The Effect of Weighted Term Frequencies on Probabilistic Latent Semantic Term Relationships

Laurence A.F. Park and Kotagiri Ramamohanarao

ARC Centre for Perceptive and Intelligent Manchines in Complex Environments,
Department of Computer Science and Software Engineering,
The University of Melbourne, Australia

Abstract. Probabilistic latent semantic analysis (PLSA) is a method of calculating term relationships within a document set using term frequencies. It is well known within the information retrieval community that raw term frequencies contain various biases that affect the precision of the retrieval system. Weighting schemes, such as BM25, have been developed in order to remove such biases and hence improve the overall quality of results from the retrieval system. We hypothesised that the biases found within raw term frequencies also affect the calculation of term relationships performed during PLSA. By using portions of the BM25 probabilistic weighting scheme, we have shown that applying weights to the raw term frequencies before performing PLSA leads to a significant increase in precision at 10 documents and average reciprocal rank. When using the BM25 weighted PLSA information in the form of a thesaurus, we achieved an average 8% increase in precision. Our thesaurus method was also compared to pseudo-relevance feedback and a co-occurrence thesaurus, both using BM25 weights. Precision results showed that the probabilistic latent semantic thesaurus using BM25 weights outperformed each method in terms of precision at 10 documents and average reciprocal rank.

Keywords: probabilistic latent semantic analysis, probabilistic model, information retrieval.

1 Introduction

For most information retrieval systems, a text document is a sequence of independent terms. Through further analysis of the document set, we are able to find clusters of terms that are related to each other; this process is considered to be the discovery of hidden topics. When given a collection of text documents, latent semantic analysis (LSA) [2] or probabilistic latent semantic analysis (PLSA) [3] are used to discover term relationships to hidden topics within the document set and hence relationships to other terms within the document set. The term relationships are calculated using the term frequencies found within a set of documents. Therefore, the term relationships are document set specific and are used to assist the increase of precision during the information retrieval process.

A. Amir, A. Turpin, and A. Moffat (Eds.): SPIRE 2008, LNCS 5280, pp. 63–74, 2008.

Latent semantic indexing, uses latent semantic analysis to construct an index based on relationships between the documents, terms and calculated topics. The process involves representing each document and term as a set of topics; when a query is provided, the documents with the most related topics to the query topics are considered the most relevant. It can be shown that this process is a mixture of term expansion using the latent semantic term relationships and document retrieval using a document-term frequency index [5, 7].

Recent experiments have shown that we are able to store probabilistic latent semantic information in a thesaurus and hence separate it from the document index [6, 8] This separation was shown to provide many benefits, including faster query times and using much less storage space when compared to a latent semantic index.

So far, probabilistic latent semantic term relationships have only been calculated using the raw frequency counts of each term in each document. It is well known that there are many biasing factors found with raw term frequency counts and there have been many research experiments performed by the information retrieval community in order to understand and remove these biases [1, 4]; the state of the art being BM25. This method is a term frequency weighting scheme that tries to remove any biases using probabilistic analysis of the document set.

We believe that the biasing factors found in raw term frequencies that disrupt the information retrieval process, also affect the term relationship calculations when using probabilistic latent semantic analysis. Therefore, we hypothesise that the term relationships obtained using PLSA will be more effective if calculated using weighted term frequencies rather than raw term frequencies. To examine this hypothesis, we will use the probabilistic latent semantic thesaurus, since it is able to isolate the term relationships and the effect the term weighting has on them.

This paper provides the following important contributions:

- An analysis of the effects of document and term weights on the PLSA term relationships through examination of retrieval results.
- A comparison of weighted PLSA to BM25 pseudo-relevence feedback and co-occurrence thesaurus term expansions methods.

In this document we will analyse the effectiveness of PLSA calculated term relationships when using the BM25 weighing scheme to weight our term frequencies. This will be compared to PLSA term relationships using raw term frequencies. The article will proceed as follows: section 2 will review the concept of PLSA and how it is used to discover hidden term relationships. Section 3 examines the bias found in term frequencies, how we can reduce their effect using BM25 and how to apply these effects to PLSA. Finally, we will examine the experiments performed and discuss the results in section 4.

2 Latent Semantic Analysis

Before we can begin our analysis of the effect of term weights on the PLSA term relationships, we must explain how the term relationships are calculated and how we can extract them from the latent semantic analysis process. In this section we will examine the latent semantic analysis concept.

