to appear in the *Computational Intelligence* special PACLING issue
based on a PACLING 2001 Best Paper Award winner
re-revised December 14, 2006

Searching for Explanatory Web Pages Using Automatic Query Expansion

Manabu Tauchi **Nigel Ward**
School of Engineering, University of Tokyo*

Nigel Ward
Computer Science
University of Texas at El Paso
El Paso, Texas, USA 79968-0518
+1-915-747-6827
fax: +1-915-747-5030
nigel@cs.utep.edu

Manabu Tauchi
manabu1023@hotmail.co.jp

Note: We thank the reviewers for their corrections to our revised submission.

Abstract: When one tries to use the Web as a dictionary or encyclopedia, entering some single term into a search engine, the highly-ranked pages in the result can include irrelevant or useless sites. The problem is that single-term queries, if taken literally, underspecify the type of page the user wants. For such problems automatic query expansion, also known as pseudo-feedback, is often effective. In this method the top n documents returned by an initial retrieval are used to provide terms for a second retrieval. This paper contributes, first, new normalization techniques for query expansion, and second, a new way of computing the similarity between an expanded query and a document, the "local relevance density" metric, which complements the standard vector product metric. Both of these techniques are shown to be useful for single-term queries, in Japanese, in experiments done over the World Wide Web in early 2001.

Keywords: search engines, reranking, pseudo-feedback, local relevance density, terminology, Japanese

Acknowledgements: We thank Minoru Terada, Kazutaka Maruyama, and an anonymous referee for support and advice.

*Tauchi is currently with Goldman Sachs and Ward is with the University of Texas at El Paso.

1 Introduction

The World Wide Web contains massive amounts of information of every kind, constantly updated. We can access this information at home or from the office, for any purpose, and have come to rely on it more and more, in place of traditional information sources. For example, when we want to investigate the meaning of a term, we frequently search for pages which describe or explain that term on the Web, instead of looking it up in a dictionary or encyclopedia.

In 2001 this was difficult[†]. Often the top-ranked pages obtained from search engines did not include any with useful explanations of the query term. Today, in 2006, the situation has improved: it is much easier to find explanatory web pages, perhaps in part due to techniques like those introduced here.

The rest of this paper is structured as follows. Section 2 explains the basic idea of automatic query expansion. Section 3 presents our refinements to the way term weights are computed. Section 4 introduces a "local relevance density" metric for estimating whether a page contains explanatory material about a term. Section 5 describes the implementation. Section 6 describes the evaluation method and Section 7 reports the results, showing that automatic query expansion is valuable in itself, and that our two improvements give further improvements in search quality. Finally Section 8 points out some open questions, Section 9 discusses the significance of these results, and Section 10 summarizes.

2 The Vector Space Model and Automatic Query Expansion

Fundamental to web search engines is the vector space model. In this model all documents and queries are represented as vectors, with one component in each vector for every distinct term (word) that occurs in the document collection. Typically document vectors are weighted to enhance the effect of distinctive terms, that is, to give more importance to terms common in the document but rare in the collection as a whole.

In this model similarity is estimated by the inner product of the document vector and the query vector. Thus a document is similar to a query to the extent that the query terms also appear, frequently and distinctively, in the document. The search engine then presents to the user a list of links to the most similar documents.

The vector space model presupposes that a query contains sufficient information to meaningfully compute the similarity between the query and each document. For single-term queries this is often not the case. This means that the user may be presented with pages which are mostly only of marginal relevance. The user then has to scan many pages or iteratively refine the query, adding terms to narrow down the retrieval result to the sort of pages he wants.

[†]The original version of this paper [1] appeared in 2001.

The basic idea behind automatic query expansion is that this process of query refinement does not necessarily require human input. Specifically, since the top ranked documents of the intial query are likely to include terms relevant to the initial query, those terms can be harvested and used to create a new query. This method is also called "pseudo-feedback" in contrast to the use of real, user-provided feedback on page relevance, and also sometimes "automatic local analysis". This insight dates back to the early years of information retrieval [2, 3]. Automatic query expansion has shown good results on some, but not all, information retrieval problems [2, 4].

