}(A) = \sum \text{Freq}\{All\}_{T}(A)$, T is the topic that the query term falls into, $T \in QC$. Similarly, the total number of words in the windows around query terms in the sets of {O} and {All} were calculated under query categories. That means, there is a unique pair of $\text{totalFreq}\{O\}_{QC}$ and $\text{totalFreq}\{All\}_{QC}$ for each category. Based on the KLD term-scoring function shown in Formula 4.5, the KLD score for subjective adjective A for query category qc was calculated according to Formula 4.15.

$$\text{KLDScore}_{qc}(A) = [P_{\{O\}qc}(A)] * \log[P_{\{O\}qc}(A)/ P_{\{All\}qc}(A)] \quad (4.15)$$

Where $P_{\{O\}qc}(A) = \text{Freq}\{O\}_{QC}(A)/ (\text{totalFreq}\{O\}_{QC})$ and $P_{\{All\}qc}(A) = \text{Freq}\{All\}_{QC}(A)/ (\text{totalFreq}\{All\}_{QC})$.

The normalized KLD scores were calculated according to Formula 4.16. The maxKLD_{qc} was the maximum KLD score for all 1336 subjective adjectives under the query category qc.

$$\mathrm{KLD}_{qc}(\mathrm{A}) = \frac{\mathrm{KLD\,Score}_{qc}(\mathrm{A})}{\max \mathrm{KLD}_{qc}} \tag{4.16}$$

For document re-ranking, we first determined which topic category the query term belongs to. And then we looked up the KLD scores of the subjective adjectives under the particular category that the topic falls into. $c(t_i)$ was calculated as Formula 4.17, replacing KLD(A) with KLD_{qc} (A) in Formula 4.8.

$$c(t_i) = \begin{cases} 1 + \sum_{j=1}^{|J|} KLD_{qc}(A_j) & if\,|J| > 0 \\ 0 & otherwise \end{cases} \tag{4.17}$$

Where: $\mathrm{KLD}_{qc}(A_j)$ – KLD score of the subjective lexical unit A_j under query category qc, |J| – the number of subjective lexical units occurring in the window of 30 words around t_i. The contribution of the ith instance of the query term t, $c(t_i)$ was used to calculate wf value as shown in Formula 4.9. Substituting tf with wf, term weights for query terms were calculated in the standard BM25 formula (Sparck Jones, et al., 1998), in the same way as in Formula 4.10. Likewise, the matching scores of documents were calculated exactly in the same way as in Formula 4.11.

4.5.4 Method 4 - Collocates

In the previous proposed methods, we used the list of 1336 subjective adjectives, which is a predefined and manually constructed lexicon, to identify opinionated contents. Collocates,

which are words strongly associated with the query terms in the corpus, can be automatically extracted from relevant documents as opinion targets. Take the document BLOG06-20060119-067-0020907647 about topic #903 (”Steven Jobs”) for instance. Figure 4.3 is an excerpt from the text BLOG06-20060119-067-0020907647. It shows words in the windows of 30 surrounding four occurrences of the query term “Steven Jobs”. As can be seen from the text, none of the subjective adjectives directly modify the query term. However, words around the query term, such as “review”, “links”, “reference”, “comments” and “learn” contain subjectivity. These words can express the author’s opinions about the query. Also, the expression of opinion is via collocates of different parts of speech. Therefore, collocates can be used to build the learning vocabulary for opinion detection.

The Top Ten Best (and Worst) Communicators of 2005 | Main | Seeing Is Believing. Steve Jobs is #1, another great Job by Jobs at Macworld. We rated Steve Jobs as the #1 Communicator of 2005 in our annual Ten Best and Worst list, and he did it again. For an in depth review of the presentation and impact,

…

Listed below are links to weblogs that reference Steve Jobs is #1: Comments I would like to extend my deepest congratulations to Steve Jobs for nabbing the top spot once again. But then again, we must learn that communication would not be effective if he did not have management skills that would give him this honor.

Figure 4.3: An excerpt from document BLOG06-20060119-067-0020907647 for topic #903

There have been a lot of research on collocates extraction and query expansion using collocates (van Rijsbergen, 1977; Vechtomova et al., 2003). Meanwhile, in recent years, there has been a growing interest in using learning vocabulary for opinion retrieval. Yang et al.(2007) extracted

opinion terms by identifying high frequency terms from the positive blog training data (i.e. opinionated blogs) and excluding those that also have high frequency in the negative blog training data to generate the lexicon. By taking into account the proximity of words, such as "I", "you", "we", and "us", as well as the opinion indicator words such as "feel", "like", "hate" and "think", Zhou et al. (2007) generated the learning vocabulary by extracting words around the predefined words within the windows of 20.

Motivated by the use of collocates in IR research, we propose a method to explore opinion retrieval using collocates of the query terms, rather than simply using the fixed lexicon of subjective adjectives. Unlike query expansion, collocates are not explicitly added as additional terms to the query, but are used to calculate KLD scores for document scoring. We want to investigate whether collocates, which are proximate to the query terms within the windows of 30 and extracted from relevant documents could reveal the subjectivity expressed in the relevant documents.

In this method, collocates of query terms, which are words around the query terms within the windows of 30, are used to construct the lexicon instead of the manually constructed list of 1336 adjectives. The procedure we used to select collocates is as follows: in the 1000 top documents retrieved in response to the original query terms, all terms surrounding instances of a query term within the windows of 30 are extracted. In cases where windows surrounding query term instances overlap, terms are extracted only once. Stop words and noise terms, such as "Mr.", "www" and numbers, are manually eliminated. We construct a list of collocates containing 5319 words. To calculate the KLD scores, we apply the same procedure as discussed

in the proposed method 1 to the list of 5319 collocates, instead of the list of 1336 subjective adjectives. Similar to the calculation of KLD scores of subjective adjectives shown in Formula 4.6, KLD scores of collocates were calculated according to Formula 4.18. For each collocate C,

$$\text{KLDScore}(C) = [P_{\{O\}}(C)] * \log[P_{\{O\}}(C)/ P_{\{All\}}(C)] \tag{4.18}$$

Where: $P_{\{O\}}(A) = \text{Freq}\{O\}(C) / \text{totalFreq}\{O\}$ and $P_{\{All\}}(A) = \text{Freq}\{All\}(C) / \text{totalFreq}\{All\}$.

Then, the normalized KLD scores of collocates were calculated according to Formula 4.19. The maxKLD was the maximum KLD score for all 5319 collocates.

$$\text{KLD}(C) = \frac{\text{KLD Score}(C)}{\text{maxKLD}} \tag{4.19}$$

For document re-ranking, in each document retrieved in response to original query terms, all instances of query terms and collocates were identified. For each occurrence of a query term, we determined whether there were any collocates before or after the query term in window of 30. If so, the KLD scores of collocates were added to $c(t_i)$ to as shown in Formula 4.20 to calculate wf.

$$c(t_i) = \begin{cases} 1 + \sum_{j=1}^{|J|} KLD(C_j) & if\,|J| > 0 \\ 0 & otherwise \end{cases} \tag{4.20}$$

Where: $c(t_i)$ – contribution of the ith instance of the query term t to wf; $KLD(A_j)$ – KLD score of the collocates lexical unit C_j, $|J|$ – the number of collocates lexical units occurring in the window of 30 words around t_i. Then, instead of tf, wf was used in the standard BM25 formula (Sparck Jones, et al., 1998), in the same way as in Formula 4.10. Similarly, the matching scores of documents were calculated exactly in the same way as in Formula 4.11.

Chapter 5

Data and Experiment Setup

5.1 Data

In the experiment, we used Blog-06 test collection from TREC as the collection of documents (blogs). This collection has been created and used for 2006 – 2008 TREC Blog Tracks, though the topics have been changed annually. As shown in Table 3.1, Blog-06 test collection consists of 100,649 blogs, with the total uncompressed size of 148GB. The retrieval unit is a permalink document, or a blog post and its associated comments. The total number of permalink documents is 3.2 million. It is a large collection to simulate the real world environment for the experiment.

Based on the same collection of documents, two sets of topics and relevance judgment results file, one from 06 Blog Track and the other from 07 Blog Track, were used for the purpose of training and testing. To be more specific, the calculation of KLD scores of subjective adjectives was run on topics and relevant judgment results file from 06 Blog Track, and the experiment testing was run on topics and relevant judgment results file from 07 Blog Track. Accordingly, the final experiment evaluation measures were calculated by comparing the experiment results and the Blog 07 relevance judgment results. There are 50 topics respectively in Blog06 and Blog07. The topics were diverse as they were selected by NIST from a donated collection of queries sent to commercial blog search engine.

The relevance assessment procedures are the same for Blog-06 and Blog-07. According to the 5-point scale for the relevance judgment: -1 – not judged (due to the offensive language or subject of the document); 0 – non-relevant; 1 – relevant; 2 – relevant, negative opinion; 3 – relevant, mixed positive and negative opinion; 4 – relevant, positive opinion, documents are labeled as -1, 0, 1, 2, 3, or 4 for the given topics. Table 5.1 and Table 5.2 show the relevance assessments of documents for Blog07 and 07 relevance judgments.

Relevance Scale	Label	Nbr. Of Documents
Not Judged	-1	0
Not Relevant	0	47491
Ad hoc-Relevant	1	8361
Negative Opinionated	2	3707
Mixed Opinionated	3	3664
Positive Opinionated	4	4159
Total	-	67382

Table 5.1: Relevance assessments of documents in Blog06 relevance judgment file

Relevance Scale	Label	Nbr. Of Documents
Not Judged	-1	0
Not Relevant	0	42434
Ad hoc-Relevant	1	5187
Negative Opinionated	2	1844
Mixed Opinionated	3	2196
Positive Opinionated	4	2960
Total	-	54621

Table 5.2: Relevance assessments of documents in Blog07 relevance judgment file

5.2 Baselines

5.2.1 Baseline 1

The first baseline used in our evaluation experiments was the well-known BM25 document ranking function implemented in the Wumpus IR system (Büttcher et al., 2006). In our experiment, the run for this baseline is named "BM25". BM25 is selected as the baseline because it has consistently demonstrated high performance results in TREC evaluations. BM25 was also used for the first stage (documents retrieval) of our opinion retrieval algorithm. As different values of the parameters k_1 and b (default values are 1.2 and 0.75, respectively) in BM25, contribute to different performance for different collections, before commencing the experiments, we evaluated BM25 with different values for k_1 and b on Blog-06 collection. Comparisons between different values for k_1 and b are shown in Table A.1 and A.2 in Appendix A respectively for topical relevance and opinion relevance evaluation results. As shown in the tables, b = 0.1 and k_1 = 1.75 yielded the best performance results in MAP and were therefore, used in all the baselines and all runs in the evaluation experiments.

5.2.2 Baseline 2

The second baseline we used to evaluate the experiment performance was our implementation of BM25, counting only the instances of query terms co-occurring with subjective adjectives within the window of 30. That means, only query term instances which occurred within the window of 30 of subjective adjectives were counted towards term frequency in standard BM25. Contribution of ith instance of the query term t, $c(t_i)$ was calculated as Formula 5.1.

$$c(t_i) = \begin{cases} 1 & if\,|J| > 0 \\ 0 & otherwise \end{cases} \quad (5.1)$$

Where: |J| is the number of subjective adjectives occurring in the window of 30 words around t_i. The calculation of document matching scores is the same as in standard BM25 (Sparck Jones, et al., 1998). The detailed formulae for calculating document matching scores are described in Section 4.5.1.

This baseline is named as "BM25op". It is selected as baseline because it is comparable to our proposed methods, which use KLD score for document weighting. As you can see from Formula 5.1, in the BM25op, it assigned a constant weight of 1 to those query terms, which co-occur with any subjective adjectives within a window of 30. That is, regardless of the significance of the subjective adjectives, their contributions to the document weighting are equally constant. While the integration of KLD scores of subjective adjectives reward the adjectives differently based on their ability to discriminate between relevant and non-relevant documents, KLD scores take into account the significance of each subjective adjective in document weighting. Thus, the comparison between BM25op and the reported methods using KLD scores can capture any relative benefit of KLD scores. Therefore, baseline 2 acts as a stricter baseline than baseline 1.