2.1 Document Retrieval

The process in which the idea is transferred from the author's mind to the written article and then to the reader's mind, is a very lossy process. If we were able to model this process, then we would be able develop better methods of transferring our ideas to paper and also better methods of transferring ideas from paper to our own minds. Information retrieval systems try to model the former process in order to calculate which ideas are present in a document. Once the content of a document is known, the retrieval system can calculate better relevance judgements when given a query.

A basic document retrieval system comprises an inverted index containing the terms that are found in each document and an application to extract these values and compute document scores based on a provided query. When a query is given, the lists of documents associated to each query term are extracted from the index and combined using a document score function such as:

$$s(d, Q) = \sum_{t \in Q} w_{d,t} w_t w_{q,t} \tag{1}$$

where $s(d, Q)$ is the document score of document d given the set of query terms Q, $w_{d,t}$ is the document-term weight, w_t is the term weight, and $w_{q,t}$ is the query-term weight. Each of the weight values $w_{d,t}$, w_t and $w_{q,t}$ are based on $f_{d,t}$, f_t and $f_{q,t}$ respectively, where $f_{d,t}$ is the frequency of term t in document d, f_t is the number of documents term t appears in, and $f_{q,t}$ is the frequency of term t in query q.

Equation 1 shows us that document retrieval methods, which use a document-term index containing term frequencies, base their document score calculation on the occurrence of the user supplied query terms in each document. This allows the retrieval system to provide fast query times and use a conservative amount of storage, but the model suggests that all of the terms in the document set are independent of each other. For example, a search for "baby clothes" will return documents containing the terms "baby" or "clothes", but not provide documents containing related terms such as "infant", or "suits". This model assumes that authors write documents in the following manner:

1. the idea is constructed in the author's mind
2. specific terms are chosen from the term pool to express the idea on paper.

This model is shown in figure 1. Note that in this model, if other terms are chosen for the document, it would express a different idea because each of the terms are assumed independent of each other. We can see that this model does not reflect the actual process that an author does use to write a document.

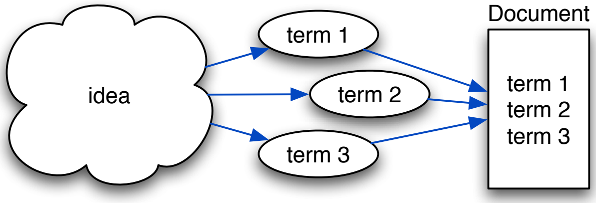

Fig. 1. A naïve document creation model. The author chooses specific terms for the document. If different terms are chosen the document will not convey the same message. This is the model used by retrieval systems that assume all terms are independent of each other (such as an inverted index of terms).

2.2 Latent Topics

We have seen in the previous section that the document retrieval model implies a poor document creation model. To make the model more realistic, we introduce an intermediate stage where the author chooses topics from a set of independent topics, to represent the document. Each of the topics contains a set of associated terms which are then chosen to include in the document. The process becomes:

1. the idea is constructed in the authors mind
2. specific topics are chosen from the topic pool to express the author's idea
3. for each topic, terms are chosen from the associated topic term pool to express the idea on paper

where the topic term pool is a set of terms that are related to the associated topic. Note that although the topics are independent, the associated terms may appear in many topics due to the synonomy found in many terms. This model is shown in figure 2. The final step allows the author to choose any of the terms associated to the selected topic to use within the document. This process suggests that as long as two documents contain the same topics, they can convey the same idea even though they contain different terms. The chosen topics must be the same in each document, but they are not written in the document; they are hidden from the reader and expressed in the terms that have been written.

Latent semantic analysis is the process of discovering these hidden topics and their relationship to the term and document set.

2.3 Probabilistic Latent Semantic Analysis

Probabilistic latent semantic analysis (PLSA) [3] is the process of calculating the term, topic and document relationships using probabilistic means. In this section, we will explain the basic concepts behind the method.