For the Web, the value of automatic query expansion has been explored primarily in the context of the Web Track of the Text Retrieval Competitions (TREC) from 1999 to 2003 [5]. However the conditions of these experiments did not match typical real-world Web search in a number of ways. One difference was that the queries were quite long; even though some work has studied so-called "short" queries, these still averaged over 2 terms [6]. Another difference was that the pages in the collection were all taken from the .gov domain and thus were all of generally high quality. A third difference is that the search tasks were not representative of the real tasks of users on the web; and in particular there was no evaluation with tasks resembling the search for explanatory web pages. (The "topic distillation" task, although similar, involved searching for a collection of web pages useful as background information for a report, rather than just finding information to satisfy one's own curiosity about the meaning of a term.)

3 Creating the Expanded Query Vector

Although previous work casts no light on the specific focus of this paper, the search for explanatory web pages, we did expect automatic query expansion to work well for this task. Clearly a page with an explanation will contain not just the term of interest but also related terms, which is exactly what an expanded query specifies.

We also expected automated query expansion to solve two specific common problems. The first was occasional high rankings for pages which only incidentally mentioned the query term, but which were highly rated for other reasons. The second was occasional high ratings for pages which were a pinpoint match to the initial query, according to the vector space model, but which had little content. An example was one highly ranked page for the query "IT Revolution" which contained only the text "Howdy. The IT Revolution; it's sweeping the world. So I made a homepage. Bye."

We implemented our own version of query expansion, quite straightforwardly, with only three small refinements.

First, we did not want longer pages to contribute more to the expanded query vector just because they happened to be longer. Thus we do not treat all the retrieved documents as providing one huge bag of terms. Rather we first compute the document vector for each document, then we normalize each of these vectors to have length 1, and only then do we average them together to get the expanded query vector.

Second, we did not wish generally common words to affect the expanded query vector, reasoning that they are just noise. We are only interested in terms which are

Term	$R_{\text{foot-binding}}$
纏足 (foot binding)	9.8
お婆さん (old woman)	7.3
奇習 (strange custom)	6.8
幼女 (infant girl)	6.4
宋 (Song dynasty)	6.3
宦官 (eunuch)	6.2
足首(ankle)	6.1
…	
今 (now)	0.5
ハーブ (herb)	0.5

Figure 1: The values for R_n obtained using as initial query the Japanese word for "foot binding"

characteristic of the set of documents in the initial retrieval. There are various ways to formalize this notion [7]; we chose a simple one, defining R_n, the "relevance" of a term to the search term, to be

$$R_n = \max(\log \frac{D_n}{G_n}, 0)$$

where D_n is the Domain probability, estimated by the average probability of term n in the initial retrieval set, and G_n is the General probability, that is, the probability of term n in the Web as a whole, estimated from a fairly random sample of web pages, with smoothing. (If D_n is zero then R_n is defined to be 0.) The reason for taking the logarithm of the probability ratio is to prevent a few highly frequent terms from excessively dominating the result.

Third, we did not want words which occur only in pages from one site (such as company names) to appear in the expanded query: such terms are removed (given 0 weight).

Figure 3 illustrates by showing some terms highly relevant to an initial query on foot binding.

4 The Local Relevance Density

Some pages contain words relating to a term but not actually an explanation of that term. This is sometimes seen in long pages and pages which are on a topic related to the

纏足は中国の宋の時代から広まった風習です。

9.8 3.1 6.3 1.8 5.0

(Foot binding is a Chinese custom from the age of the Song Dynasty)

9.8 3.1 5.0 1.8 6.3

Figure 2: Example of a sentence with high co-occurrence relevance to a search on "foot binding"

うちのハーブが、今、纏足 状態です

0.8 0.5 0.5 9.8 0.5

(The herb in my flower pot is in a state like foot binding now.)

0.5 0.8 0.5 9.8 0.5

Figure 3: Example of a sentence with low co-occurrence relevance to a search on "foot binding"

expanded query but not about the queried concept itself, as in news reports. To exclude such pages, we developed a notion of "local relevance density": this is a measure of the extent to which a page contains a specific region (passage) in which query-related terms occur densely.

The first step in the calculation is computing the "co-occurrence relevance" CR_k of each sentence. This measures the extent to which a sentence has a number of terms from the expanded query which are occurring together.

$$CR_k = \frac{\sum_i \sum_{j,i \neq j} R_i \cdot R_j}{{N_k}^2}$$

where R_i is the relevance of term i as computed above, and N_k is the number of terms in sentence k. If a term occurs two or more times in a sentence then it is counted only once.