5.3 Runs

According to our four proposed methods, there are four evaluation experiments. In all experiments, we used user-defined phrases (text enclosed in double quotes) for the first stage

(initial documents retrieval). And for the second stage (opinion-based documents re-ranking), experiments were conducted with different types of queries constructed from the topic titles: single terms and phrases. Experimental runs tagged with "-s" mean using single terms for the second stage, while runs tagged with "-p" mean using phrases for the second stage. Also, experimental runs suffixed with "-all" represent all query terms in the document were counted, while experimental runs without suffixed of "-all" mean only query term instances which occurred within the fixed window were counted towards wf (weighted tf).

For method 1, we conducted experiments to calculate KLD scores of subjective adjectives which co-occur with the query terms within the window of 30 and apply the KLD scores in weighting for document re-ranking. The runs tagged with "KLD" used KLD scores of subjective adjectives for document re-ranking as described in Section 4.5.1. Four runs were conducted: KLD-s, KLD-s-all, KLD-p, and KLD-p-all.

Considering distances between query term and subjective adjective for each instance of co-occurrence, method 2 calculated KLD scores of subjective adjectives based on their distance from the query terms and assigned KLD scores to subjective adjectives according to their positions accordingly. The runs tagged with "KLD-Dist" presented method 2 described in Section 4.5.2. Four runs were conducted: KLD-Dist -s, KLD-Dist -s-all, KLD-Dist -p, and KLD-Dist -p-all.

For method 3, experiments were conducted to calculate KLD scores of subjective adjectives under predefined query categories. And then the assignment of KLD scores was based on

categories that the topic falls into. “KLD-QCat” denoted the runs for method 3 as discussed in Section 4.5.3. Four runs were conducted: KLD-QCat-s, KLD-QCat-s-all, KLD-QCat-p, and KLD-QCat -p-all.

Further in method 4, we first extracted a list of 5319 collocates around the query terms within the window of 30, and calculated KLD scores of the list of collocates. There are two thresholds for the method – one used collocates with top 1000 KLD scores and the other used collocates with top 500 KLD scores. The runs tagged with “KLD-1000Col” and “KLD-500Col” corresponds to the method 4 described in Section 4.5.4 using the two thresholds. Eight runs were conducted: KLD-1000Col-s, KLD-1000Col-s-all, KLD-1000Col-p, KLD-1000Col-p-all, KLD-500Col-s, KLD-500Col-s-all, KLD-500Col-p, KLD-500Col-p-all.

5.4 Architecture and Evaluations

All scripts in the experiments are coded in Perl because of its advantage in text manipulation. Perl is a high-level, general-purpose, dynamic programming language. The powerful text processing facilities without arbitrary data length limits makes it the ideal language for manipulating text files.

All our experiments are implemented under Wumpus IR system. Wumpus is an experimental search engine developed at the University of Waterloo (Büttcher et al., 2006). It is publicly available for download under the terms of the GNU General Public License (GPL). Wumpus not only allows multi-user access, but also supports file system indexing service that can automatically keep track of all changes in the documents and update the index accordingly.

Wumpus enables efficient search on large-scale text collections, which consist of many hundreds of gigabytes of text and dozens of millions of documents. Also, complicated structural queries can be implemented by Wumpus using generalized concordance lists (GCL) query language (Clarke et al., 1995).

Taking into account that with the fixed topics, documents and relevance judgments, TREC approach can easily repeat the experiment, tune the parameters and be conducted regardless of human factors for large-scale test collections within a short period of time, we apply TREC approach for experiments evaluation. TREC_EVAL program is used to calculate the standard measures of system effectiveness by comparing the experiment results file and the relevance judgments file. In our experiments, we mainly focus on evaluation measures of MAP, R-Prec, bpref, and P@10 for all topics. There are two types of evaluations done in the experiments – topical relevance evaluation and opinion relevance evaluation. The topical relevance evaluation measures the performance of methods in retrieving documents which are about a specific topic, while the opinion relevance evaluation measures the performance of methods in retrieving opinionated documents, i.e. which express opinions about the given topic. In more detail, topical relevance evaluation compares experiment results with relevance judgment results that are categorized as relevant documents with the label of 1, 2, 3 or 4, according to the 5-point scale of the relevance judgment file, while opinion relevance evaluation compares experiment results with relevance judgment results that are categorized as opinionated relevant documents only with the label of 2, 3 or 4.

Chapter 6

Result and Discussion

6.1 Parameters Setting

Based on the comparison between opinion finding with and without stemming on Blog-06 datasets done by Vechtomova (2007), better results were obtained without stemming. Thus, stemming was not used in either baseline, or experimental methods.

Different values for k_1 and b with BM25 were evaluated on Blog-06 datasets. Topical relevance and opinion relevance evaluation results with different values of b and k_1 using measures of MAP, R-Prec, bpref, and P@10 are shown respectively in Table A.1 and A.2 in Appendix A. As shown in the tables, b = 0.1 and k_1 = 1.75 yielded the best performance and were therefore, used in all the baselines and all runs in the evaluation experiments.

In the experiments, the window was defined as n words before and after the query term occurrence in text. If a subjective adjective appears within the window of the query term, it may indicate that the author expresses opinion about the query term. The reasons we chose to use fixed-size window instead of a natural language unit, such as a sentence, are as follows. First, adjectives may not directly modify the query terms, but words that are highly associated with the query terms, such as pronouns, features or parts of the query terms. For instance, “I love the movie Harry Porter, because it (its leading role) is very interesting”, the target of “interesting”

is “it” (pronoun) or “its leading role” (component), though the movie is the actual target. Secondly, using fixed-size window can avoid syntactic errors when splitting text into sentences, especially in blogs, where sentence construction is loose. Experiments for opinion re-ranking with different window sizes were evaluated using the proposed method 1 described in Section 4.5.1 counting only query terms co-occurring with subjective adjectives within the windows of 10, 20 and 30 using single terms and phrases for re-ranking respectively. Results are shown in Table 6.1 and 6.2.

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
KLD-p-w10	0.2361	0.3414	0.3482	0.5840	0.1413	0.2398	0.2233	0.3520
KLD-p-w20	0.2923	0.3836	0.3881	0.6713	0.1997	0.2812	0.2603	0.4550
KLD-p-w30	0.3679	0.4126	0.4372	0.7120	0.2892	0.3430	0.3340	0.5540

Table 6. 1: Results for runs using KLD-p with window sizes of 10, 20 and 30

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
KLD-s-w10	0.2366	0.3421	0.3500	0.5850	0.1434	0.2400	0.2207	0.3420
KLD-s-w20	0.2934	0.3840	0.3876	0.6720	0.2001	0.2833	0.2620	0.4500
KLD-s-w30	0.3776	0.4209	0.4426	0.7320	0.3046	0.3573	0.3453	0.5760

Table 6. 2: Results for runs using KLD-s with window sizes of 10, 20 and 30

Results show that window size of 30 gives the best performance for both runs KLD-p and KLD-s. Therefore, window size of 30 was used in all the baselines and all runs in the evaluation experiments.

6.2 Results Analysis - Method 1

By comparing evaluation results in Table 6.3 and 6.4, it can be seen that the use of phrases for document re-ranking gives better results than single terms. Also, when queries were treated as single terms for re-ranking, applying KLD scores only to query term instances which occur within the window of 30 of subjective adjectives performed better than applying KLD scores to all query terms, and contrarily, when queries were considered as phrases, applying KLD scores to all query terms gives better performance.

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
BM25 (Baseline1)	0.3691	0.4162	0.4277	0.7020	0.2873	0.3487	0.3227	0.5220
BM25op-s (Baseline2)	0.3845	0.4235	0.4436	0.7260	0.3059	0.3550	0.3382	0.5600
KLD-s	0.3776	0.4209	0.4426	0.7320	0.3046	0.3573	0.3453	0.5760
KLD-s-all	0.3779	0.4224	0.4345	0.7040	0.2976	0.3497	0.3292	0.5400

Table 6.3: Evaluation Results for runs using KLD scores of subjective adjectives with single terms for re-ranking

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
BM25 (Baseline1)	0.3691	0.4162	0.4277	0.7020	0.2873	0.3487	0.3227	0.5220
BM25op-p (Baseline2)	0.3793	0.4261	0.4451	0.7320	0.2996	0.3527	0.3370	0.5640
KLD-p	0.3679	0.4126	0.4372	0.7120	0.2892	0.3430	0.3340	0.5540
KLD-p-all	0.3706	0.4165	0.4337	0.6920	0.2906	0.3448	0.3284	0.5240

Table 6.4: Evaluation Results for runs using KLD scores of subjective adjectives with phrases for re-ranking

With single terms for re-ranking, comparing the best run using KLD scores of subjective adjective with single terms, KLD-s, with BM25op-s (baseline 2), the performance dropped 0.42% in MAP for opinion retrieval, but there is 2.9% improvement in P@10. Comparing with

BM25, KLD-s improved the performance of opinion retrieval by 6%. On the other hand, comparing the run using KLD scores of subjective adjectives with phrases used for re-ranking, KLD-p, with BM25op-p (baseline 2), we can observe 3.4% setback in MAP for opinion relevance, as well as a drop of 1.8% in P@10. In conclusion, KLD-s yields the best results (MAP = 0.3046) in opinion retrieval among all runs using KLD score for document re-ranking in experiments of method 1, but it cannot outperform BM25op.

The difference between performance of method 1 and performance of baseline 2 is particularly important as it determines whether KLD scores of nearby subjective adjectives have benefit in adjusting the weights of query term instances for document re-ranking in opinion finding. The differences in average precision (opinion relevance) per topic between KLD-s and BM25op-s (baseline 2) runs are shown in Figure 6.1.

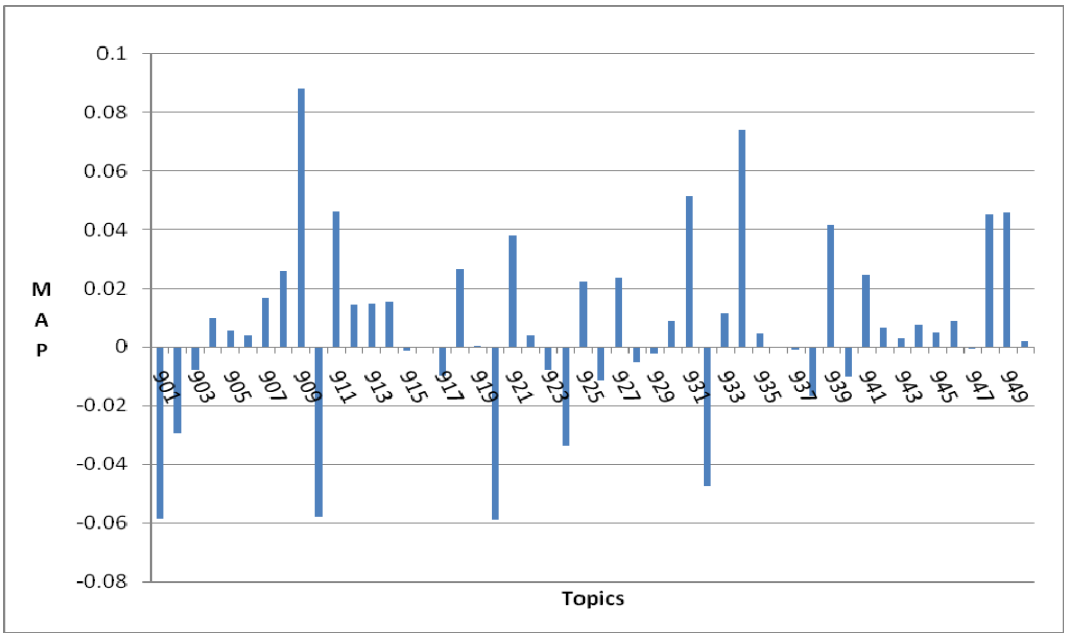

Figure 6.1: Differences in average precision (opinion relevance) per topic between KLD-s and BM25op-s runs

Among the topics that were improved by the opinion re-ranking based on KLD scores of subjective adjectives around the query terms within the windows of 30 are: #909 Barilla, #934 cointreau, #931 "fort mcmurray", #911 "SCI FI CHANNEL", # 949 ford bell, and #948 sorbonne. Among the topics that dropped in performance are: #920 "andrew coyne", #901 jstor, #910 "Aperto Networks", #932 goobuntu, #924 "mark driscoll", #902 "lactose gas" and #938 "plug awards".