Consider the document set as being a bag filled with tokens; one token for every occurrence of a term in the document set. Each token has an associated term and document label attached. We can say that $P(d, t)$ is the probability that we put our hand in the bag and take out a token with the document label

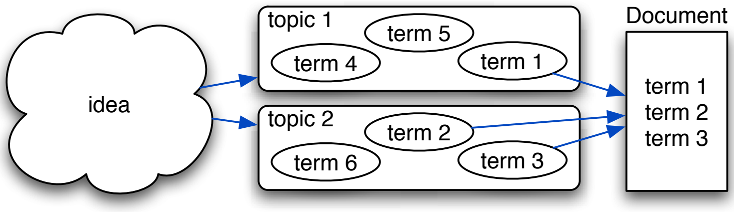

Fig. 2. The LSA document model. The author chooses specific topics for the document and then chooses a term from the topic to place in the document. This model implies that documents containing different terms can convey the same message, as long as the replacement terms are associated with the same topic.

d and term label t associated with it. Therefore if $f_{d,t}$ tokens are in the bag with labels d and t, implying that term t appeared $f_{d,t}$ times in document d, we obtain the sample probability:

$$\hat{P}(d,t) = \frac{f_{d,t}}{\sum_{\delta \in D} \sum_{\tau \in T} f_{\delta,\tau}} \tag{2}$$

where D and T are the set of document and terms respectively and $\hat{P}(d,t)$ is the sample probability of document d and term t. PLSA attempts to model these sampled document-term probabilities as the sum of hidden topic distributions:

$$P(d,t) = \sum_{z \in Z} P(d|z)P(t|z)P(z) \tag{3}$$

where Z is the set of hidden topics, $P(d,t)$ is the probability of term t being related to document d, $P(d|z)$ is the probability of document d given topic z, $P(t|z)$ is the probability of term t given topic z, and $P(z)$ is the probability of topic z. Using this model, we must fit our $|D| \times |T|$ samples using $|D| \times |Z| + |T| \times |Z| + |Z|$ parameters, where $|Z|$ is much smaller than $|D|$ and $|T|$.

3 Removing Bias in PLSA

Many weighting schemes have been developed for document retrieval systems to remove the bias found in non-homogeneous document collections [1, 4, 10]. Factors such as document length and term rarity can lead to the favour of certain irrelevant documents if not normalised.

We would expect that these biases also exist when calculating term relationships. We have seen that our samples $\hat{P}(d,t)$ are crucial in the calculation of the unknown probabilities based on z. This leaves us with the question, what do we base our document-term sample probabilities on? We hypothesise that the biases found within raw term frequencies also affect the calculation of term relationships performed during PLSA.

In this section, we will examine the popular BM25 weighting scheme and how we can apply it to PLSA.

3.1 BM25 Weighting Scheme

The BM25 weighting scheme [4] has a probabilistic background based on the modeling of relevant and irrelevant documents using Poisson distributions [9]. It has been developed for use in relevance feedback systems, but when simplified to use no document relevance information, it is still very competitive [12].

The simplified (no relevance feedback) document scoring equation can be shown as:

$$s(d, Q) = \sum_{t \in Q} w_{d,t} w_t \tag{4}$$

where d is the document to be scored, Q is the set of query terms, $w_{d,t}$ and w_t are the document-term and term weights respectively.

The term weight is calculated as either the log odds of the term appearing in a document:

$$w_t = \log \left(\frac{N - f_t + 0.5}{f_t + 0.5} \right) \tag{5}$$

or the negative log of the probability of the term appearing in a document:

$$w_{t+} = \log \left(\frac{N}{f_t} \right) \tag{6}$$

where N is the number of documents and f_t is the number of documents containing term t. The term weight is used to reflect the importance of the term due to its rarity. For example a term that appears in all documents is not useful as a query term, since it will return all documents, therefore its weight is low. A term that appears in one document is very useful as a query term, therefore its weight should be high.

The document-term weight is the function:

$$w_{d,t} = \frac{(k_1 + 1) f_{d,t}}{K + f_{d,t}} \tag{7}$$

where $f_{d,t}$ is the frequency of term t in document d, k_1 is a positive constant, and K is the pivoted document normalisation value. This function has two purposes. The first is to reduce the effect of large $f_{d,t}$ values. When searching for documents, one that contains twenty occurrences of a query term is not twice as relevant as one that contains ten occurrences of the same query term. In fact, they would both be considered just as relevant as each other. This function achieves this by reducing the increase in weight due to an increase in the term frequency. The second is to normalise the weight due to document length. A document that contains the query terms once in ten pages is not as relevant as one that contains the query terms in one page. The K value achieves this by normalising the documents based on their length.