Continuing the foot binding example, Figures 2 and 3 illustrate the computation of the co-occurrence relevance for two sentences from two pages.

The second step is to look for pages containing a clump of sentences which individually have high co-occurrence relevance. Figure 4 illustrates how the co-occurrence relevance might vary across the sentences of a page. In this example, the page is crowded with relevant sentences near the middle. Such a region is likely to be a rich source of explanatory information, and a page containing any such region should be highly rated. Specifically, the Local Relevance Density of a page is defined as:

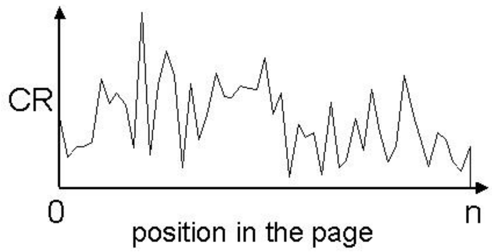

Figure 4: Example of how the Co-occurrence Relevance might vary within a page

$$LD_i = \max_{1 \le k \le n_i} \sum_x CR_x \times \max(10 - |x - k|, 0)$$

where n_i is the number of sentences in page i. Thus we use a moving filter with the shape shown in Figure 5, which implies that a page with potentially explanatory sentences is scored higher to the extent that the sentences are closer together. The value 10 was chosen because empirically a 10 sentence half-width was found to give good results. The maximum value across the page is used as the value for that page, based on the reasoning an explanatory clump appearing anywhere on the page makes that page valuable.

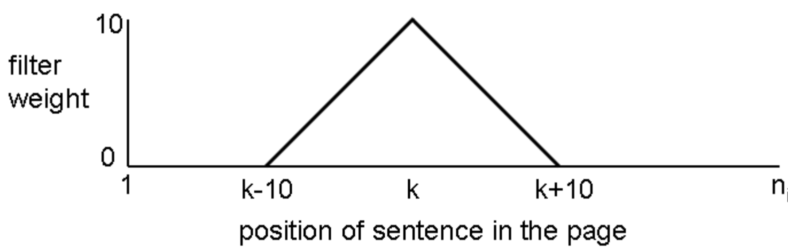

Figure 5: The Local Relevance Density filter, for detecting clumped explanatory (high CR) sentences.

5 Implementation

We developed a protoype search engine, dubbed Perrie, incorporating the above ideas. Figure 6 shows the process flow in Perrie.

To avoid the work associated with crawling and indexing the web, Perrie was implemented as a meta-search engine, parasitic upon Infoseek, which was probably the most popular search engines in Japan when this work was started, in 2000. Given a search

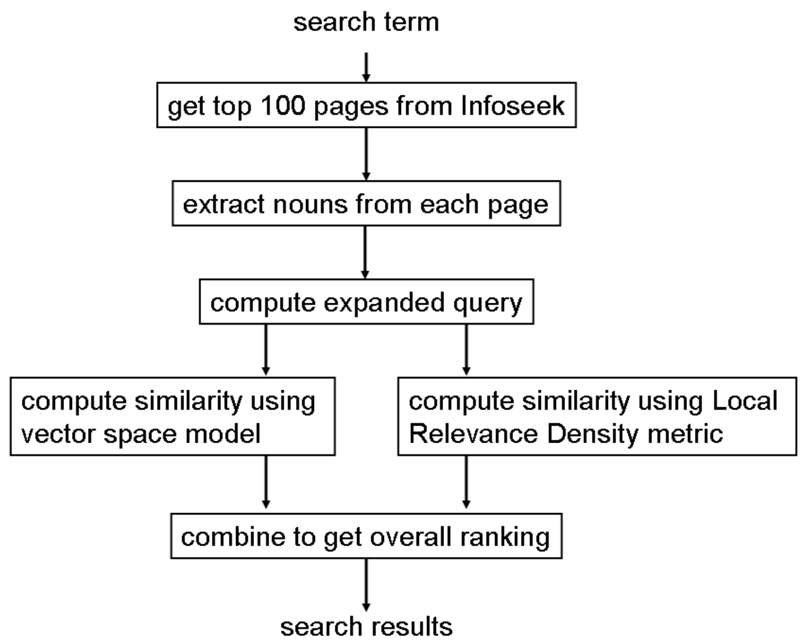

Figure 6: Overview of Processing in Perrie

term, Perrie used Infoseek to retrieve a starter set of web pages; the size of this set was 100, the largest number we felt comfortable processing. Infoseek's ranking of these pages was ignored. Perrie then invoked the morphological analyzer "Chasen" [8] on each page to obtain the nouns; only nouns were used as terms.