The experimental results show that all runs cannot achieve better performance than the corresponding baselines. KLD-s, KLD-s-all and KLD-p demonstrated statistically insignificant performance. T-test shows that the runs KLD-s (P = 0.052506), KLD-s-all (P = 0.280012), KLD-p (P = 0.878314) and KLD-p-all (P = 0.441646) were statistically insignificant compared to BM25op.

Comparing with the Proximity-based method by Vechtomova (2007) discussed in Section 4.2.2, runs "Proximity-based-s" and "Proximity-based-p" were conducted with b=0.1 and k_1=1.75, which are different from the performance tuning values (b=0.75 and k_1=1.2) used in Proximity–based method. Based on the same dataset, the evaluation results in Table 6.5 show that compared with Proximity-based-s, KLD-s has 1.8% improvement in MAP and 4.3% improvement in P@10 for opinion relevance. Also as shown in Table 6.6, compared with Proximity-based-p, KLD-p has 0.07% improvement in MAP and 1.5% improvement in P@10. In addition, KLD-p-all has improvement over Proximity-based-p by 0.5% in MAP.

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
Proximity-based-s	0.3701	0.4048	0.4383	0.6980	0.2993	0.3430	0.3391	0.5520
KLD-s	0.3776	0.4209	0.4426	0.7320	0.3046	0.3573	0.3453	0.5760
KLD-s-all	0.3779	0.4224	0.4345	0.7040	0.2976	0.3497	0.3292	0.5400

Table 6.5: Evaluation Results for runs using Proximity-based method and KLD scores of subjective adjectives with single terms for re-ranking

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
Proximity-based-p	0.3627	0.4038	0.4366	0.6940	0.2890	0.3301	0.3312	0.5460
KLD-p	0.3679	0.4126	0.4372	0.7120	0.2892	0.3430	0.3340	0.5540
KLD-p-all	0.3706	0.4165	0.4337	0.6920	0.2906	0.3448	0.3284	0.5240

Table 6.6: Evaluation Results for runs using Proximity-based method and KLD scores of subjective adjectives with phrases for re-ranking

The difference in performance of Proximity-based method and proposed method 1 (KLD) can be attributed to the use of KLD score. From the results, using KLD scores slightly outperformed the proximity-based method as the use of KLD in opinion-finding re-ranking gave more weights to terms that made the relevant documents divergent to the rest of the documents in the overall collection.

6.3 Results Analysis - Method 2

In the results shown in Table 6.7 and 6.8, all runs for method 2 gained improvement compared to the baseline 1, but dropped in performance compared with baseline 2. And it is obvious that applying KLD scores only to query term instances which occur within the window of 30 of subjective adjectives performed better than applying KLD scores to all query terms.

With single terms used for re-ranking, comparing KLD-Dist-s with BM25op-s (baseline 2), there is 0.8% drop in MAP for opinion retrieval, but 2.1% improvement in P@10. Compared with BM25 (baseline 1), KLD-Dist-s improved the performance of opinion retrieval by 5.7% in MAP and 9.6% improvement in P@10. On the other hand, with phrases used for re-ranking, comparing KLD-Dist-p with BM25op-p (baseline 2), we can observe 2.5% fall in MAP for opinion relevance, as well as 1.8% fall in P@10.

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
BM25 (Baseline1)	0.3691	0.4162	0.4277	0.7020	0.2873	0.3487	0.3227	0.5220
BM25op-s (Baseline2)	0.3845	0.4235	0.4436	0.7260	0.3059	0.3550	0.3382	0.5600
KLD-Dist-s	0.3732	0.4170	0.4387	0.7220	0.3036	0.3542	0.3413	0.5720
KLD-Dist-s-all	0.3771	0.4236	0.4340	0.6980	0.2958	0.3493	0.3295	0.5340

Table 6.7: Evaluation Results for runs using KLD scores of subjective adjectives at different distances with single terms for re-ranking

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
BM25 (Baseline1)	0.3691	0.4162	0.4277	0.7020	0.2873	0.3487	0.3227	0.5220
BM25op-p (Baseline2)	0.3793	0.4261	0.4451	0.7320	0.2996	0.3527	0.3370	0.5640
KLD-Dist-p	0.3644	0.4105	0.4339	0.7080	0.2919	0.3490	0.3335	0.5540
KLD-Dist-p-all	0.3699	0.4218	0.4334	0.6980	0.2899	0.3517	0.3301	0.5300

Table 6.8: Evaluation Results for runs using KLD scores of subjective adjectives at different distances with phrases for re-ranking

Also, the performance of KLD scores associated with position did not show improvement from the original KLD method 1. This may be explained by the fact that KLD score calculation based on position to some extent restricted the KLD scores to a particular position, rather improving

the approximation of contributions of subjective adjectives in distinguishing between relevant documents and the rest of the collection.

The differences in average precision (opinion relevance) per topic between KLD-Dist-s and BM25op-s (baseline 2) runs are shown in Figure 6.2. Among the topics that were improved by the opinion re-ranking based on KLD scores of subjective adjectives around the query terms within the windows of 30 are: #909 Barilla, #934 cointreau, #910 "Aperto Networks", #918 varanasi, #948 sorbonne, and #939 "Beggin Strips". Among the topics that dropped in performance are: #938 "plug awards", #920 "Andrew coyne", #901 jstor, #931 "fort mcmurray", #947 "sasha cohen" and #924 "mark driscoll".

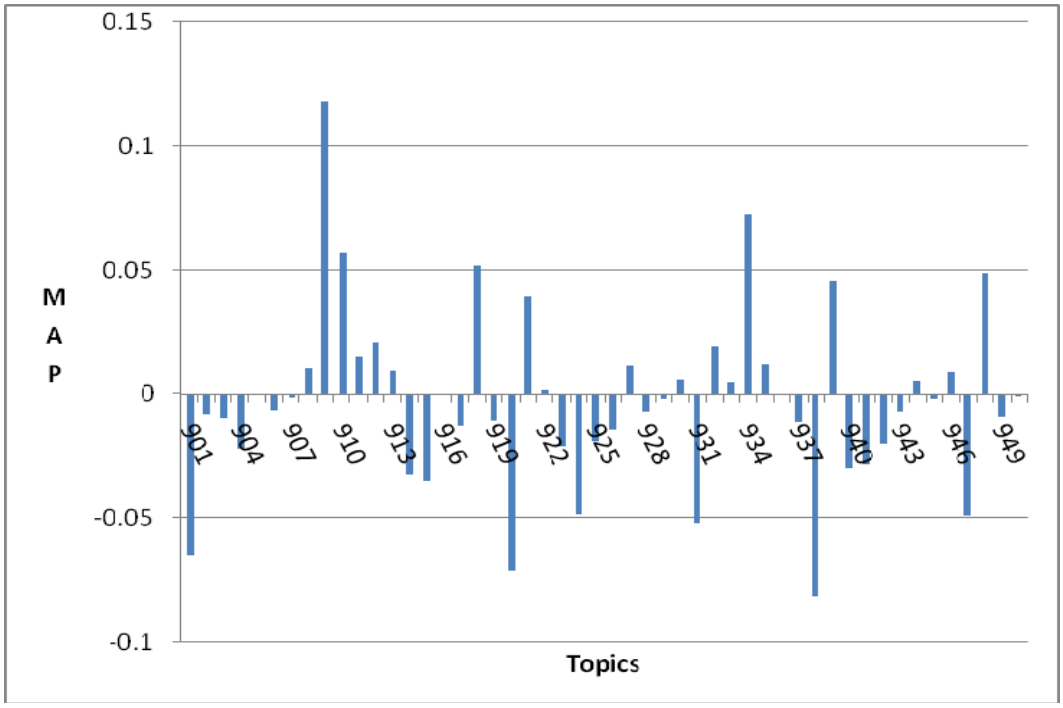

Figure 6.2: Differences in average precision (opinion relevance) per topic between KLD-Dist-s and BM25op-s runs

The experimental results show that all runs achieved better performance than baseline1, but none of them achieved better performance than baseline 2. The runs KLD-Dist-s (P=0.646752) demonstrated statistically insignificant performance compared to BM25op-s. KLD-Dist-s-all (P=0.371746), KLD-Dist-p (P=0.6880417) and KLD-Dist-p-all (P=0.383421) also demonstrated statistically insignificant performance compared to BM25op-s.

6.4 Results Analysis - Method 3

By comparing evaluation results in Table 6.9 and 6.10, it can be seen that the use of single terms for document re-ranking gives better results than phrases. Also, it is worthy to point out that applying KLD scores to query term instances which occur within the window of 30 of collocates achieved better performance than applying KLD scores only to all query terms.

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
BM25 (Baseline1)	0.3691	0.4162	0.4277	0.7020	0.2873	0.3487	0.3227	0.5220
BM25op-s (Baseline2)	0.3845	0.4235	0.4436	0.7260	0.3059	0.3550	0.3382	0.5600
KLD-QCat-s	0.3790	0.4166	0.4434	0.7340	0.3077	0.3637	0.3530	0.5820
KLD-QCat-s-all	0.3804	0.4214	0.4344	0.7000	0.2974	0.3551	0.3306	0.5280

Table 6.9: Evaluation Results for runs using KLD scores of subjective adjectives based on query categories with single terms for re-ranking

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
BM25 (Baseline1)	0.3691	0.4162	0.4277	0.7020	0.2873	0.3487	0.3227	0.5220
BM25op-p (Baseline2)	0.3793	0.4261	0.4451	0.7320	0.2996	0.3527	0.3370	0.5640
KLD-QCat-p	0.3732	0.4104	0.4412	0.7220	0.3000	0.3604	0.3499	0.5680
KLD-QCat-p-all	0.3787	0.4252	0.4386	0.7100	0.2941	0.3555	0.3322	0.5240

Table 6.10: Evaluation Results for runs using KLD scores of subjective adjectives based on query categories with phrases for re-ranking

Comparing KLD-QCat-s with BM25op-s (baseline 2), there is a 0.6% improvement in MAP for opinion retrieval, as well as 4% improvement in P@10. Compared with BM25 (baseline 1), KLD-Qcat-s improved the performance of opinion retrieval for 7.1% in MAP and 11.5% in P@10. On the other hand, with phrases used for re-ranking, comparing KLD-Qcat-p with BM25op-p (baseline 2), we can observe 0.13% improvement in MAP for opinion relevance, as well as 0.7% in P@10. Also, compared with BM25 (baseline 1), KLD-Qcat-p improved the performance of opinion retrieval for 4.4% in MAP and 7.7% in P@10.

