Term importance, Boolean conjunct training, negative terms, and foreign language retrieval: probabilistic algorithms at TREC-5

Fredric C. Gey, Aitao Chen, Jianzhang He, Liangjie Xu, and Jason Meggs
U.C. Data Archive & Technical Assistance (UC DATA)
University of California, Berkeley
e-mail: gey@ucdata.berkeley.edu

ABSTRACT

The Berkeley experiments for TREC-5 extend those of TREC-4 in numerous ways. For routing retrieval we experimented with the idea of term importance in three ways -- training on Boolean conjuncts of the most important terms, filtering with the most important terms, and, finally, logistic regression on presence or absence of those terms. For ad-hoc retrieval we retained the manual reformulations of the topics and experimented with negative query terms. The ad-hoc retrieval formula originally devised for TREC-2 has proven to be robust, and was used for the TREC-5 ad-hoc retrieval and for our Chinese and Spanish retrieval. Chinese retrieval was accomplished through development of a segmentation algorithm which was used to augment a Chinese dictionary. The manual query run BrklyCH2 achieved a spectacular 97.48 percent recall over the 19 queries evaluated before the conference.

1. Introduction

From the beginning of the TREC conference series, the UC Berkeley Text Retrieval Research Group has been developing probabilistic algorithms to retrieve full-text documents from collections as large as 1 million documents The Berkeley approach has been 'bare bones', concentrating on fundamental algorithm features rather than a 'kitchen sink' approach of adding ad-hoc features (like passage retrieval or phrase discovery, etc.) merely because they add a modicum to performance. Our core approach has been to use the statistical technique of *logistic regression* to predict relevance as a function of statistical attributes of query terms common to both document and query. Logistic regression is comparable to other probabilistic approaches such as neural networks (Kwok 1996), inference networks (Turtle and Croft 1991) and 2-Poisson (Robertson and Walker 1994).

2. TREC-5 routing methodology

The Berkeley TREC-4 routing results were puzzlingly low, since our basic training seemed to show that we should have achieved better results than TREC-3. After the TREC-4 conference we reran our algorithms from TREC-3 on the TREC-4 data and found that they performed better than TREC-4 run Brkly12. The basic feature of the TREC-3 algorithm was to:

- Perform a χ^2 analysis to find terms which had the highest statistical dependence on relevance
- Choose all terms (for query expansion) where the χ^2 had probability value greater than 5 percent significance level. This produced between 300 and 4114 terms depending upon the query, with a mean query size of 1,357.

For TREC-4 we had believed that we could truncate the number of expansion terms to the 300 most important ones (in terms of the size of χ^2). This judgment turned out to be faulty -- a retrospective run using all expanded terms improved performance by 20% over the official Brkly12 run. It seems that an incremental evidence contribution from a large number of insignificant terms can be more important than the contribution from a few important terms.

Thus, for TREC-5 we returned to the concept of massive query expansion using the χ^2 discrimination measure to choose terms. This, using the TREC-3 formula for combination of evidence from terms, forms the method for our first routing run, Brkly13. The number of terms for TREC-5 routing varied from a minimum of 714 (topic 222) to a maximum of 3839 (query 82), with an average of 2032 terms over the 50 queries.

2.1. Boolean filtering for important terms

Despite the evidence that massive query expansion seems to yield our best performance, we have continued to be intrigued by the OKAPI results which depend upon choosing the 15 "best" terms for expansion and ignore the rest (Robertson, Walker, et.al. 1995). We wondered whether some use of these "best" terms might strip away noise documents retrieved by other terms. What we wish is to utilize those terms which have the most effective discriminating power on the basis of their relevance history.

For example, if we consider the TREC-5 topic 003 (which was also a TREC-4 topic):

Domain: International Economics
Topic: Joint Ventures
Description: Document will announce a new joint venture involving a Japanese company.
Narrative: A relevant document will announce a new joint venture and will identify the partners (one of which must be Japanese) by name, as well as the name and activity of the new company.

The following ranks the query terms according to their term absence logodds of relevance:

Important terms for TREC query 003

Query	Term	Logodds
003	joint	-7.578775
003	ventur	-6.022744
003	japan	-5.706141
003	compan	-1.363744
003	produc	-0.790130
003	corp	-0.536089

From this table we conclude that the most important terms are *Japan, joint* and *venture.* The scale indicates that if the term *Japan* is absent from a document, the document has odds 8 times (3.73 times value of e) less likely to be relevant than if the terms *produc* or *corp* are absent from the document.