The expanded query vector was then derived as described in Section 3. Note that the initial query was only used for creating the expanded query vector, in contrast to methods where the second retrieval is done using the average of the initial query and the expanded query. Forgetting the initial query was appropriate because our second "retrieval" stage was only a re-ranking, operating only within the set of pages which, thanks to Infoseek, had some relation to the initial query term. Of course, in most cases the initial query term did appear prominently in the expanded query vector.

Since implemented as a meta-search engine, Perrie did this work, including the initial retrieval, on demand. However this is no reason that this could not be precomputed and stored, for faster response times.

The similarity of each of the 100 pages to the expanded query vector was then computed in two ways: using the standard inner product (Section 2) and using our Local Relevance Density Measure (Section 4). For purposes of computing the Local Relevance Density, each page was divided into sentences using HTML tag information [9] followed by morphological analysis using Chasen.

The overall "explanatory value" of each page was then computed as the product of the Local Relevance Density and the inner product score. This was done since we hypothesized that both of the similarity metrics were valuable, since the sort of page users probably want is both highly relevant and has a dense explanatory section.

The 100 pages were then ranked based on this explanatory value, and presented in order to the user.

6 Evaluation

6.1 Evaluation Method

To evaluate system performance we needed first of all a set of queries. While there are standard query testsets, there are none for explanatory web pages, and none for Japanese, so we chose to obtain our own. To do this we recruited subjects and gave them the task: "Please use the Internet in order to acquire the meaning or some information about a word. First think of some word about which you want to learn more." The terms they chose are shown in Figure 7. Of these terms 8 are historical, 5 medical, 4 relate to current events, and 2 are recent coinages.

Next we needed a measure of how well various pages matched the various queries in terms of meeting users' needs. This was obtained for each of the top 40 pages returned by Infoseek for each query, by having subjects rate each page on a scale from 1 (not at all related or not at all helpful) to 7 (very satisfying). Users were asked to base their scoring on the content of the page itself, not considering the value of pages linked to from that page.

Finally we needed a way to judge the quality of web search: specifically the ranking given by Perrie. Web search quality can be quantified in several ways [4]. Many evaluation metrics are based on the premise that all documents can be divided into two sets, those which are relevant and those which are not. However, this assumption fails for web pages, which are typically relevant or not to a degree. Moreover, unlike traditional information retrieval tasks, where users want all relevant documents, Web searchers typically are happy when they find any one page that has the information they want.

We therefore use a novel evaluation criterion. The basic idea is to consider a search engine's ranking better to the extent that it correlates with the user's ratings. Thus the Raw Ranking Accuracy is defined as

$$RRA = \sum_{i=1}^{20} \frac{\text{(user-given rating of } i^{th} \text{ ranked page)}}{\log(i+1)}$$

where the denominator is there to reduce the credit given for good pages which are given low ranks by the search engine. Some metrics for judging the quality of rankings use a document's ranking in the denominator, presumably reflecting the idea that the user's overhead in accessing a new document is substantial. In the search engine context, however, users are typically able to detect likely pages from the summaries, so significant

a. IT革命 (IT revolution)
b. 出師表 (Chinese historical document)
c. ADHD (attention deficit hyperactivity disorder)
d. 関羽 (Guan Yu, a historical figure)
e. 屈原 (Qu Yuan, a historical figure)
f. 直木三十五 (Naoki Sanjugo, author)
g. ユーゴスラビア (Yugoslavia)
h. マイライン (myline, long-distance provider system)
i. 行為障害 (conduct disorder)
j. オギノ式 (rhythm method)
k. 筋ジストロフィー (muscular dystrophy)
l. ,ニューディール政策 (New Deal)
m. ラマーズ法 (Lamaze method)
n. 纏足 (foot binding)
o. 金庫株 (share buyback)
p. ワルサーP38 (Walther P38, a make of pistol)
q. スウィングバイ (swing-by)
r. ハプスブルグ家 (House of Hapsburg)
s. 宮部みゆき (Miyuki Miyabe, author)

Figure 7: The Terms the Subjects Chose to Search For.

credit should still be given if a good page is ranked somewhat below number one. This is why the factor used here is the reciprocal of the log, not the reciprocal itself.