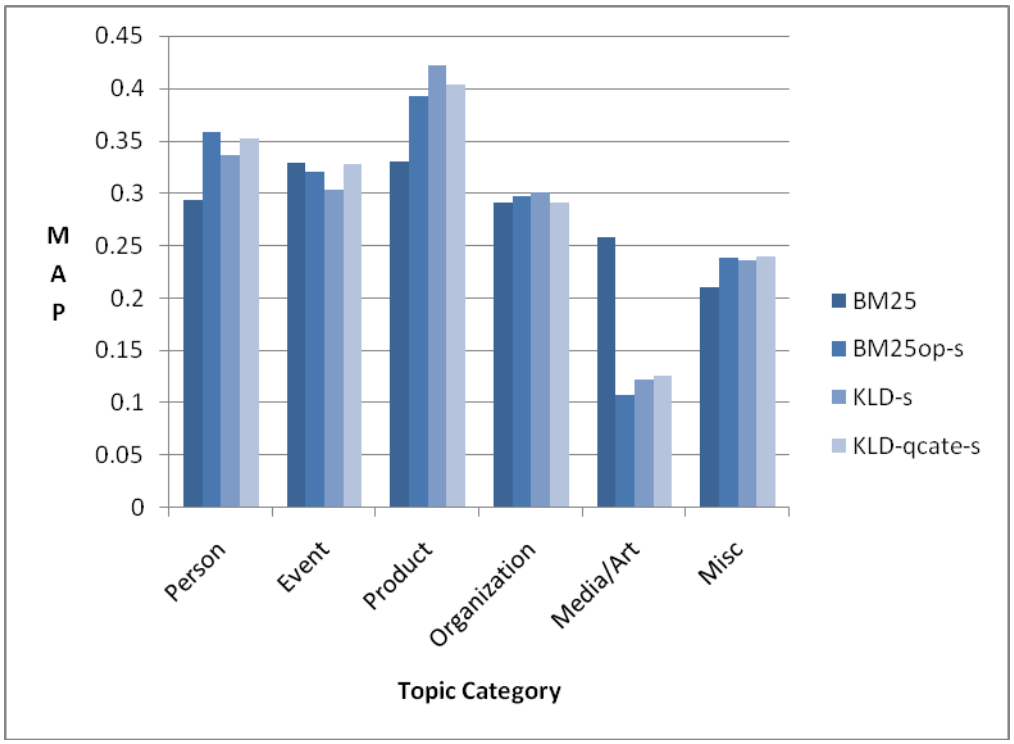

Figure 6.3: Opinion relevance results in MAP by categories

The opinion relevance performance of BM25, BM25op-s, KLD-s and KLD-Qcat-s in MAP by categories is presented in Figure 6.3. As can be seen from Figure 6.3, compared to BM25op-s (baseline 2), KLD-Qcat-s gained improvement in categories - “Event”, “Product”, “Media/Art”

and “Misc”. “Person” and “Organization” are the two categories that dropped in performance against “KLD-p”. The drop in “Organization” was due to topic #926 “hawthorne heights” and #948 sorbonne, while the drop in “Misc” was due to the topic #931 “fort mcmurray” and #942 “lawful access”.

The opinion relevance performance of BM25, BM25op-s, KLD-s and KLD-Qcat-s in P@10 by categories is presented in Figure 6.4. The average performance in categories “Product” and “Media/Art” is the same for BM25op-s (Baseline 2) and KLD-Qcate-s. Compared to BM25op-s, the biggest improvement by using KLD-Qcat-p was achieved by “Organization”, followed by “Misc”, “Person” and “Event”.

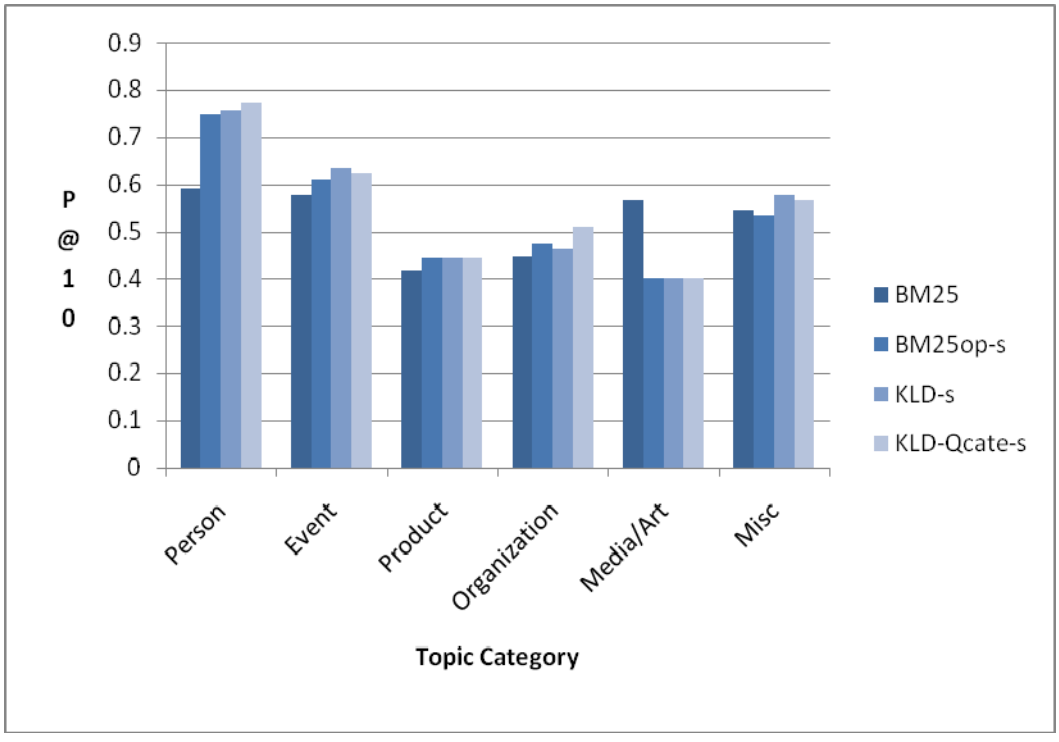

Figure 6.4: Opinion relevance results in P@10 by categories

The experimental results show that KLD-Qcat-s and KLD-Qcat-p achieved better performance than the corresponding baselines. KLD-Qcat-s and KLD-Qcat-p demonstrated statistically insignificant performance. T-test shows that the runs KLD-Qcat-s (P= 0.745309) and KLD-Qcat-p (P=0.538613) are statistically insignificant compared to BM25op-s and BM25op-p. Also, the runs KLD-Qcat-s-all (P=0.183234) and KLD-Qcat-p-all (P=0.536143) are statistically insignificant compared to BM25op respectively.

6.5 Results Analysis - Method 4

In the results shown in Table 6.11 and 6.12, all runs for method 4 dropped in performance compared to the BM25op (Baseline 2). The runs applying KLD scores all collocates around the query terms within the windows of 30 achieved improvement compared to BM25 (Baseline 1). Apparently, runs using single terms for document re-ranking achieved better results than phrases in all measures. Also, runs using the threshold of top 1000 KLD scores gave slightly better results than using the threshold of top 500 KLD scores using both single terms and phrases for re-ranking.

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
BM25 (Baseline1)	0.3691	0.4162	0.4277	0.7020	0.2873	0.3487	0.3227	0.5220
BM25op-s (Baseline2)	0.3845	0.4235	0.4436	0.7260	0.3059	0.3550	0.3382	0.5600
KLD-1000Col-s	0.3168	0.3516	0.3976	0.6920	0.2430	0.2806	0.2920	0.5000
KLD-1000Col-s-all	0.3702	0.4117	0.4244	0.6980	0.2893	0.3456	0.3219	0.5280
KLD-500Col-s	0.3078	0.3508	0.3944	0.6580	0.2371	0.2751	0.2961	0.4820
KLD-500Col-s-all	0.3701	0.4119	0.4243	0.6960	0.2892	0.3457	0.3218	0.5260

Table 6.11: Evaluation Results for runs using KLD scores of collocates with single terms for re-ranking

Run Name	Topical Relevance				Opinion Relevance			
	MAP	R-prec	bpref	P@10	MAP	R-prec	bpref	P@10
BM25 (Baseline1)	0.3691	0.4162	0.4277	0.7020	0.2873	0.3487	0.3227	0.5220
BM25op-p (Baseline2)	0.3793	0.4261	0.4451	0.7320	0.2996	0.3527	0.3370	0.5640
KLD-1000Col-p	0.3120	0.3486	0.3953	0.6780	0.2372	0.2760	0.2881	0.4860
KLD-1000Col-p-all	0.3703	0.4226	0.4307	0.7040	0.2882	0.3493	0.3244	0.5220
KLD-500Col-p	0.3063	0.3496	0.3945	0.6460	0.2327	0.2707	0.2927	0.4620
KLD-500Col-p-all	0.3703	0.4231	0.4305	0.7040	0.2881	0.3493	0.3243	0.5220

Table 6.12: Evaluation Results for runs using KLD scores of collocates with phrases for re-ranking

With single terms used for re-ranking, comparing KLD-1000Col-s with BM25op-s (baseline 2), there is a big drop of 20.6% in MAP for opinion retrieval, and a drop of 10.7% in P@10. Compared with BM25 (baseline 1), KLD-1000Col-s dropped 15.4% in MAP for opinion retrieval. On the other hand, with phrases used for re-ranking, compared with BM25op-p (baseline 2), KLD-1000Col-p dropped 20.8% in MPA as well as 7.1% in P@10 for opinion relevance.

The drop in performance using KLD scores of collocates around the query terms could be explained by the fact that collocates of query terms may consist of a number of noisy terms, which don't contain opinions about the query terms. The experimental results show that all runs failed to achieve better performance than BM25op (Baseline 2), but all runs using KLD scores of all collocates around the query terms within the windows of 30 achieved slight improvement compared to BM25 (Baseline 1).

The runs KLD-1000Col-s (P=0.005128) demonstrated statistically insignificant performance compared to BM25. And T-test shows that the runs KLD-1000Col-p (P=0.000853), KLD-

500Col-s (P=0.003251) and KLD-500Col-p (P=0.00093) are statistically significant compared to BM25op correspondingly.

Chapter 7

Conclusion and Future Work

In this thesis, four hypotheses were proposed to study the use of KLD scores of subjective adjectives and collocates in document re-ranking for opinion retrieval. Below are the conclusions for the four hypotheses.

Hypothesis 1: Document ranking using KLD scores of subjective adjectives results in performance improvement in the opinion retrieval task compared to baseline systems.

The runs using KLD scores of subjective adjectives around the query terms within windows of 30 outperformed BM25 (Baseline 1), but were not as good as BM25op (Baseline 2). The result of t-test analysis show the result is statistically insignificant.

Hypothesis 2: Document ranking using KLD scores of subjective adjectives calculated for each distance from a query term results in performance improvement in the opinion retrieval task compared to baseline systems.

The results analysis partially supports this hypothesis. The runs did not outperform BM25op (Baseline 2), though improved compared with BM25 (Baseline 1).

Hypothesis 3: Document ranking using KLD scores of subjective adjectives calculated for each topic category results in performance improvement in the opinion retrieval task compared to baseline systems.

The results of analysis support this hypothesis. The method of using KLD scores of subjective adjectives calculated for each topic category slightly improves the opinion retrieval performance compared with both baseline systems. The topics under categories "Event" and "Product" achieved improvement in opinion retrieval. This indicates that using KLD scores of subjective adjectives under topic category is recommended for the topics under categories "Event" and "Product".

Hypothesis 4: Document ranking using KLD scores of collocates of query terms results in performance improvement in the opinion retrieval task compared to baseline systems.

The experiment using KLD scores of collocates of query terms dropped in opinion retrieval performance compared to the baseline systems. Collocates of query terms are automatically selected from words around the query terms within the window of 30 did not prove to be helpful for opinion detection.

Future studies can extend this work in the following directions:

- Opinion polarity. The current work focuses on retrieving documents which contain any opinions about the input query. Opinion polarity (positive or negative) is outside of its scope. The polarity of subjective adjectives can be taken into account to retrieve documents that contain either positive, or negative opinions about the query target.

- Inter-relationship between query terms and collocates. As people may express opinions not only towards the targets directly, but also towards something related to the targets. For example, in the topic category "Product", people may express opinions about special features or parts of the product, instead of the product itself. For the review of automobile model "Lexus RX" as example, opinions could be expressed on its specific features "fuel efficiency", or components "wheels". If we could identify these related concepts based on their relationships with the targets, it may be helpful for opinion retrieval.