Retrieval performance by an official Berkeley TREC-4 Brkly12 entry for this query using probabilistic retrieval by a logistic regression equation (Gey et. al. 1995) produced average precision of 0.2852, below the median precision of 0.3765 for that query and less than half the best precision entry of 0.6781. However, if we restrict our ranking to only documents which contain the words Japan, joint, and venture, the performance increases to 0.6443, a substantial improvement and close to the best TREC-4 entry for this query. Further experiments, described in detail in (Gey, Chen 1996), showed that an intelligent choice of Boolean filters for all 50 queries of TREC-4, would have improved the overall precision of Brkly12 from 0.2163 to 0.3317, fifty-three percent higher than our official run.

2.2. Training on Boolean conjuncts

The use of the triples of Boolean conjuncts led us, in preparation for TREC-5, to the concept of training on Boolean "minterms" for the top terms chosen for query expansion in routing. Minterms are the elementary conjunct combinations of term presence or term absence when a Boolean query is transformed into its *disjunctive normal form.* The idea was to rank the minterms on their historical relevance density, which would give their order as to how each subset of documents might produce the greatest number of relevant documents in rank order. As we ran retrospective runs on the TREC-

4 disks2 and disk3 collections we found that precision improved the more minterms we used, until, using the top 15 terms (32,767 minterms) we would achieve a staggering precision of 0.8048.

Of course this was too good to be true -- we had fallen into the trap of applying our training set to our training collection rather than a new collection. When we applied the minterm rankings and coefficients to another collection (disk1) , performance always decreased, no matter how many or how few minterms we used. It seems that while the relevance history of terms found in relevant documents has considerable predictive power, the Boolean combination of terms does not.

Boolean minterms experiments average precision		
Number of terms	**retrospective**	**predictive**
5	0.2985	0.3194
10	0.4935	0.2405
15	0.8048	0.1086

We are pursuing further failure analysis of these results.

2.3. Logistic regression in routing

After having spent several months modifying our software in pursuing the blind alley of Boolean minterms, and with three weeks left before the results were due, the Berkeley group needed to find another approach. We continued to believe that some account needs to be made which incorporates the contribution from the most important terms. After some thought we settled upon an algorithm which trains on the 15 most important terms (in order of historical logodds of relevance when the term is absent from the document) and the total evidence from all terms chosen for query expansion (i.e. the RSV of our baseline run Brkly13). Thus the formula for the Brkly14 run is as follows:

$$\log O(R \mid q_{i,} d_j) = c_0 + c_1 \log(tf_1 + 1) + \cdots + c_{15} \log(tf_{15} + 1) + c_{16} weight + c_{17} \log DL$$

where tf_i means the term frequency of of the ith term in the document, *weight* is the logodds of relevance weights summed over all expanded query terms (the TREC-3 formula) found in the document, and finally DL_j is length of the document. This formula not only includes contributions from all terms, but adds an appropriate contribution for the 15 most important terms. Note that there are actually 50 regression equations, one for each query. This training-by-query was another improvement over TREC-4, where we produced one equation averaged over all 50 queries. Our training set was all relevant documents and a one-percent sample of non-relevant documents from the entire collection (not just the judgment set for that query). It is worth noting that the SPSS statistical package will solve all 50 equations in 15 minutes, as opposed to 15 minutes per equation using the S Plus statistical system.

2.4. TREC-5 routing performance

The Brkly14 run (average precision of 0.2601) performed 21 percent better than Brkly13 (average precision of 0.2156) in overall precision over the 45 queries, and 16 percent better over the 39 queries with more than two relevant documents. Brkly14 performed median or above in 35 of the 45 queries with 5 bests for their queries. If we compare TREC-5 to TREC-4 overall, it seems that the average median precision over all participating groups has dropped from .298 to .204, raising a question as to whether the target collection for TREC-5 was more difficult. This seems likely, since the distribution of relevant documents is highly skewed -- four queries (111, 142, 189, 202) account for 49.3 percent of all relevant documents for the 45 queries.

3. TREC-5 ad-hoc methodology

Berkeley did not spend much effort refining its approach to ad-hoc retrieval. The formula for ad-hoc has remained unchanged since TREC-2, when the concept of Optimized Relative Frequencies (ORFs) was introduced.