3.2 Applying the Weights

Probabilistic latent semantic analysis calculates the maximum likelihood fit of the raw term frequencies (shown in equation 2). We want to perform a maximum likelihood fit of the term frequencies with biases removed, therefore we will perform PLSA on weighted term frequencies rather than raw term frequencies.

To use the weighted term frequencies, we simply substitute the weighted value where raw term frequencies are found. Therefore our new PLSA relationship becomes:

$$\hat{P}(d,t) = \frac{\omega_{d,t}}{\sum_{\delta \in D} \sum_{\tau \in T} \omega_{\delta,\tau}} \tag{8}$$

where $\omega_{d,t}$ is the weighted term frequency $f_{d,t}$.

PLSA uses the weighted term frequencies to construct a probabilistic model of the document set, therefore it is a requirement that the weight associated with each term in a document is positive. If we examine equation 5, we find that the log function returns negative values when applied to values less than one, which would occur when term t appears in over half of the documents. This property makes w_t unsuitable for use as an estimate of $P(d,t)$. The term weighting in equation 6 and the document-term weighting in equation 7 can never be negative, which make these weighting equations more suitable for our needs. Therefore we have the choice of using either of $\omega_{d,t} = w_{d,t}$, $\omega_{d,t} = w_t$ or $\omega_{d,t} = w_{d,t}w_t$. Once the weights are applied to every frequency value, we use the PLSA method to obtain the value of $P(d,t)$ and each of its components.

4 Experiments

We wish to analyse the effect of using weighted terms during the calculation of the PLSA term relationships. In this section, we describe the experiments performed and examine the data they produce.

We assumed that an increase in document retrieval precision implies that the term expansion is producing better terms for the query. Hence the probabilistic latent semantic analysis has established better relationships between the terms. Therefore, we will measure the effectiveness of the term relationships by examining the quality of the documents retrieved from a set of queries.

To store the weighted PLSA values, we will use a probabilistic latent semantic thesaurus (PLST), rather than a probabilistic latent semantic index (PLSI). The PLST has shown to provide greater precision, faster query times, and smaller storage space than the PLSI [8]. The PLST stores the probabilities $P(t_x|t_y)$ based on the computed $P(d|z)$, $P(z)$ and $P(t|z)$. The $P(t_x|t_y)$ values are used as a query expansion.

Our experimental environment was an information retrieval system consisting of a document-term frequency index using the BM25 weighting scheme and a probabilistic latent semantic thesaurus. Our experiments will examine the effect that weighting has on the the PLSA term relationships by performing one set of experiments with raw term frequencies ($\omega_{d,t} = f_{d,t}$) to calculate $P(t_x|t_y)$ and another set of experiments using BM25 weights ($\omega_{d,t} = w_{d,t}$, $\omega_{d,t} = w_{t+}$, and $\omega_{d,t} = w_{d,t}w_{t+}$) to calculate $P(t_x|t_y)$.

Previous work [6, 8] lead us to select the following constants for each latent semantic thesaurus: Only terms that appeared in more than 50 documents were

Table 1. Statistics of the four document sets used in the weighted latent semantic thesaurus experiments

document set	ZIFF1	ZIFF2	AP1	AP2
documents	75180	56920	84678	79919
median document length	181	167	353	346
avg. document length	412	394	375	370
unique terms	98206	82276	101708	95666
terms in 50 documents	7930	6781	10937	10498

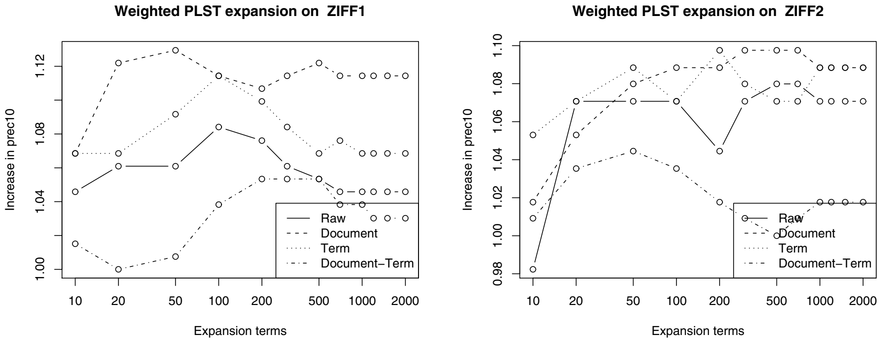

Fig. 3. A comparison of the increase in precision at 10 documents due to query expansion for PLST using $f_{d,t}$ (Raw), $w_{d,t}$ (Document), w_{t+} (Term), and $w_{d,t}w_{t+}$ (Document-Term) weights on the ZIFF1 and ZIFF2 document sets. The baseline BM25 precision at 10 documents with no expansion is 0.1985 and 0.1527 respectively.