- Pronouns detection. A fixed-size window was used in this study to extract collocates of query terms. The purpose of doing this is to capture nearby words that contains opinions about the query terms. People frequently use pronouns to refer to the concepts they mentioned in the text earlier. More sophisticated linguistic-based methods can be used to detect co-reference using pronouns (anaphora) in texts. In computational linguistics, a number of anaphora resolution methods have been developed with the purpose of identifying which of the previously mentioned nouns or noun phrases a pronoun refers to. By using such methods, more precise subjectivity extraction may be achieved.

- More complicated query structures. The proposed methods are based on the assumption that users are searching for the opinion about all the concepts expressed in the query. However, queries may be more complex, for example, a query “What does Person X think about global warming?” The user is not interested in opinions about Person X, but about his/her views/attitudes towards global warming.

Bibliography

Amati, G. and van Rijsbergen, C. J. (2002). Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems*, 20(4), 357-389.

Baker, M. C. (2003). Lexical categories: verbs, nouns and adjectives. Cambridge University Press.

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). Modern information retrieval.

Bennett, G., Scholer, F. and Uitdenbogerd, A. (2008). A comparative study of probabilistic and language models for information retrieval. *In Proceedings of the Nineteenth Australasian Database Conference,* 65-74.

Bruce, R. F. and Wiebe, J. M. (2000). Recognizing subjectivity: a case study in manual tagging. Natural Language Engineering, 5(2), 187-205.

Buckley, C. and Voorhees, E. M. (2000). Evaluating evaluation measure stability. *In Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 33–40.

Büttcher, S., Clarke, C. and Lushman, B. (2006). Term proximity scoring for ad-hoc retrieval on very large text collections. *In Proceedings of the 29th ACM conference on research and development in information retrieval* (ACM-SIGIR), pp. 621–622.

Carpineto, C., de Mori, R., Romano, G. and Bigi, B. (2001). An information-theoretic approach to automatic query expansion. *ACM Trans*. Information System, 19(1), 1–27.

Clarke, C. L. A., Cormack, G. V. and Burkowski, F. J. (1995). An algebra for structured text search and a framework for its implementation. *The Computer Journal*, 38(1), pp.43-56.

Cover, T.M. and Thomas, J.A. (1991). Elements of information theory, Wiley, New York.

Croft, W. B. and Happer, D. J. (1979). Using probabilistic models of document retrieval without relevance information. *Journal of Documentation*, 35, 285-289.

Dave, K., Lawrence, S. and Pennock, D. M. (2003). Mining the peanut gallery: opinion extraction and semantic classification of product reviews. *In Proceedings of the 12th World Wide Web Conference.* 173-192.

Dixon, R. M. W. and Aikhenvald, A. Y. (2004). Adjective classes a cross linguistic typology. Oxford University Press.

Fan, W., Gordon, M. D. and Pathak, P. (2004). Discovery of context-specific ranking functions for effective information retrieval using genetic programming. *IEEE Transactions on Knowledge and Data Engineering*, 16(4).

Fuhr, N. (1992). Probabilistic model in information retrieval.

Greenbaum, S. (1996). The Oxford English grammar. Oxford University Press.

Harman, D. (1993). Overview of the First Text REtrieval Conference (TREC-1). *National Institute of Standards and Technology*.

Hatzivassiloglou V. and McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. *In Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics*, pp. 174-181.

Hu, M. and Liu, B. (2004). Mining opinion features in customer reviews. *In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery & Data Mining,* 45-52.

Hurst, M. and Nigam, K. (2004). Retrieving topical sentiments from online document collections. *In proceedings of the 11th conference on document recognition and retrieval,* 134-163.

Kolla, M. and Vechtomova, O. (2007). Enterprise search: Identify relevant sentences and using them for query expansion. *In proceedings of the Sixteenth Text Retrieval Conference,* 62-71.

Li, X. and Roth, D. (2001). Exploring evidence for shallow parsing. *In Proceedings of the Annual Conference on Computational Natural Language Learning.* 38-42.

Macdonald, C. and Ounis, I. (2006). The TREC Blog06 Collection: Creating and Analysing a Blog Test Collection DCS Technical Report TR-2006-224. Department of Computing Science, University of Glasgow.

Mishne, G. (2006). Multiple ranking strategies for opinion retrieval in blogs, The University of Amsterdam at the 2006 TREC Blog Track. *In Proceedings of the Fifteenth Text REtrieval Conference*, 73.

Mishne, G. and de Rijke, M. (2006). A study of blog search. *In Proceedings of ECIR-2006,* LNCS vol 3936, Springer.

Nuray, R. and Can, F. (2003). Automatic ranking of retrieval systems in imperfect environments. *In Proceedings of the Twenty-Sixth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 379–380.

Ounis, I., Macdonald, C. and Soboroff, I. (2006). Overview of the TREC-2006 Blog Track.

Ounis, I., de Rijke, M., Macdonald, C., Mishne, G. and Soboroff, I. (2006). Overview of the TREC-2006 Blog Track.

Rijkhoek, P. (1998). On degree phrases and result clauses. PhD thesis, University of Groningen, Groningen.

Robertson, S. E. (1977). The probability ranking principle in IR. *Journal of documentation*, 33(4), 294-304.

Robertson, S. E., Maron, M. E. and Cooper, W. S. (1982). Probability of relevance: a unification of two competing models for document retrieval. *Information Technology: Research and Development,* 1, 1-21.

Robertson, S. E., Walker, S., Hancock-Beaulieu, M. M. and Gatford, M. (1994). Okapi at TREC-3**,** TREC-3.

Roussinov, D. and Fan, W. (2006). Learning Ranking vs. Modeling Relevance. *In Proceedings of the 39th Annual Hawaii International Conference on System Sciences*.

Skomorowski, J. and Vechtomova, O. (2007). Ad hoc retrieval of documents with topical opinion. *In Proceedings of the 29th European Conference on Information Retrieval*.

Sparck Jones, K., Walker, S. and Robertson, S. E. (1998). A probabilistic model of information retrieval: development and status.

Sparck Jones, K., Walker, S. and Robertson, S. E. (2000). A probabilistic model of information retrieval: development and comparative experiments. *Information Processing and Management*, 36(6), 779-808, 809-840.

Sparck Jones, K. and Robertson, S. E. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science*, 27, 129-146.

Toral, A., Muñoz, R. and Monachini, M. (2008). Named Entity WordNet. *In Proceedings of the 6th Conference on Language Resources and Evaluation,* 132-145.

Toral, A. (2005). DRAMNERI: a free knowledge based tool to Named Entity Recognition. *In Proceedings of the 1st Free Software Technologies Conference*, pp. 27-32.

Turney, P. D. (2002). Thumbs up or thumbs down? Semantic orientation applied to unsupervised classification of reviews*. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics,* 417-424.

Turpin, A. and Hersh, W. R (2001). Why batch and user evaluations do not give the same results. *In proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval*, 225-231.

van Rijsbergen, C. J. (1977). A theoretical basis for the use of co-occurrence data in information Retrieval, *Information Process Manage,* 33(2), 106–119.

van Rijsbergen, C. J., Harper, D. J. and Porter, M. F. (1978). The selection of good search items. *Information Process Manage*, 17(2), 77–91.

Vechtomova, O. (2007). Using subjective adjectives in opinion retrieval from blogs. *In Proceedings of the 16th Text Retrieval Conference*, 78-83.

Voorhees, E. M., and Buckley, C. (2004). Retrieval evaluation with incomplete information. *SIGIR Conference*, 27-42.

Wiebe, J. M. (2000). Learning subjective adjectives from corpora. *In proceedings of the 17th national Conference on Artificial Intelligence*, 217-223.

Wiebe, J. M., Bruce, R. F. and O'Hara, T. (1999). Development and use of a gold standard data set for subjectivity classifications. *In Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics,* 246–253.

Xu, J. and Croft, W. B. (1999). Cluster-based language models for distributed retrieval. *In Proceedings of the 22nd International ACM SIGIR Conference on Research and Development Information Retrieval (SIGIR 99),* pp. 254-261.

Yang, K., Yu, N. and Zhang, H. (2007). WIDIT in TREC 2007 Blog Track: combining lexicon-based methods to detect opinionated blogs. *In proceedings of the 16th Text Retrieval Conference,* 78-84.

Yi, J., Nasukawa, T., Bunescu, R. and Niblack, W. (2003). Sentiment analyzer: extracting sentiments about a given topic using natural language processing techniques. *In Proceedings of the 3rd IEEE International Conference on Data Mining,* 125-131.

Zhang, W. and Yu, C. (2006). UIC at TREC 2006 Blog Track. *In Proceedings of the Fifteenth Text REtrieval Conference,* 32-36.

Yang, K., Yu, N., A.Valerio, and H. Zhang (2006). WIDIT in TREC 2006 Blog Track. In Proceedings of the Fifteenth Text REtrieval Conference, 38.

Zhou, G., Joshi, H. and Bayrak, C. (2007). Topic categorization for relevancy and opinion detection. *In Proceedings of the 16th Text Retrieval Conference*, 45-52.

Appendices

Appendix A: BM25 Parameters Evaluation Results

MAP		k1							
		0.5	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**
b	**0.1**	0.2970	0.2973	0.2978	0.2982	0.2987	0.2991	0.2989	0.2983
	0.2	0.2990	0.2995	0.3000	0.3008	0.3013	0.3014	0.3015	0.3013
	0.3	0.2993	0.2999	0.3006	0.3014	0.3017	0.3019	0.3018	0.3014
	0.4	0.2969	0.2975	0.2979	0.2989	0.2990	0.2993	0.2991	0.2987
	0.5	0.2934	0.2939	0.2946	0.2952	0.2956	0.2958	0.2958	0.2953
	0.6	0.2894	0.2898	0.2909	0.2915	0.2919	0.2919	0.2918	0.2914
	0.75	0.2814	0.2819	0.2828	0.2834	0.2837	0.2839	0.2839	0.2834

R-Prec		k1							
		0.5	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**
b	**0.1**	0.3748	0.3764	0.3774	0.3788	0.3788	0.3797	0.3796	0.3795
	0.2	0.3748	0.3749	0.3768	0.3769	0.3780	0.3788	0.3787	0.3792
	0.3	0.3747	0.3752	0.3766	0.3771	0.3786	0.3788	0.3788	0.3790
	0.4	0.3735	0.3746	0.3752	0.3762	0.3775	0.3784	0.3788	0.3786
	0.5	0.3729	0.3736	0.3741	0.3743	0.3752	0.3750	0.3757	0.3751
	0.6	0.3715	0.3729	0.3729	0.3731	0.3734	0.3737	0.3735	0.3725
	0.75	0.3645	0.3654	0.3670	0.3671	0.3682	0.3681	0.3683	0.3665

Bpref		k1							
		0.5	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**
b	**0.1**	0.4083	0.4087	0.4091	0.4091	0.4094	0.4096	0.4094	0.4090
	0.2	0.4094	0.4097	0.4099	0.4102	0.4106	0.4108	0.4110	0.4112
	0.3	0.4083	0.4088	0.4092	0.4096	0.4099	0.4104	0.4106	0.4105
	0.4	0.4062	0.4066	0.4067	0.4076	0.4076	0.4080	0.4082	0.4081
	0.5	0.4034	0.4039	0.4042	0.4048	0.4050	0.4056	0.4059	0.4057
	0.6	0.4001	0.4008	0.4015	0.4019	0.4022	0.4024	0.4025	0.4024
	0.75	0.3940	0.3941	0.3942	0.3946	0.3950	0.3956	0.3959	0.3960

P10		k1							
		0.5	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**
b	**0.1**	0.6102	0.6102	0.6102	0.6102	0.6102	0.6122	0.6143	0.6122
	0.2	0.6245	0.6224	0.6224	0.6204	0.6224	0.6224	0.6306	0.6327
	0.3	0.6163	0.6143	0.6143	0.6143	0.6163	0.6163	0.6204	0.6204
	0.4	0.6020	0.6020	0.6020	0.6041	0.6061	0.6082	0.6061	0.6061
	0.5	0.6041	0.6041	0.6082	0.6082	0.6082	0.6082	0.6082	0.6082

	0.6	0.6122	0.6143	0.6163	0.6184	0.6184	0.6163	0.6163	0.6143
	0.75	0.6000	0.6020	0.6000	0.6000	0.6000	0.6020	0.6020	0.6041

Table A.1: The Blog 06 topical relevance evaluation results with different values of b and k_1 using measures of MAP, R-Prec, bpref, and P@10.