Berkeley's basic formula for prediction of logodds of relevance between a query Q and document D when there are M match terms in common between query and document, is

$$\log O(R \mid t_1, \ldots, t_M) \approx$$

$$-3.51 + \frac{1}{\sqrt{M}+1}\,\Phi + 0.0929\,M$$

where Φ is the expression

$$37.4 \sum_{j=1}^{M} \frac{Qtf_j}{QL+35} + 0.330 \sum_{j=1}^{M} \log \frac{Dtf_j}{DL+80} - 0.1937 \sum_{j=1}^{M} \log \frac{Ctf_j}{CL}$$

where

Qtf_j = number of times the j'th term occurs in the query and QL = total number of all term occurrences in query (query length);

Dtf_j = number of times the j'th term occurs in the document, and DL = total number of all term occurrences in document (document length);

Ctf_j = number of times the j'th term occurs in the collection, divided by CL, the total number of all term occurrences in the collection (collection length);

M = number of distinct terms common to both query and document.

This formula was used, without modification, in Berkeley's Chinese retrieval experiments discussed below.

Since Berkeley's TREC-4 performance was only average on the extremely short queries introduced in TREC-4, we speculated that the contribution from query terms:

$$\frac{Qtf_j}{QL+K_q}$$

might be overwhelmed by the constant K_q=35. Indeed, Kwok has noted (Kwok, 1996) that the average query size for the 49 queries of TREC-4 was about 6 terms, while the average query size for the 50 queries of TREC-3 was 19 unique terms.

In order to test this, Berkeley ran a number of retrospective tests for both short and long queries to see if they could be improved upon. We trained on the short description queries from topics 151-200 on TREC DISK2 and tested on the short queries from topics 201-250. The results, summarized in this table:

TREC-5 Tests on query constant **Average Precision for different values of K_q**		
query constant	Average precision	R-Precision
10	0.1864	0.2442
15	0.1951	0.2507
20	0.1955	0.2506
25	0.1933	0.2494
30	0.1880	0.2498
35 (base)	0.1961	0.2511
40	0.1770	0.2384

show that results are not particularly dependent upon constant value unless it exceeds 40 or falls below 15. Since the best performance comes with $K_q = 35$ we retained this value for our TREC-5 experiments.

3.1. A cautionary tale of ad-hoc query expansion

In TREC-4 our official runs used a technique of query expansion whereby the top documents of a 'trial retrieval' are assumed to be relevant, and some of their terms are added to the query. Following TREC-4, Berkeley did a more systematic series of runs connected with query expansion. These runs, summarized in the table below, show that Berkeley would have been better served by making its official entry the run without expansion, which at 0.2945 was nearly 10 percent better than the 10 document/ 20-term expansion of the actual entry.

TREC-4 Brkly10 Ad-hoc Entry Average Precision for document/term expansions				
0.2945*	10 terms	20 terms	50 terms	100 terms
10 docs	0.2821	**0.2660****	0.2335	0.1995
20 docs	0.2646	0.2533	0.2263	0.1857
30 docs	0.2589	0.2477	0.2138	0.1885

*** - No expansion, ** - Brkly10 official entry**

Moreover, the same is true of our TREC-4 automatic entry, Brkly9; the official entry using query expansion had a precision of 0.1388. Without query expansion, the average precision would have been 0.2001, more than 45 percent better than our official entry.

Berkeley abandoned this ad-hoc query expansion method for TREC-5, feeling that we have not yet mastered this art. Thus all TREC-5 ad-hoc runs use only the original (automatic or manually constructed) query terms, as do our Spanish and Chinese runs.

3.2. Manual query reformulation

Manual query reformulation has become the de-facto approach to enhancing retrieval for the ad-hoc problem. Berkeley has continued this tradition in TREC-5. Of course, the TREC committee decision to allow us to examine the top documents of a trial retrieval made this reformulation easier, although Berkeley used a combination of this and extra-curricular retrieval from a commercial news database in order to find additional terms.

3.3. Negative terms

During the course of TREC-4 Berkeley made a few cursory investigations into Boolean filtering with manually generated negative terms. A negative term, in a Boolean model, would be one which would exclude the document if the term appeared in it (i.e. the term would appear as AND NOT *term* in a Boolean query formulation). For TREC-4 we found negation to be entirely too rigid a condition.