Only the top 20 ranked pages were used in this computation, based on the observation that most users will not examine more than about 20 hits.

The RRA is then normalized to obtain the Ranking Accuracy as follows,

$$RA = \frac{\text{RRA - Expected RRA}}{\text{Ideal RRA - Expected RRA}}$$

where the "Ideal RRA" is the value that would be obtained if the ranking was perfectly in accordance with the user's rating, and the "Expected RRA" is the expected value for a search engine that ranks the pages of the set at random. Thus a Ranking Accuracy of 0 indicates random performance and 1 indicates perfect performance.

6.2 Results

We had several hypotheses: first, that automatic query expansion was of value for the Web itself, second, that our specific little refinements added value, third, that our Local

Density Metric was useful, and fourth that the Local Density Metric was not redundant to the usual vector space similarity metric.

To test these hypotheses, we computed the Ranking Accuracy for each of the 19 queries, for each of four methods:

A. A baseline ranking of pages using the vector space model with simple query expansion. Compared to our methods, this baseline differed in that it included none of the three refinements presented in Section 3, and in that it did the second retrieval using a query vector obtained by averaging the initial query vector and the expanded query vector.

B. A ranking of pages using the vector space model and the expanded query computed as described in Section 3.

C. A ranking of pages using the local relevance density and the expanded query computed as described in Section 3.

D. A combination of B and C, namely Perrie.

The ranking accuracies obtained by Infoseek and methods A, B, C, and D are shown in Table 1, and the average performance for each is illustrated in Figure 8.

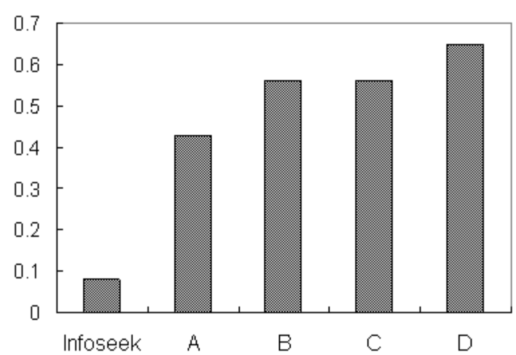

Figure 8: Average Ranking Accuracy

Comparing methods B, C, and D, which all use our extended query vector, no significant difference was seen between method B, using the vector space model and C, using local density. However with method D (Perrie), using both, the ranking accuracy is .09 points above methods B and C on average, and these differences are statistically significant (by the matched pairs t-test with 18 degrees of freedom, t=3.0 for BD and 2.4 for CD, $p < .02$ for both). This shows that these two reranking methods are not mutually redundant but rather complementary.

query	Ranking Accuracy				
	Infoseek	A	B	C	D
a	.39	.39	.41	.65	.52
b	.25	.24	.78	.59	.78
c	–.21	.48	.60	.73	.77
d	–.17	.66	.84	.85	.86
e	.09	.39	.30	.45	.48
f	.30	.48	.27	.76	.80
g	.17	.05	.26	.49	.44
h	–.36	.32	.37	.48	.56
i	.07	.23	.48	.39	.41
j	.25	.39	.60	.68	.57
k	.43	.64	.47	.37	.41
l	–.16	.22	.44	.49	.63
m	.05	.50	.67	.77	.78
n	.16	.80	.75	.11	.71
o	.19	–.10	.70	.61	.84
p	.10	.74	.81	.77	.89
q	.15	.38	.54	.45	.60
r	–.32	.60	.64	.35	.60
s	.15	.50	.62	.70	.74
average	.08	.43	.56	.56	.65

Table 1: Ranking Accuracy for Infoseek and 4 Query Expansion Methods

An average difference of 0.22 points was observed between the baseline method, A, and our combined method, D. differences for all pairings, except BC, were also statistically significant. Thus all hypotheses were supported.