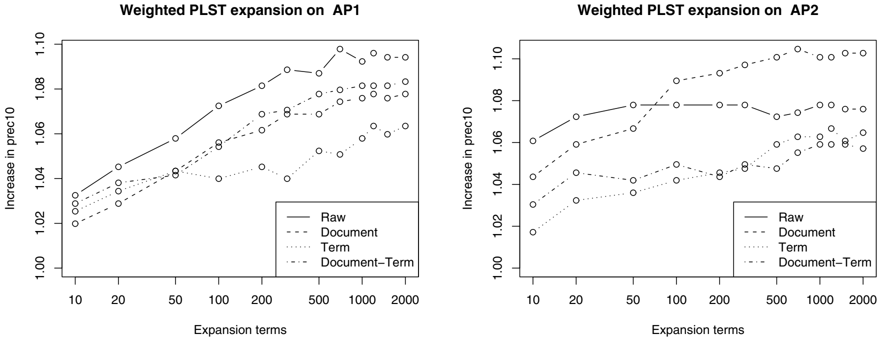

Fig. 4. A comparison of the increase in precision at 10 documents due to query expansion for PLST using $f_{d,t}$ (Raw), $w_{d,t}$ (Document), w_{t+} (Term), and $w_{d,t}w_{t+}$ (Document-Term) weights on the AP1 and AP2 document set. The baseline BM25 precision at 10 documents with no expansion is 0.3781 and 0.3554 respectively.

included in the thesaurus; the expansion terms were mixed with the query terms at a ratio of 0.6 to 0.4 respectively. Experiments were run on four separate document sets from TREC disks 1 and 2 named ZIFF1, ZIFF2, AP1 and AP2 (shown in table 1). On each document set, query expansions were performed using expansion sizes 10, 20 50, 100, 200, 300, 500, 700, 1000, 1200, 1500 and 2000. The expansion of size zero (implying no query expansion is performed) was used as a baseline to examine the precision of the retrieval system without using the probabilistic latent semantic term relationships. Therefore the results are presented in terms of increase in precision relative to this baseline. By setting the term expansion size to zero, we are switching off the PLST and thus our system becomes a BM25 document-term frequency index.

The experimental results showing precision at 10 documents are shown in figures 3 and 4. We can see in these plots that the $w_{d,t}$ weighted PLSA term relationships provide higher precision for most of our query expansion sizes for the ZIFF1 and ZIFF2 document sets. For the AP2 document set, PLSA using the raw term frequencies ($f_{d,t}$) provides higher precision than the $w_{d,t}$ weighted PLSA term expansion for 10, 20 and 50 terms. For all other expansion sizes the BM25 weighted PLSA expansion provides higher precision. It is interesting to note that the PLSA using the raw term frequencies ($f_{d,t}$) is generally flat for the three mentioned document sets. For the AP1 document set, we can see that PLSA using the raw term frequencies ($f_{d,t}$) provides higher precision for all levels of query expansion and is followed closely by the document weighted ($w_{d,t}$) expansion.

Significance testing using Wilcoxon's signed rank test was performed for three measures and is shown in table 2. The three measures used are mean average precision (MAP), precision after 10 documents (Prec10), and average reciprocal rank (ARR). MAP is used to judge the precision where many documents are required from the retrieval system, Prec10 is used to judge a system where a few documents are wanted, and ARR is used to judge a system where one document is wanted. The measures Prec10 and ARR are more useful for systems such as

Table 2. P-values from the Wilcoxon signed rank test. A P-value < 0.05 (marked with *) implies that using the associated weighting caused a significant increase in the associated measure. The measures shown are mean average precision (MAP), precision at 10 documents (Prec10) and average reciprocal rank (ARR).

Method	MAP	Prec10	ARR
$w_{d,t}$	0.618	0.011^*	$2.97 \times 10^{-06*}$
w_{t+}	0.995	0.989	0.227
$w_{d,t}w_{t+}$	1	1	0.9999182

Table 3. Storage sizes in megabytes for each of the thesauruses using different weighting schemes. We can see that there is a clear drop in storage size for each of the four document sets, when weights are applied during the thesaurus construction.