Yet for TREC-5 there seemed to be queries for which some type of negation would be helpful. For example, retrieval using topic 292,

> *<num> Number: 292*
> *<title> Topic: Worldwide Welfare*
> *<desc> Description:*
> *Identify social programs for poor people in countries other than the U.S.*
> *<narr> Narrative:*
> *To be relevant a document would identify a welfare program in a foreign country and explain how it works to aid citizens who have little or no income. It would include those who can't work because of a disability and people who have the extra burden of small children. The document should indicate how these people are supported or not supported. A relevant document should identify the source of the monies used to support such welfare programs.*

retrieves many documents about the continuing controversy over welfare reform legislation in the United States.

Our speculation was that if "negative" terms such as 'Clinton', if they appeared in a document, the final weight would be reduced, the noise documents from the U.S. welfare system would be weeded out. Our final algorithm for this was to take the weight equations above and divide by the square root of the number of negative terms plus one. This seemed, prospectively, to work in that the first 10 documents of the Brkly17 run for this query were:

1. Welfare Overhaul Gets Final Touches In Senate
2. Senators Meet To Try To Find Welfare Plan Acceptable to Reagan
3. Surging Welfare Costs and the Struggle to Control Them
4. leading article: reforming usa welfare
5. Finance Committee Clears Bill To Overhaul Welfare System
6. An ecu for europe's poor: delors wants to put poverty higher on the agenda
7. Canada to unveil radical social security blueprint
8. Poor in the Country What The Presidential Candidates Propose
9. read clinton's lips: no more welfare: america
10. french senate approves state aid for companies cutting workers' hours

while the first 10 documents of the Brkly18 run were:

1. Canada to unveil radical social security blueprint
2. french senate approves state aid for companies clipping cutting
3. (no title: topic Reagan welfare reform)
4. benefits 'must target the poor'
5. New Study -- Most Homeless People Are Just Poor _ Not Mentally Ill
6. THE IMPORTANCE OF TWO-PARENT FAMILIES (House - April 28, 1994)
7. graduates 'happier on government unemployment benefits welfare dole than in stop-gap jobs'
8. one-day strike closes government unemployment benefits welfare offices
9. agency aids jobless graduates
10. Welfare Overhaul Gets Final Touches In Senate

Unfortunately, however, precision for Brkly18 was 0.0279, worse than Brkly17's precision of 0.0287 for topic 292. Overall precision over 47 queries was 0.2044 for Brkly18 and 0.2417 for Brkly17 runs without negative terms. This phenomenon held true for Spanish and Chinese as well. While we continue to believe in the viability of negative terms, how they should be combined remains elusive.

4. Berkeley ad-hoc results

In TREC-5 we had the luxury of submitting four runs, two automatic and two manual. In the following table we place the results of these four runs, together with the average of medians over all TREC-5 runs of that particular category;

TREC-5 Berkeley Ad-hoc Entries Average Precision for 47/50 queries				
	47 queries		50 queries	
Run number	Berkeley	All TREC-5	Berkeley	All TREC-5
Brkly15	0.1475	0.1529	0.1420	0.1437
Brkly16	0.2125	0.2026	0.2076	0.1905
Brkly17	0.2417	0.2314	0.2346	0.2174
Brkly18	0.2044	0.2314	0.1983	0.2174

Brkly15 - automatic run, short queries
Brkly16 - automatic run, long queries
Brkly17 - manual run, reformulated queries
Brkly18 - manual run, reformulated queries with negative terms

5. TREC-5 Chinese retrieval

Berkeley has the good fortune of having three team members who are native Chinese speakers. We were impressed with the performance of University of Central Florida (UCF) on Spanish retrieval at TREC-4, a performance attributed to the availablity of native Spanish speakers and students from Mexico who were intimately familiar with the socio-political landscape of the Mexico. The Spanish document collection of TREC-4 came from a Monterrey Mexico news service. Moreover, the UCF group spent an average of 40 hours per topic constructing an elaborate knowledge base for each query, in this way retrieving considerably more relevant documents than were retrieved by the other Spanish participants.

While the Berkeley group did not have the luxury of an extra 1,000 hours for query construction, the Chinese speakers and readers from our team spent about three hours per topic constructing manual queries for Chinese. This effort seems to have paid off with an overall precision of 0.4610 and 10 best-of-topic for our manual run BrklyCH2. BrklyCH2 only missed 35 relevant documents out of 1399 total relevant found for the 19 queries evaluated by the time of the conference. Since BrklyCH2 used negative terms, we also did an unofficial run without negative terms which found 8 more relevant documents and had an average precision of 0.4673.