Compared with Infoseek, search engine Perrie obtained a higher ranking accuracy for 18 of 19 words, showing an increase of 0.57 points on average. (Interestingly the two cases where Perrie gave little or no benefit were the cases where the initial simple query gave quite good results.) Moreover, at least one of the pages scored highest by the subject was found among top 3 ranked pages for 14 of the 19 words.

6.3 Discussion

Analysis of cases where valuable pages were given low scores by Perrie revealed four main causes. The first was bifurcated search results to the initial query. For example, the subject who searched on Yugoslavia wanted information about the political situation, but many of the pages returned by the initial retrieval were in fact about Yugoslavia the soccer team. The pages ranked highly by Perrie thus came from both clusters, resulting in a relatively low match to the ranking given by this subject.

The second cause was query drift, a known problem with automatic query expansion [4, 10]. For example, many of the pages which included the term "swing-by" mentioned it incidentally in the course of arguing that a prophecy of Nostradamus predicted that the Cassini spacecraft would crash into the earth. "Cassini" and "Nostradamus" were accordingly prominent in the expanded query, and thus Perrie's highly-ranked sites included some which were more about the prophecy than about swing-by.

The third cause was the effect of images on web pages on subjects' rankings, which of course Perrie did not model.

The fourth cause was one subject who rated an English language page highly; of course that page was matched poorly by the expanded query vector.

7 Open Questions

Our results represent merely one datapoint in a vast space of possible methods and experiments in automatic query expansion. One topic to explore is how general these findings are, given that these results were specific to the Web as it was in early 2001, specific to the Japanese language, specific to single-term search, and specific to search for explanatory pages. Another direction for future work is the examination of the various detailed choices used in our method: some were rather ad hoc and need to be reexamined both theoretically and empirically. Another question of interest is how our methods may perform in combination with more recent innovations in automatic query expansion [11, 3, 6, 12, 13, 14, 15].

Of course, the value of automatic query expansion itself, compared to other methods, is not well understood. For example, a partial alternative to query expansion for finding explanatory pages is the use of syntactic pattern matching to find pages containing definitions [16]. More generally, Hawking and Craswell have observed that "retrieval methods based entirely on document content can be substantially outperformed by others which make use of "web evidence", such as anchortext, link measures, and URL or site structure" [5]. It seems likely that content-based evaluation still has something to add, especially when the aim is finding pages containing meaningful information about a term, but the truth of this is an open question.

Finally, since users probably do not want to have to use a special search engine, or issue a special command, in order to find explanatory web pages, there is the question of whether automatic query expansion improves or hurts [10] the results for other types of queries, and, if the impact is negative, whether it is possible to reliably classify queries as seeking explanations or something else.

8 Prospects

In 2001, when we completed this work, we thought it was likely that automatic query expansion would be adopted in search engines in the near future. Now in 2006 we are in that future, but we still do not know whether this has been adopted: the techniques used in commercial search engines are trade secrets.

We can say that the task of finding explanatory search pages is much easier today. Feeding our 19 test queries and their English equivalents to Google revealed only two cases where a good explanation existed and was not at or easily accessible from the top search result. (The worst performance was for "swing-by", where the first explanation appeared at rank 12, preceded by a number of pages on specific space probes, a bland dictionary definition, a German language page, a spam page, and a page on the Java Swing toolkit. Incidentally, this failure cannot be taken as evidence that Google is not using automatic query expansion; it has often been noted that this technique is prone to degrade performance on some fraction of the queries [10], and indeed this particular query was problematic for our method also.) Thus there is still some room for improvement, but not all that much.

We can also note, however, that this good performance has been achieved at a substantial cost. Although part of the performance is due to search engine improvements, much is also due to improvements in the web. Obviously, there is more explanatory content available than there was 5 years ago. Equally important, the web is better organized. Wikipedia is the shining example, but many content providers and organizations have become better at organizing content in ways that help search engines perform better. This improvement in searchability comes at a huge cost in site design and "search engine optimization" efforts by webmasters. In a sense, this is a burden imposed on content providers and aggregators by the limitations of today's search engines. It may be naive to hope for a Web where a content provider can put something valuable almost anywhere, without much planning or negotiation, and the people who want that content will still be able to find it. Nevertheless, it is premature to be satisfied with the current state of web search. The techniques presented here may help.