MAP		**k1**							
		0.5	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**
b	**0.1**	0.1956	0.1958	0.1961	0.1963	0.1966	0.1966	0.1962	0.1955
	0.2	0.1937	0.1940	0.1941	0.1946	0.1947	0.1944	0.1944	0.1939
	0.3	0.1909	0.1912	0.1913	0.1917	0.1917	0.1916	0.1913	0.1907
	0.4	0.1869	0.1872	0.1874	0.1878	0.1877	0.1876	0.1873	0.1866
	0.5	0.1827	0.1829	0.1831	0.1834	0.1834	0.1833	0.1831	0.1824
	0.6	0.1782	0.1783	0.1787	0.1790	0.1789	0.1787	0.1784	0.1778
	0.75	0.1713	0.1713	0.1718	0.1721	0.1721	0.1721	0.1719	0.1713

R-Prec		**k1**							
		0.5	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**
b	**0.1**	0.2694	0.2694	0.2694	0.2694	0.2692	0.2694	0.2690	0.2691
	0.2	0.2674	0.2671	0.2671	0.2664	0.2665	0.2667	0.2675	0.2676
	0.3	0.2657	0.2653	0.2654	0.2652	0.2656	0.2661	0.2660	0.2662
	0.4	0.2631	0.2634	0.2630	0.2633	0.2639	0.2641	0.2637	0.2631
	0.5	0.2605	0.2608	0.2601	0.2595	0.2603	0.2609	0.2607	0.2603
	0.6	0.2589	0.2580	0.2577	0.2581	0.2587	0.2595	0.2594	0.2590
	0.75	0.2536	0.2538	0.2539	0.2542	0.2548	0.2546	0.2545	0.2549

Bpref		**k1**							
		0.5	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**
b	**0.1**	0.2630	0.2631	0.2631	0.2630	0.2629	0.2628	0.2625	0.2621
	0.2	0.2606	0.2606	0.2606	0.2606	0.2605	0.2602	0.2602	0.2600
	0.3	0.2583	0.2583	0.2583	0.2583	0.2583	0.2581	0.2583	0.2574
	0.4	0.2552	0.2551	0.2553	0.2553	0.2552	0.2551	0.2548	0.2542
	0.5	0.2524	0.2522	0.2522	0.2520	0.2521	0.2520	0.2518	0.2511
	0.6	0.2488	0.2488	0.2488	0.2488	0.2486	0.2483	0.2479	0.2472
	0.75	0.2434	0.2426	0.2425	0.2427	0.2426	0.2426	0.2425	0.2420

P10		**k1**							
		0.5	**0.75**	**1**	**1.25**	**1.5**	**1.75**	**2**	**2.5**
b	**0.1**	0.4204	0.4204	0.4204	0.4224	0.4224	0.4245	0.4245	0.4245
	0.2	0.4327	0.4306	0.4286	0.4286	0.4286	0.4286	0.4347	0.4367
	0.3	0.4184	0.4184	0.4184	0.4184	0.4184	0.4184	0.4224	0.4224
	0.4	0.3918	0.3918	0.3939	0.3939	0.3959	0.3980	0.3959	0.3959

	0.5	0.3939	0.3939	0.3939	0.3939	0.3939	0.3939	0.3939	0.3918
	0.6	0.3837	0.3837	0.3837	0.3837	0.3837	0.3816	0.3816	0.3796
	0.75	0.3612	0.3612	0.3592	0.3571	0.3571	0.3592	0.3592	0.3571

Table A.2: The Blog 06 opinion relevance evaluation results with different values of b and k_1 using measures of MAP, R-Prec, bpref, and P@10.

Appendix B: Query Categorization

Person	**Event**	**Product**	**Organisation**	**Media / Art (TV show/film/book /song/album)**	**Miscellaneous**
852:larry summers	853:state of the union	856:macbook pro	863:netflix	851:March of the Penguins	859:letting india into the club?
854:Ann Coulter	861:mardi gras	862:blackberry	866:Whole Foods	858:super bowl ads	865:basque
855:abramoff bush	867:cheney hunting	879:hybrid car	877:sonic food industry	860:arrested development	869:muhammad cartoon
857:jon stewart	868:joint strike fighter	883:heineken	884:Qualcomm	864:colbert report	878:jihad
870:barry bonds	887:World Trade Organization	885:shimano	882:seahawks	872:brokeback mountain	889:scientology
871:cindy sheehan		888:audi		875:american idol	890:olympics
873:bruce bartlett		891:intel		876:life on mars	894:board chess
874:coretta scott king		893:zyrtec		881:Fox News Report	896:global warming
880:natalie portman		900:mcdonalds		886:west wing	898:Business Intelligence Resources
892:Jim Moran				895:Oprah	899:cholesterol
897:ariel sharon					

Table B.1: Query categories for Blog 06 topics

Person	**Event**	**Product**	**Organisation**	**Media (TV Show/film/book/ song/album)**	**Miscellaneous**
903:Steve Jobs	905:king funeral	901:jstor	910:"Aperto Networks"	911:SCI FI CHANNEL	902:lactose gas
904:alterman	906:davos	909:Barilla	912:nasa	921:"Christianity Today"	918:varanasi
908:"carrie underwood"	907:brrreeeport	932:goobuntu	915:allianz	928:"big love"	927:oscar fashion
920:"andrew coyne"	913:sag awards	934:cointreau	916:dice.com		929:"brand manager"
922:"howard stern"	914:northernvoice	939:"Beggin Strips"	917:snopes		931:fort mcmurray
924:"mark driscoll"	923:challenger	944:"Opera Software" OR "Opera Browser" OR "Opera Mobile" OR "Opera Mini"	919:pfizer		933:"winter olympics"
935:mozart	925:mashup camp	946:tivo	926:hawthorne heights		942:lawful access
940:Lance Armstrong	936:grammys		930:ikea		943:censure
941:"teri hatcher"	938:"plug awards"		937:LexisNexis		945:bolivia
947:sasha cohen			948:sorbonne		
949:ford bell			950:Hitachi Data Systems		

Table B.2: Query categories for Blog 07 topics

Appendix C: Positive Normalized KLD of Subjective Adjectives

good	1	violent	0.06704	large	0.0375
right	0.71324	rich	0.06409	inappropriate	0.03682
white	0.68058	sound	0.06368	responsible	0.03571
content	0.63438	obvious	0.0575	negative	0.03569
great	0.39054	angry	0.0572	independent	0.03544
little	0.38891	huge	0.05688	dumb	0.03501
TRUE	0.29471	thorough	0.05681	popular	0.03491
sure	0.28939	simple	0.05678	friendly	0.03435
big	0.26925	powerful	0.05503	welcome	0.03403
offensive	0.24721	fair	0.05491	modest	0.03385
wrong	0.21368	controversial	0.05428	proud	0.03374
live	0.21095	strong	0.05407	slow	0.03331
pretty	0.19396	smart	0.05405	lazy	0.03328
kind	0.18884	ridiculous	0.05315	lame	0.0332
lost	0.18217	perfect	0.05191	fake	0.03248
bad	0.16467	positive	0.05175	pathetic	0.0323
sorry	0.16338	brilliant	0.0498	moving	0.03216
top	0.15221	open	0.04971	strange	0.03211
funny	0.15132	safe	0.04886	decent	0.03199
fine	0.14127	honest	0.04858	ugly	0.03167
poor	0.13335	deep	0.0477	silly	0.03166
stupid	0.11786	holy	0.04755	growing	0.03084
alert	0.11621	interested	0.04715	terrible	0.03081
fun	0.10819	crazy	0.0467	absurd	0.03067
happy	0.10696	limited	0.0448	dark	0.03053
original	0.10178	willing	0.04429	difficult	0.03027
nice	0.10032	wonderful	0.04381	horrible	0.0302
sad	0.09999	guilty	0.0437	leading	0.02978
dead	0.09963	hilarious	0.04358	hot	0.02957
straight	0.09933	central	0.04285	satisfying	0.02918
handy	0.09824	frank	0.04254	suspect	0.02869
illegal	0.09411	outstanding	0.04244	uncomfortable	0.02867
hard	0.09344	mad	0.04218	innocent	0.02842
full	0.08867	ready	0.04172	awesome	0.02817
clear	0.0866	appropriate	0.04171	boring	0.02797
FALSE	0.08575	glad	0.04167	excellent	0.02796
drunk	0.08344	unlimited	0.04145	remarkable	0.02751
okay	0.08216	tired	0.04101	unfortunate	0.02726
important	0.08175	dear	0.04093	rude	0.02717
black	0.08	dangerous	0.04068	corrupt	0.02716
hurt	0.07744	afraid	0.04045	rare	0.02713
able	0.07695	proper	0.03933	reasonable	0.02713
democratic	0.07603	sick	0.03916	firm	0.02688
beautiful	0.07456	major	0.03874	noble	0.02666
evil	0.07108	correct	0.03864	worthy	0.02602
broke	0.06799	abusive	0.03798	thoughtful	0.02599

odd	0.02598	necessary	0.01658	witty	0.01225
fit	0.02536	amusing	0.01653	valid	0.01218
fresh	0.02533	sweet	0.01653	misguided	0.01205
careful	0.02528	drunken	0.01642	optimistic	0.01201
busy	0.02488	grave	0.01633	profound	0.01194
awful	0.02439	misleading	0.01632	preferable	0.0119
normal	0.02438	unlawful	0.01608	sympathetic	0.01175
outrageous	0.02419	modern	0.01607	negligent	0.01173
direct	0.02412	inferior	0.016	bitter	0.01165
super	0.02397	lucky	0.01596	entertaining	0.01162
famous	0.02355	legitimate	0.01594	irrelevant	0.01151
fantastic	0.02333	annoying	0.01585	prepared	0.01147
sacred	0.02326	cute	0.0158	specific	0.01137
minor	0.02276	superior	0.01567	intriguing	0.01136
dirty	0.02274	successful	0.01563	predictable	0.01135
quiet	0.02261	arresting	0.01554	loyal	0.01131
sharp	0.02225	lovely	0.01548	unusual	0.01128
respectful	0.02223	clever	0.01535	sinister	0.0111
weak	0.02197	reckless	0.01508	notorious	0.01106
insane	0.02191	empty	0.01505	lax	0.01104
prominent	0.02119	extensive	0.01493	useful	0.01097
solid	0.02098	clean	0.01489	calm	0.01094
quick	0.02077	delightful	0.01461	fitting	0.01087
pure	0.02074	rational	0.01457	harsh	0.01085
brave	0.02055	bizarre	0.01452	mock	0.01084
declined	0.02031	peaceful	0.0145	eager	0.01061
irresponsible	0.02029	vast	0.0145	desperate	0.01058
talented	0.02007	undisputed	0.01445	questionable	0.0105
ignorant	0.01987	bright	0.01443	deceased	0.0104
impossible	0.01979	frequent	0.0144	superb	0.01037
accurate	0.01975	cold	0.01395	bold	0.01035
relevant	0.01968	rotten	0.01395	faithful	0.01034
key	0.01908	painful	0.01393	petty	0.01032
universal	0.01903	complex	0.01381	helpful	0.01029
tragic	0.01895	immoral	0.01359	vigorous	0.01028
sensitive	0.01873	fell	0.01345	unfair	0.01026
careless	0.01823	cruel	0.01323	avid	0.01018
engaging	0.0182	exciting	0.0132	tumultuous	0.00994
swift	0.01789	vicious	0.01292	consistent	0.00965
genuine	0.01763	tall	0.0129	captive	0.00965
stable	0.01746	incompetent	0.01287	influential	0.00963
acceptable	0.01725	ill	0.01269	attractive	0.00955
confusing	0.01724	compassionate	0.01266	faulty	0.00951
ashamed	0.01723	capable	0.01266	unhappy	0.00944
comfortable	0.01707	expensive	0.01265	worthless	0.00943
spectacular	0.01704	unethical	0.01264	complicated	0.00943
experienced	0.01697	credible	0.01262	easy	0.00936
compact	0.01691	suspicious	0.01246	lush	0.00935
familiar	0.0167	hypocritical	0.01241	extreme	0.00927
unacceptable	0.01662	threatening	0.01227	nasty	0.00924