5.1. Chinese segmentation software

It is well known that the Chinese character set consists of representations similar to phonemes in English, where words are usually composed of up to three adjacent characters. Unlike English and most European languages where word boundaries are distinguished by blank space, Chinese word formations must be distinguished by contextual recognition.

There are two main approaches to find wording boundaries in Chinese (He 1988, Chen and Lieu 1992, Wu and Tseng 1995). The first one uses a set of rules and a dictionary that includes components of one and two Chinese characters. The components in the dictionary are classified into different categories. After the components in the dictionary are recognized in Chinese context, the rules are applied to concatenate the components into a word. The second approach uses an exhaustive dictionary to match the longest string in the context. Since not all the words can be included in the dictionary the remaining strings are divided mechanically into short strings. On the other hand, Chinese has other features which lessen the work of preparing recognition software. In particular verb variants are non-existent, in that tense recognition is performed by the additional words "past" or "future" explicitly included within the text.

For TREC-5, the Berkeley group obtained a public domain Chinese dictionary of 91,000 words from the World Wide Web Chinese software site (http://www.ifcss.org/ftp-pub/software/data/). In addition a stop word list of 444 words was constructed manually. The Berkeley group then used its segmentation software to match substrings of Chinese character streams within the TREC-5 document collection against the initial dictionary. Character strings which did not automatically match dictionary words were output and examined manually, and those which were actual Chinese words were added to the dictionary. This process was iterated several times until an additional 46,659 words were added to the Chinese dictionary.

Our segmentation algorithm employed this basic strategy: starting with the first character in a document, the text was matched against a dictionary at each character in sequence. The text is searched from the first character, one character at a time. The longest match found from each starting character is kept, if its last character extends beyond the end of all previous matches.

Any non-matched space was considered unknown. This could be optionally output in a number of ways:

1) as single characters;
2) as a complete segment;
3) both;
4) alone, without the matches;
5) not output, suppressed.

For TREC-5 we used option 2 and wrote out the complete segment.

6. Chinese results

Our automatic runs, of course, utilized the same segmentation algorithms to construct the queries from the topic descriptions. The official TREC-5 automatic entry BrklyCH1 had overall precision (for, of course, 19 topics) of 0.3192, fourteen percent above the average of median precisions (0.2809) over the fifteen automatic runs. A striking difference between automatic and manual construction can be found in Chinese topic CH14 "Cases of AIDS in China" which uses the common (familiar) word for AIDS used in Hong Kong and Taiwan. This form, which roughly translates as "disease of love," is only found in five documents of the TREC-5 Chinese collection. In other documents the official term for AIDS (which is phonetically similar to the English pronounciation) is used. For the BrklyCH1 automatic run on this topic only 23 out of 57 relevant documents were retrieved (with a precision of 0.0715). For the BrklyCH2 manual run on topic 14 the addition of the official term for AIDS retrieved 46 out of 57 relevant for a precision of 0.4768. Overall the BrklyCH2 precision of 0.4610 was forty-four percent better than the BrklyCH1 automatic run's average precison of 0.3192.

7. TREC-5 Spanish retrieval

The main effort of TREC-5 Spanish work went into improving Berkeley's morphologic stemming software. A new, larger ortho-irregular verb list was obtained from D. German (German 1996), which was split, processed and analyzed. It was then refined to improve upon ambiguous definitions and added to the new Spanish stemmer. For the new Spanish collection, 184,469 verb instances were reduced to 3375 unique verb stems. This stemming seems to have improved our results (there was no change in the ad-hoc retrieval Spanish formula from TREC-4).

In addition, the TREC 5 Spanish queries and collection were modified in some new ways. As acronyms are generally short words that may lose their distinction through stemming and conversion to lower-case, a system of tagging acronyms was developed to try to preserve their uniqueness. This was used automatically on the collection and queries. Secondly, a system of correcting spelling mistakes, especially missing accent marks, was developed using a look-up table generated from a massive, unstemmed wordlist. This most likely made a significant difference in our automatic performance since the queries were full of spelling errors, especially missing accent marks, many of which our software caught. Finally, our automatic query was expanded such that terms estimated to be important words or names were repeated four times.

The BrklySP5 automatic run using the short descriptive Spanish query had an overall percision of 0.2526 with two runs (Spanish query 57 at precision 0.6726 and Spanish query 68 at 0.5778 precision) achieving best overall performance. The BrklySP6 run was our manual run with negative terms. Its overall precision of 0.3488 was thirty eight percent better than the automatic run.