Automatic query expansion and our Local Relevance Density measure may also have applications for other information retrieval tasks.

9 Summary

This paper presented experiments with Perrie, a prototype 'term search engine', designed for the purpose of using the Web as an encyclopedia. The results showed that automatic query expansion works well even for single-term search, that automatic query expansion works well on the real Web, that the detailed weightings of the terms in the expanded query are important, and that using the local density of relevant sentence as a measure of explanatory-ness is effective.

This work demonstrated the feasibility, to some extent, of automatically detecting documents which contain explanations, and of doing so based on document content only.

References

[1] Tauchi, Manabu and Nigel Ward. Searching for Explanatory Web Pages using Query Expansion. in *Pacific Association for Computational Linguistics (PACLING)*, pp 269–276. 2001.

[2] Baeza-Yates, Ricardo, and Berthier Ribeiro-Neto. Query Operations, Chapter 5 of *Modern Information Retrieval*, ACM Press, 1999.

[3] Carpineto, Claudio, Renato de Mori, Giovanni Romano and Brigitte Bigi. An Information-Theoretic Approach to Automatic Query Expansion. *ACM Transactions on Information Systems*, 19, pp 1-27. 2001.

[4] Ruthven, Ian and Mounia Lalmas. A Survey on the use of Relevance Feedback for Information Access Systems. *Knowledge Engineering Review*, 18, pp 95-145, 2003.

[5] Hawking, David and Nick Craswell. Very Large Scale Retrieval and Web Search. in Ellen M. Voorhees and Donna K. Harman, eds., *TREC: Experiment and Evaluation in Information Retrieval*. MIT Press, 2005.

[6] Carpineto, Claudio, Giovanni Romano and Vittorio Giannini. Improving Retrieval Feedback with Multiple Term-Ranking Function Combination. *ACM Transactions on Information Systems*, 20, pp 259-290. 2002.

[7] Buckley, Chris, Amit Singhal, Mandar Mitra, and Gerard Salton. New Retrieval Approaches using SMART : TREC4. in *The Fourth Text REtrieval Conference(TREC-4)*, D. K. Harman (ed.), pp 25-48. 1996,

[8] Matsumoto, Yuji, Akira Kitauchi, Yoshitaka Hirano, Tatuo Yamashita, and Masayuki Asahara. Users Manual for Chasen version 2.0; A Morphological Analyzer for Japanese. JAIST-Nara Technical Report. 1999

[9] Kiyota, Yoji and Sadao Kurohashi. Gisting and KWIC-index Generation from Web Texts. Information Processing Society of Japan. Workshop Notes 2000-NL-137-5, 2000.

[10] Mitra, Mandar, and Amit Singhal and Chris Buckley. Improving Automatic Query Expansion. in *SIGIR '98*, pp 206-214. ACM, 1998.

[11] Improving pseudo-relevance feedback in web information retrieval using web page segmentation. Yu, Shipeng, Deng Cai, Ji-Rong Wen, and Wei-Ying Ma. in *Proceedings of the 12th International Conference on World Wide Web*. pp 11-18. ACM, 2003.

[12] Xu, Jinxi, and W. Bruce Croft. Improving the Effectiveness of Information Retrieval with Local Context Analysis. *ACM Transactions on Information Systems*, 18, pp 79-112. 2000.

[13] Kurland, Oren, Lillian Lee, and Carmel Domshlak. Better than the Real Thing? Iterative pseudo-query processing using cluster-based language models. in *SIGIR '05*. ACM, 2005.

[14] Tao, Tao and ChengXiang Zhai. A Two-stage Mixture Model for Pseudo Feedback. in *SIGIR '04*. pp 486-487. ACM 2004.

[15] Sahlgren, Magnus, Jussi Karlgren, Rickard Cöster, and Timo Järvinen. SICS at CLEF 2002: Automatic Query Expansion Using Random Indexing. In C. Peters, M. Braschler, J. Gonzalo and M. Kluck (eds.), *Advances in Cross-Language Information Retrieval, Third Workshop of the Cross-Language Evaluation Forum, CLEF 2002*, pp.311-320, Springer 2003.

[16] Sakurai, Yuu, and Satoshi Satoh. Using the World Wide Web for Terminology Search. Information Process ing Society of Japan, Workshop Notes 2000-NL-137-4, 2000.