cynical	0.00923	courageous	0.00734	straightforward	0.00575
premium	0.00922	convenient	0.00731	fictional	0.00575
endless	0.00917	bankrupt	0.0073	autonomous	0.00572
breathtaking	0.00911	encouraging	0.00726	distasteful	0.00571
disturbing	0.00907	glorious	0.00723	vague	0.00571
enormous	0.00901	disappointing	0.00697	cozy	0.00568
wealthy	0.00898	revealing	0.00695	risky	0.00563
deadly	0.00897	steady	0.00694	unpleasant	0.00563
useless	0.00896	timely	0.00685	secretive	0.00561
insensitive	0.00884	damaging	0.0068	obnoxious	0.0056
arrogant	0.00883	foolish	0.0068	maverick	0.0056
broad	0.00875	disastrous	0.00679	notable	0.00558
shallow	0.00875	fabulous	0.00679	fugitive	0.00557
significant	0.0087	naive	0.00675	promising	0.00555
challenging	0.00857	best-known	0.00672	frustrating	0.00553
bloody	0.00857	hysterical	0.00672	tremendous	0.00547
dramatic	0.00856	intimate	0.00671	tolerant	0.00545
scary	0.00854	precious	0.0067	essential	0.00544
unwilling	0.00842	edgy	0.0067	elaborate	0.00543
upscale	0.0084	self-serving	0.00665	wide-ranging	0.00541
murderous	0.00837	unexpected	0.00664	stellar	0.00539
active	0.00834	qualified	0.00661	generous	0.00535
sour	0.00833	foul	0.00656	bogus	0.00535
intense	0.0083	meaningless	0.00653	wise	0.00534
gorgeous	0.00829	enthusiastic	0.00652	ludicrous	0.00531
understandable	0.00828	obscure	0.00649	rough	0.00528
outspoken	0.00818	rousing	0.00646	dishonest	0.00528
compelling	0.00818	unable	0.00644	animated	0.00526
selfish	0.00808	supportive	0.00643	innocuous	0.00525
intensive	0.00798	grateful	0.00641	magic	0.00518
disingenuous	0.00791	trivial	0.00636	selective	0.00517
widespread	0.0079	biased	0.00636	crude	0.00517
terrific	0.0079	lethal	0.00636	incapable	0.00516
inflammatory	0.0079	spirited	0.00634	phony	0.00513
sensible	0.00788	gentle	0.0063	logical	0.0051
sober	0.00782	stiff	0.00627	awkward	0.00509
impressive	0.00779	abandoned	0.00627	grotesque	0.00505
appalling	0.00778	convincing	0.00624	giant	0.00502
polite	0.00775	inflationary	0.00624	little-known	0.00492
humorous	0.00774	destructive	0.00605	vivid	0.00487
hopeful	0.00771	vulgar	0.006	bland	0.00487
fatal	0.00769	affected	0.00597	pale	0.00487
sincere	0.00768	savage	0.00597	sweeping	0.00481
charming	0.00767	intelligent	0.00595	obsessive	0.00479
plausible	0.00767	tiny	0.00591	noteworthy	0.00479
deaf	0.00761	solemn	0.00591	resigned	0.00479
smug	0.00756	worthwhile	0.0059	lucrative	0.00479
sleepy	0.00746	doubtful	0.00587	tedious	0.00476
blatant	0.00744	disruptive	0.00583	primary	0.00475
favorable	0.00736	helpless	0.00578	shiny	0.00474

greedy	0.00474	poisonous	0.00392	puzzling	0.0032
established	0.00473	haunting	0.00392	ominous	0.00317
unnatural	0.00472	astute	0.00391	punishable	0.00316
upbeat	0.00472	confident	0.00389	perverse	0.00313
first-class	0.0047	unstable	0.00388	delicious	0.00311
extremist	0.00468	regrettable	0.00385	gifted	0.0031
humane	0.00464	profitable	0.00383	egregious	0.00308
depressing	0.00464	monstrous	0.00382	unreasonable	0.00308
refreshing	0.00461	magnificent	0.0038	potent	0.00308
rocky	0.00461	substantive	0.0038	lame-duck	0.00305
baseless	0.0046	insular	0.0038	unsuspecting	0.00303
fraudulent	0.00458	constructive	0.00377	dim	0.00302
irrational	0.00458	earnest	0.00376	weary	0.00302
lonely	0.00457	passionate	0.00373	truthful	0.00302
involuntary	0.00456	novel	0.00371	improper	0.003
meaningful	0.00452	impatient	0.00371	shrewd	0.00299
hesitant	0.00452	restricted	0.0037	coherent	0.00298
superficial	0.00452	obscene	0.00369	traumatic	0.00298
balanced	0.00439	unfinished	0.00369	untrue	0.00298
precise	0.00438	hostile	0.00368	overblown	0.00297
chilly	0.00436	unsatisfactory	0.00366	crowded	0.00295
believable	0.00435	secure	0.00365	deceptive	0.00295
giddy	0.00434	irresistible	0.00363	stubborn	0.00295
competent	0.00433	epic	0.00358	marvelous	0.00295
staunch	0.00429	triumphant	0.00358	inflated	0.00295
seasoned	0.00427	harmless	0.00356	idle	0.00293
celebrated	0.00425	humiliating	0.00354	chaotic	0.00289
rapid	0.00424	juvenile	0.00354	intact	0.00289
contentious	0.00423	admirable	0.00354	defunct	0.00284
high-ranking	0.00421	unconstitutional	0.00353	pedestrian	0.00282
unclear	0.00419	patient	0.00351	embattled	0.0028
mighty	0.00415	distressing	0.00351	perceptive	0.00278
mindless	0.00413	cautious	0.00347	unanimous	0.00274
flawed	0.00413	inaccurate	0.00347	preposterous	0.00274
vain	0.00413	benign	0.00347	ample	0.00269
eloquent	0.00413	slack	0.00347	jealous	0.00269
lousy	0.00413	articulate	0.00346	immense	0.00267
covert	0.00412	dull	0.00343	illegitimate	0.00266
divisive	0.0041	needless	0.00341	overdue	0.00265
sunny	0.00405	hapless	0.00341	uneven	0.00264
expressive	0.00405	problematic	0.0034	intuitive	0.00263
implicit	0.00398	uninformed	0.0034	unparalleled	0.00259
magical	0.00398	enjoyable	0.0034	ruthless	0.00259
reluctant	0.00395	astounding	0.00339	appealing	0.00258
enhanced	0.00395	heroic	0.00337	valuable	0.00257
exhaustive	0.00394	sophisticated	0.00336	self-destructive	0.00256
fiery	0.00393	well-established	0.00332	undetermined	0.00256
overt	0.00393	murky	0.00331	moody	0.00255
realistic	0.00392	splitting	0.00328	unwise	0.00255
responsive	0.00392	clumsy	0.00322	pleasant	0.00255

mediocre	0.00253	persistent	0.00212	bloated	0.00172
now-defunct	0.00253	ambiguous	0.00208	charismatic	0.00172
gigantic	0.00253	malicious	0.00207	fearful	0.00172
unpopular	0.00252	wrongful	0.00207	respectable	0.00172
infamous	0.00252	palatable	0.00206	famed	0.00171
advanced	0.00252	indefinite	0.00202	colossal	0.00171
thrilling	0.00251	touchy	0.00202	precarious	0.00171
feeble	0.00245	ecstatic	0.00202	flimsy	0.0017
amiable	0.00245	best-selling	0.00202	insufficient	0.00169
disinterested	0.00245	arbitrary	0.00202	nonexistent	0.00166
toxic	0.00244	unavailable	0.00201	objective	0.00165
subversive	0.00244	neat	0.002	repressive	0.00164
moot	0.00244	poetic	0.00199	scandalous	0.00164
inadequate	0.00244	polished	0.00199	insignificant	0.00164
reflective	0.00243	hungry	0.00199	passive	0.00164
dismal	0.00238	unpredictable	0.00199	mundane	0.00163
ardent	0.00238	undecided	0.00197	noisy	0.00163
skilled	0.00236	sloppy	0.00196	horrendous	0.00162
wide-open	0.00236	prompt	0.00196	relentless	0.00162
vigilant	0.00235	fierce	0.00196	tense	0.00161
explosive	0.00234	sluggish	0.00193	lawful	0.0016
indirect	0.00232	substantial	0.00193	trim	0.0016
unjust	0.00232	high-powered	0.00192	concerted	0.00158
agonizing	0.00231	staid	0.0019	depressed	0.00157
unresolved	0.00231	dogged	0.00188	eminent	0.00157
virulent	0.00231	handsome	0.00185	speedy	0.00157
prudent	0.00231	usable	0.00185	evasive	0.00156
extraneous	0.0023	draconian	0.00185	far-fetched	0.00155
opportunistic	0.00229	ill-fated	0.00185	advantageous	0.00155
frantic	0.00229	fervent	0.00185	well-received	0.00155
brutal	0.00229	abundant	0.00185	smooth	0.00154
savvy	0.00228	luxurious	0.00184	powerless	0.00154
instructive	0.00226	systematic	0.00184	uneasy	0.00154
cheerful	0.00226	keen	0.00183	hollow	0.00154
fortunate	0.00225	illiterate	0.00183	halt	0.0015
divine	0.00225	intolerable	0.00182	conclusive	0.0015
somber	0.00224	monumental	0.00179	distinctive	0.0015
wide	0.00224	ripe	0.00179	aghast	0.00149
outdated	0.00222	slow-moving	0.00179	knowledgeable	0.00149
uniform	0.00222	quaint	0.00178	compulsory	0.00148
suicidal	0.00221	reliable	0.00177	unsavory	0.00148
fuzzy	0.00218	stressful	0.00177	adept	0.00148
grandiose	0.00217	forthright	0.00177	immune	0.00148
unprofitable	0.00217	full-scale	0.00177	unsafe	0.00146
youthful	0.00217	literate	0.00177	rewarding	0.00146
authoritarian	0.00216	stodgy	0.00177	cautionary	0.00144
obsolete	0.00216	catastrophic	0.00174	less-expensive	0.00142
unfulfilled	0.00215	fickle	0.00173	victorious	0.00142
imperial	0.00214	unfavorable	0.00173	peripheral	0.0014
sketchy	0.00214	permissible	0.00173	banal	0.0014