As an example of the use of negative terms, Spanish topic SP58 on the narcodollar financing of Colombian President Ernesto Samper's election would retrieve documents about Samper visiting disaster areas after earthquakes and volcanic eruptions. For this query we added the negative terms 'avilanch' and 'seismo'. Since such a query also retrieves documents about narcodollars in Mexico and Brazil we added these words to the negative list. The result is that the precision for query SP58 decreased from 0.6421 for BrklySP5 to 0.2075 for BrklySP6. Once again, negative terms have yet to prove their viability.

Time and again it was clear that the addition of native Spanish speakers would have helped in manual query construction. We will be exploring a partnership with the UC Berkeley Center for Latin American Studies for future TREC conferences.

8. Summary

UC Berkeley's participation in TREC-5 led us to a number of different experiments for the routing problem, experiments which were informative but not always successful. Our final approach was to combine evidence from massive query expansion with a regression which weighted the fifteen most important terms of the expansion according to their predictive capacity. Our ad-hoc and foreign language retrieval experiments have proven the robustness of the TREC-2 algorithm which relies on "optimized relative frequencies" as clues. The Chinese experiments show that careful query construction is the fundamental cornerstone of excellent retrieval results.

9. Acknowledgments

Many of the central ideas were originally developed with Professor William Cooper, leader of the Berkeley TREC team for TREC-1 through TREC-3. We continue to use hacked over versions of the SMART system for our retrieval. We are again grateful to Daniel German for his Spanish morphological dictionary. A portion of this work was supported by grant NSF IRI-9630765 from the Database and Expert Systems program of the Computer and Information Science and Engineering Directorate of the National Science Foundation.

10. References

Chen K Liu S-H (1992) "Word Identification for Mandarin Chinese Sentences," *Proceedings of COLING-92, The 15th International Conference on Computational Linguistics,* Nantes, France, August 23-28,1992, pp 101-107.

German, D (1996) private communication. He can be reached at http://csgww.uwaterloo.ca/˜dmg/home.html.

Gey F Chen A (1996) "Intelligent Boolean Filtering for Routing Retrieval," UC DATA Technical Report IS96-1, January 1996, available from the authors.

Gey F Chen A He J Meggs J (1995) "Logistic Regression at TREC-4: Probabilistic Retrieval from Full-text Document Collections," in (Harman 1995b).

Harman D, ed. (1995a) *Proceedings of the Third NIST Text Retrieval Conference (TREC3),* National Institute for Standards and Technology, Washington, DC, November 2-4, 1994, NIST Special Publication 500-225, April 1995.

Harman D, ed. (1995b) *Proceedings of TREC-4, the Fourth Text REtrieval Conference,* National Institute of Standards and Technology, Gaithersburg, MD Nov 1-3, 1995.

Jianzhang He (1987). "The approach and experiments of the automatic word extraction in Chinese Science & Technology documents," Ching Pao Ko Hsueh (Information Science), v.8, no. 4, August 1987, pp 35-45. (in Chinese)

Kwok K (1995), "A network approach to probabilistic information systems," *ACM Transactions on Information Systems*, v. 13, no. 3, July 1995, pp 324-353.

Kwok K (1996) "A New Method of Weighting Query Terms for Ad-hoc Retrieval," *Proceedings of SIGIR96, the 19th Annual International Conference on Research and Development in Information Retrieval,* Zurich, Switzerland, August 18-22, 1996, pp. 187-195.

Robertson S Walker S Jones s Hancock-Beaulieu M and Gatford M, "Okapi at TREC-3," in (Harman 1995a).

Robertson S Walker S, "Some Simple Effective Approximations to the 2-Poisson Model for Probabilistic Weighted Retrieval," *Proceedings of SIGIR94, the 17th Annual International Conference on Research and Development in Information Retrieval,* Dublin, Ireland, July 3-6, 1994, pp. 232-241.

Turtle H and W Croft (1991). "Evaluation of an Inference-Network-Based Retrieval Model, *ACM Transactions on Information Systems,* v. 9, no. 3, July 1991, pp 187-222.

Wu Z Tseng G (1995) "ACTS: An Automatic Chinese Text Segmentation System for Full Text Retrieval," *Journal of the American Society for Information Science,* v.46, no. 2, January 1995, pp 83-96.