Weight	AP1	AP2	ZIFF1	ZIFF2
$f_{d,t}$	99.25	92.56	55.93	43.93
$w_{d,t}$	86.75	82.68	41.68	31.87
w_t	78.50	73.50	40.56	32.93
$w_{d,t}w_t$	82.31	76.68	34.06	27.37

Web search engines, where the user does not require specific documents, but only a few documents to satisfy their information need.

We can see from the P-values that there is no significant increase in either of MAP, Prec10 and ARR when using w_{t+} and $w_{d,t}w_{t+}$ values for the PLST. There is, however, a very significant increase in Prec10 and ARR when using $w_{d,t}$ weights for the PLST.

We have also provided the storage required for each of the probabilistic latent semantic thesauruses in table 3. It is interesting to see that the storage required for each of the weighted thesauruses was much less than that needed by the thesaurus using raw term frequencies ($f_{d,t}$). This is probably due to the weighted values having a smaller range and thus requiring less bits for each level in each range.

4.1 Comparison to BM25 Pseudo-Relevance Feedback

To obtain an understanding of how well our weighted PLSA query expansion method performs, we have provided a comparison to the results obtained when using BM25 pseudo-relevance feedback [4] and a BM25 co-occurrence thesaurus.

Relevance feedback, unlike our static thesaurus method, is the dynamic process of supplying the retrieval system with a set of documents relevant to the query. The retrieval system then extracts a set of terms from the relevant documents to use as a query expansion. Pseudo-relevance feedback, unlike relevance feedback, does not obtain any relevance information from the user; it chooses the top ranking documents to the query as the set of pseudo-relevant documents. This set of documents is then used to obtain the term expansion. Pseudo-relevance feedback using BM25 weighting has been a very successful query expansion method at TREC, therefore it is a useful benchmark.

A co-occurrence thesaurus is a table of term to term relationships obtained by calculating:

$$P(t_x|t_y) = \frac{\sum_{d \in D} f_{d,t_x} f_{d,t_y}}{\sum_{t_z \in T} \sum_{d \in D} f_{d,t_z} f_{d,t_y}} \tag{9}$$

The co-occurrence thesaurus is used just as the PLST is used.

Previous experiments comparing PLSA using raw term frequencies to pseudo-relevance feedback and co-occurrence thesaurus using BM25 weights showed that PLSA provided significant increases in ARR and Prec10, but pseudo-relevance feedback provided greater MAP. We have shown that the BM25 weighted PLSA provides significant increases in ARR and Prec10 over PLSA, but there is no significant increase in MAP. Therefore we will observe the difference in ARR and Prec10 between BM25 weighted PLSA, pseudo-relevance feedback and the co-occurrence thesaurus. The prior results suggest the pseudo-relevance feedback will produce the greatest MAP.

We have produced plots in figure 5, comparing each of the mentioned method for various levels of query expansion.

We can see from these plots that the PLSA query expansion using BM25 document weights is far superior in terms of average reciprocal rank, achieving an average 8% increase. We can see that our PLST method using document

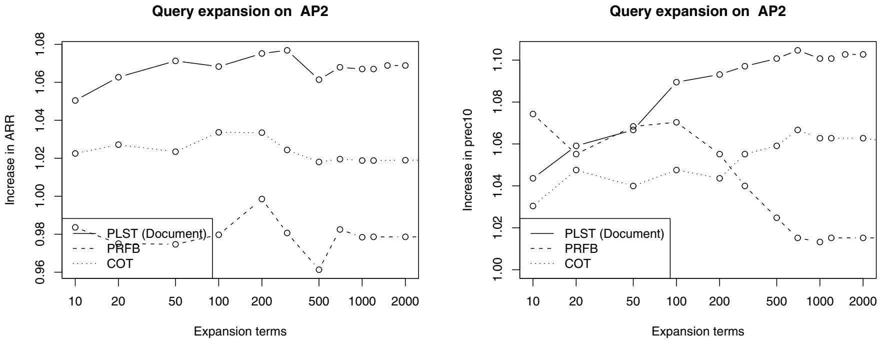

Fig. 5. Comparison of term expansion results on the AP2 document set using the average reciprocal rank (ARR) and precision at 10 documents (Prec10) measures. PLST (Document) is our probabilistic latent semantic thesaurus using document weights ($w_{d,t}$), PRFB is pseudo-relevance feedback, and COT is a co-occurrence thesaurus expansion method.

weights obtains a higher precision after 10 documents if 100 or more terms are chosen. Unfortunately, the relevance-feedback produces a greater mean average precision when using only a few terms. These results are similar to those of the PLST using unweighted term frequencies [8].