prestigious	0.00138	second-class	0.00111	illicit	0.00086
infected	0.00137	skittish	0.00111	affluent	0.00086
instrumental	0.00136	inordinate	0.00111	eternal	0.00085
capricious	0.00135	impeccable	0.00111	objectionable	0.00085
undocumented	0.00135	disabled	0.00111	scholarly	0.00084
brisk	0.00135	high-performance	0.00111	tidy	0.00084
corrective	0.00135	discouraging	0.00111	unconditional	0.00084
hard-line	0.00134	innovative	0.00109	tangled	0.00084
hardy	0.00134	inconsistent	0.00108	adequate	0.00083
mindful	0.00134	vulnerable	0.00107	desirable	0.00083
shy	0.00133	concrete	0.00107	jobless	0.00082
convoluted	0.00133	caustic	0.00107	adversarial	0.00082
unconventional	0.00133	dreary	0.00107	cutthroat	0.00082
candid	0.00132	insecure	0.00107	irate	0.00082
impartial	0.00132	principled	0.00107	spurious	0.00081
invisible	0.0013	loath	0.00106	stringent	0.00081
phenomenal	0.0013	unforeseen	0.00106	tenuous	0.00081
cash-strapped	0.0013	bemused	0.00106	memorable	0.00081
discreet	0.0013	courteous	0.00106	flagrant	0.00081
funded	0.00128	preeminent	0.00104	well-connected	0.0008
erratic	0.00127	marginal	0.00103	unconfirmed	0.0008
repetitive	0.00127	totalitarian	0.00103	redundant	0.00079
adverse	0.00127	simplistic	0.00102	ferocious	0.00078
treacherous	0.00126	unrealistic	0.00102	unequivocal	0.00078
distraught	0.00125	sparkling	0.00102	elegant	0.00077
hopeless	0.00125	unencumbered	0.00102	choppy	0.00077
manipulative	0.00123	strident	0.00101	sturdy	0.00077
delicate	0.00122	fateful	0.001	spacious	0.00077
unfettered	0.00121	contagious	0.00098	timid	0.00076
sham	0.00121	well-intentioned	0.00098	high-cost	0.00076
self-defeating	0.0012	assertive	0.00097	indiscriminate	0.00076
unjustified	0.00119	glamorous	0.00097	superfluous	0.00076
adamant	0.00119	enterprising	0.00097	workable	0.00076
large-scale	0.00119	narrow	0.00095	wee	0.00075
bleak	0.00118	decisive	0.00095	deficient	0.00074
impotent	0.00118	unsuccessful	0.00095	first-rate	0.00074
conciliatory	0.00118	incompatible	0.00094	booming	0.00073
prolific	0.00118	nifty	0.00091	willful	0.00073
defective	0.00118	cordial	0.00091	excessive	0.00072
graceful	0.00118	wry	0.00091	reputable	0.00072
rife	0.00118	stuffed	0.00091	slim	0.00071
splendid	0.00116	heady	0.0009	unemployed	0.00071
definite	0.00116	negligible	0.00089	fertile	0.00071
inept	0.00115	full-fledged	0.00089	inspirational	0.00069
unsupported	0.00115	soft-spoken	0.00089	miraculous	0.00069
playful	0.00115	uncompetitive	0.00089	unscrupulous	0.00068
forged	0.00115	debatable	0.00087	heavy-handed	0.00068
favored	0.00114	fleeting	0.00087	impersonal	0.00067
overzealous	0.00111	astronomical	0.00087	second-tier	0.00067
reticent	0.00111	full-blown	0.00087	unattractive	0.00067

unsound	0.00067	detrimental	0.00048	meticulous	0.00029
intermittent	0.00066	inefficient	0.00048	time-honored	0.00029
crippling	0.00066	dependable	0.00048	belated	0.00028
scant	0.00065	forceful	0.00047	troublesome	0.00028
authoritative	0.00065	plush	0.00046	open-ended	0.00028
pre-eminent	0.00064	lower-priced	0.00045	unwanted	0.00026
volatile	0.00064	resurgent	0.00045	uncollectible	0.00026
backward	0.00064	lenient	0.00044	unsophisticated	0.00026
far-reaching	0.00063	dire	0.00044	deflationary	0.00026
low-rated	0.00063	colorful	0.00044	spotty	0.00025
exhilarating	0.00063	handicapped	0.00044	ingenious	0.00025
sanguine	0.00063	better-known	0.00044	proportionate	0.00025
scrap	0.00063	frozen	0.00044	well-regarded	0.00025
complacent	0.00062	relaxed	0.00044	ill-advised	0.00024
unprepared	0.00062	confrontational	0.00041	imaginative	0.00024
flashy	0.00062	illogical	0.00041	offbeat	0.00024
fruitless	0.00062	restless	0.00041	uncertain	0.00024
wholesome	0.00062	unreliable	0.00041	unwarranted	0.00024
arcane	0.0006	incomplete	0.0004	icy	0.00024
pertinent	0.00059	pretentious	0.0004	perilous	0.00024
manifest	0.00059	unsustainable	0.0004	subdued	0.00024
unfit	0.00059	posh	0.0004	considerable	0.00023
frail	0.00059	shoddy	0.00039	foolproof	0.00023
distressed	0.00058	well-educated	0.00039	prodigious	0.00023
inexperienced	0.00058	abysmal	0.00037	seductive	0.00023
sustained	0.00057	indecisive	0.00037	fragile	0.00022
catchy	0.00057	undaunted	0.00037	slight	0.00022
conscientious	0.00057	hectic	0.00036	equitable	0.00022
well-informed	0.00057	frivolous	0.00036	pricey	0.00021
intrusive	0.00056	meager	0.00036	burdensome	0.0002
regressive	0.00056	untenable	0.00036	inactive	0.0002
hasty	0.00054	versatile	0.00036	uninsured	0.0002
orderly	0.00054	battered	0.00036	unworkable	0.0002
tenacious	0.00053	suitable	0.00035	vehement	0.0002
commensurate	0.00053	warm	0.00035	scenic	0.0002
streamlined	0.00053	redeeming	0.00034	life-threatening	0.0002
record-setting	0.00053	incorrect	0.00033	high-quality	0.0002
hefty	0.00053	oversized	0.00032	disproportionate	0.0002
dubious	0.00052	gritty	0.00032	stylish	0.0002
futile	0.00052	outgoing	0.00032	muddy	0.0002
divergent	0.00052	scaled-down	0.00032	ambivalent	0.0002
scarce	0.00052	lively	0.00032	skimpy	0.0002
ebullient	0.00051	feisty	0.00031	tepid	0.0002
underdeveloped	0.00051	penetrating	0.00031	self-sufficient	0.0002
turbulent	0.00049	stately	0.00031	fast-paced	0.0002
unwelcome	0.00049	tricky	0.00031	refined	0.00018
invaluable	0.00049	fractious	0.0003	unsure	0.00017
unrestricted	0.00049	panicky	0.0003	vociferous	0.00017
peculiar	0.00048	euphoric	0.0003	fluent	0.00017
allergic	0.00048	lesser-known	0.00029	worrisome	0.00017

rigid	0.00016	unfunded	4.7E-06
anemic	0.00016		
lukewarm	0.00016		
combative	0.00016		
indigent	0.00015		
disadvantaged	0.00014		
expansive	0.00013		
shortsighted	0.00012		
audacious	0.00012		
receptive	0.00012		
inflexible	0.00012		
substandard	0.00012		
consummate	0.00012		
opportune	0.00012		
expert	0.00012		
unhealthy	0.00011		
frigid	0.00011		
rosy	0.00011		
erroneous	0.0001		
indifferent	9.7E-05		
stormy	9.7E-05		
extravagant	8.9E-05		
susceptible	8.9E-05		
unspecified	8.9E-05		
autocratic	8.4E-05		
lethargic	8.4E-05		
affable	8.4E-05		
archaic	8.1E-05		
shabby	8.1E-05		
attentive	8.1E-05		
conspicuous	8.1E-05		
highest-ranking	8.1E-05		
visionary	7.4E-05		
costly	7E-05		
restrictive	6E-05		
hard-working	5.5E-05		
sporadic	5.1E-05		
unexplained	5.1E-05		
byzantine	4.6E-05		
cohesive	4.6E-05		
masterful	4.6E-05		
strenuous	4.2E-05		
illusory	4E-05		
lanky	4E-05		
seamless	3.8E-05		
hazardous	2.8E-05		
abnormal	1.6E-05		
ill-conceived	9.4E-06		
leery	9.4E-06		
stimulating	6.9E-06		

Appendix D: Top 100 Normalized KLD Scores of Collocates

cartoons	1	blogger	0.216538	clinton	0.102943
bush	0.995154	amp	0.209266	attack	0.10242
war	0.991646	macbook	0.200902	culture	0.101859
us	0.910338	google	0.199766	washington	0.100925
film	0.867677	mac	0.188852	style	0.100326
border	0.839879	real	0.187344	wireless	0.100128
muslim	0.814322	blockbuster	0.18414	whole	0.098993
netflix	0.740765	actually	0.175849	forces	0.097923
audi	0.705298	country	0.171523	heath	0.096843
comment	0.696006	paper	0.170478	liberals	0.09452
danish	0.652928	democrats	0.168618	underwood	0.094475
sheehan	0.648733	kill	0.166172	raquo	0.094233
brokeback	0.647777	won	0.164899	permanent	0.093353
comments	0.604934	republican	0.163573	reply	0.093261
movie	0.591175	son	0.161988	response	0.089705
post	0.568068	published	0.159181	enemy	0.089688
president	0.548525	world	0.159177	love	0.087464
cindy	0.518638	american	0.158974	white	0.086898
mountain	0.501882	content	0.150109	days	0.086125
link	0.491	george	0.148397	FALSE	0.086036
quot	0.412105	TRUE	0.148131	color	0.085393
prophet	0.395567	mother	0.14639	state	0.08422
don	0.388597	spain	0.145952	target	0.083926
muhammad	0.384916	republicans	0.141647	says	0.083834
only	0.380179	states	0.137487	guy	0.082422
email	0.344578	europe	0.131899	dvds	0.080773
alt	0.340224	else	0.128831	mean	0.080059
coulter	0.33814	iraq	0.12492	old	0.078435
god	0.325541	fuck	0.122732	trackback	0.078344
gay	0.300877	profile	0.122563	powerbook	0.07822
cartoon	0.298175	posten	0.119542	woman	0.078198
apple	0.295514	wing	0.119297	bomb	0.076641
abramoff	0.289216	doing	0.119216	feel	0.073836
idol	0.282887	jyllands	0.114628	mobile	0.073018
point	0.274517	archive	0.114079	jon	0.072282
editor	0.271386	carrie	0.113796	trying	0.072092
allah	0.261551	please	0.112892	blank	0.071924
ann	0.254146	start	0.112645	congress	0.071655
media	0.250802	soldiers	0.111876	mohammed	0.071512
blog	0.234529	king	0.109395	posted	0.071395
movies	0.23366	story	0.106084	remember	0.071082
read	0.228739	night	0.106026	fight	0.069765
title	0.225418	anti	0.105487	cowboy	0.069288
hunting	0.221191	stewart	0.10341	jake	0.069062
src	0.219423	thought	0.103323	queue	0.068293
vice	0.217585	feed	0.103249	casey	0.068097

public	0.067899	path	0.043645
speak	0.067038	men	0.043493
university	0.067006	nofollow	0.043482
weekend	0.066746	non	0.042883
makes	0.066723	points	0.042572
blogs	0.066343	purl	0.042205
saw	0.066255	thing	0.041851
dead	0.06577	katrina	0.04131
reading	0.065671	fair	0.040613
rest	0.065466	watching	0.040538
customers	0.064995	herself	0.040483
watch	0.06486	strict	0.040173
book	0.062722		
started	0.062438		
colbert	0.060461		
texas	0.060023		
magazine	0.059869		
jihad	0.059798		
chess	0.059719		
cartoonists	0.058972		
camp	0.056468		
politics	0.056075		
great	0.055324		
opinion	0.055222		
protest	0.055125		
shows	0.054608		
women	0.054241		
customer	0.053923		
anger	0.053132		
cartoonist	0.052838		
second	0.052469		
guys	0.051101		
dtd	0.051071		
review	0.050413		
political	0.050389		
court	0.049912		
words	0.049174		
bob	0.048571		
article	0.048421		
everyone	0.047084		
hear	0.047068		
roger	0.047022		
racist	0.046625		
belief	0.046537		
thread	0.046238		
especially	0.04506		
pay	0.044552		
yes	0.044312		
haven	0.044228		
imac	0.04421		