From these results we can see that our system would benefit a user who is searching for a few relevant documents, due to its high average reciprocal rank values. An example of this type of use would be found in typical Web searching. The pseudo-relevance feedback method would be more beneficial to a user who would want many or all relevant documents.

5 Conclusion

This article contains an analysis of the effect of using weighted terms during the probabilistic latent semantic analysis calculations and the impact it provides on probabilistic latent semantic term relationships.

We hypothesised that the term relationships obtained using PLSA will be more effective if calculated using weighted term frequencies rather than raw term frequencies. Raw term frequencies contain many forms of bias; weighted term frequencies are used to remove this bias during the query process, therefore weighted term frequencies should also be using when calculating probabilistic latent semantic term relationships.

Our hypothesis was tested by running precision experiments on a collection of document sets. We compared the precision from using a probabilistic latent semantic thesaurus built using raw term frequencies and a probabilistic latent semantic thesaurus built from weighted term frequencies. We found that using the thesaurus built from document weighted term frequencies provided a significant increase in precision at 10 document and average reciprocal rank. These results suggest that term relationships obtained using PLSA will be more effective when based on document weighted term frequencies rather than raw term frequencies.

We also compared the results obtained from the PLSA weighted thesaurus to those obtained using the BM25 pseudo-relevance feedback system. This analysis showed that the PLSA weighted thesaurus provided an average 8% increase in reciprocal rank and an increasing significance in precision after 10 documents, as the size of the term expansion increased. This implies that a PLSA weighted thesaurus retrieval system would be more useful than the BM25 pseudo-relevance feedback when found in an environment where a few document are required, such as a typical Web search.

References

1. Buckley, C., Walz, J.: SMART in TREC 8. In: Voorhees, Harman (eds.) [11], pp. 577–582
2. Dumais, S.T.: Improving the retrieval of information from external sources. Behaviour Research Methods, Instruments & Computers 23(2), 229–236 (1991)
3. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 50–57. ACM Press, New York (1999)
4. Sparck Jones, K., Walker, S., Robertson, S.E.: A probabilistic model of information retrieval: development and comparative experiments, part 2. Information Processing and Management 36(6), 809–840 (2000)
5. Park, L.A.F., Ramamohanarao, K.: Hybrid pre-query term expansion using latent semantic analysis. In: The Fourth IEEE International Conference on Data Mining, November 2004, pp. 178–185. IEEE Computer Society, Los Alamitos (2004)
6. Park, L.A.F., Ramamohanarao, K.: Query expansion using a collection dependent probabilistic latent semantic thesaurus. In: Zhou, Z.-H., Li, H., Yang, Q. (eds.) PAKDD 2007. LNCS (LNAI), vol. 4426, pp. 224–235. Springer, Heidelberg (2007)
7. Park, L.A.F., Ramamohanarao, K.: An analysis of latent semantic indexing term self preservation. ACM Transactions on Information Systems (to appear, 2008)
8. Park, L.A.F., Ramamohanarao, K.: Efficient storage and retrieval of probabilistic latent semantic information for information retrieval. The International Journal on Very Large Data Bases (to appear, 2008)
9. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: Proceedings of the 17th International Conference on Research and Development in Information Retrieval, London, pp. 232–241. Association of Computing Machinary, Inc., Springer, Heidelberg (1994)
10. Robertson, S.E., Walker, S.: Okapi/keenbow at TREC-8. In: Voorhees, Harman (eds.) [11], pp. 151–162
11. Voorhees, E.M., Harman, D.K. (eds.): The Eighth Text REtrieval Conference (TREC-8), Gaithersburg, Md. 20899, National Institute of Standards and Technology Special Publication 500-246, Department of Commerce, National Institute of Standards and Technology (November 1999)
12. Voorhees, E.M., Harman, D.K.: Overview of the eighth text retrieval conference (TREC-8). In: The Eighth Text REtrieval Conference (TREC-8) [11], pp. 1–23