Proximity Within Paragraph: A Measure to Enhance Document Retrieval Performance

Srisupa Palakvangsa-Na-Ayudhya and John A. Keane
School of Informatics, The University of Manchester
Manchester, UK
s.palakvangsa-na-ayudhya@student.manchester.ac.uk,
john.keane@manchester.ac.uk

ABSTRACT

We created a proximity measure, called *Proximity Within Paragraph (PWP)*, which is based on the concept of value assignment to queried words, grouped by associated ideas within paragraphs. Based on the WT10G dataset, a test system comprising three test sets and fifty queries were constructed to evaluate the effectiveness of PWP by comparing it with the existing method: *Minimum Distance Between Queried Pairs*. A further experiment combines the scores obtained from both methods and the results suggest that the combination can significantly improve the effectiveness.

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

General Terms: Algorithms.

Keywords: Proximity Measure, Ranking Algorithm.

1. INTRODUCTION

Two widely-known research areas investigating document segmentation are *proximity measurement* and *passage retrieval* [2]. *Proximity measurement* considers the distance between queried words within the same document; on the other hand, *passage retrieval* can be viewed as a type of proximity measure, which mainly investigates how to segment a document into smaller units (termed *passages*) and returns only the most relevant passages to users rather than the whole document. Although the area of passage retrieval has been widely investigated, some issues need to be considered. For example, an appropriate indexing method has to be chosen, as it can be slow due to the larger number of passages compared to documents [8]. Moreover, it may not be suitable for very long queries as there is a low possibility that short passages can match many queried words [3].

2. PROXIMITY WITHIN PARAGRAPH

Based on the concept of article writing in which authors usually arrange associated ideas together, PWP considers the occurrence of unique queried words and determines relevance through the proximity of these words enclosed in the same paragraph (called *logical block* hereafter). The process of PWP can be divided into three steps as below.

1) Query Analysis: this procedure mainly concerns how to assign a weight to each unique queried word. In this experiment, an equal weight is employed; however, different weighting schemes can be applied instead, such as user-specified weights.

2) Logical Block Identification: as previous research mainly focuses on a typical document, PWP addresses how to separate HTML documents into blocks with the use of seven HTML tags which are *title*, *paragraph*, *headers*, *table*, *unordered list*, *ordered list*, and *horizontal rule*.

3) Similarity Value Calculation: the occurrences of queried words inside each block are located and a score is then assigned to each block with respect to the number of unique queried words. Let w_{q_i} be the weight of a queried term q_i, N_q the number of unique queried words in a query Q, n_q the number of unique queried words in a logical block B_i, W_t the total weight of all queried words, N_b the number of logical blocks within a document, and S_{B_i} the score of B_i. The scoring scheme of PWP is shown as follows.

procedure Similarity Value Calculation
(1) **for** `each queried term` q `in a query`, $w_{q_i} = 1/N_q$
(2) `initial score of each logical block` $S_{B_i} = 0$
(3) **for** `each logical block` B_i
  (4) **if** `all` q_i `appear in` B_i `then` $S_{B_i} = N_q$
    else $S_{B_i} = n_q \times 1/N_q$
(5) `the score of` d_i `is`

$$\texttt{PWP score}_{d_i} = \frac{\sum_{i=1}^{N_b} S_{B_i}}{N_q \times W_t \times N_b}$$

end procedure

3. EXPERIMENT

The effectiveness of PWP is compared to *Minimum Distance Between Queried Pair (MQP)*, employed as a part of Inquirus [6]. The MQP score of a document d is calculated as follows.

$$\text{MQP}_d = \frac{1}{c} \times \left(c - \frac{\sum_{i=1}^{N_d-1} \sum_{j=i+1}^{N_d} min(d_{i,j}, c)}{\sum_{k=1}^{N_d-1} (N_d - k)} \right), \quad (1)$$

where $d_{i,j}$ is the minimum distance between queried words i-th and j-th, N_d the number of unique queried words in a document, and c a constant specifying the maximum useful distance between queried words. To compare the effectiveness of both methods, a test system was constructed from three test sets based on the WT10G dataset provided from TREC, and 50 short queries created from the titles and descriptions of Topics 501-550. Each test set comprises three underlying engines; Test Set 1 comprises fub01be2, Juru-

WWW 2006, May 23–26, 2006, Edinburgh, Scotland.
ACM 1-59593-332-9/06/0005.

Full and ricMM; Test Set 2 includes Ntvenx2, PDWTAHDR and uwmtaw1; finally, icadhoc2, irtLnua and uncfsls are chosen for Test Set 3. Two criteria are employed to evaluate the effectiveness: the average interpolated precision-recall (AvgPrec) [1] and the average Discounted Cumulative Gain (AvgDCG) [5]. AvgPrec measures the accuracy of a retrieval strategy to order relevant documents toward the top rank, where AvgDCG assesses the effort spent by users to gain knowledge from result lists.

The experiment further investigates whether a combination of PWP and MQP, known as *PROX* hereafter, can improve the effectiveness as research has shown that a combination of similarity values from different retrieval strategies yield considerable improvement [4, 7]. In this experiment, CombSUM [4], the sum of individual similarity values, is employed to combine the scores of both methods. The effectiveness of PWP, MQP and PROX is demonstrated in Table 1.

Table 1: The AvgPrec and AvgDCG values of Test Sets 1-3

AvgPrec	Test Set 1	Test Set 2	Test Set 3
PWP	44.3036	37.9529	40.1721
MQP	42.0312	40.2333	46.5971
PROX	47.6536	40.1107	49.6201
AvgDCG	Test Set 1	Test Set 2	Test Set 3
PWP	6.8781	5.6924	4.7919
MQP	7.0130	5.5505	5.8376
PROX	7.8925	6.2807	6.0674

4. DISCUSSION

From Table 1, the results are not consistent as PWP sometimes provides lower AvgPrec and AvgDCG. This can be explained that MQP considers only the distances between all pairs of queried words; as a result, if a particular document contains only one out of many queried words, this document will be assigned a score of zero, as there is no distance between queried words. This situation may occur when users have no knowledge regarding the topics for which they are searching, they may enter words which are not related to or commonly used in the topics; for example, users may search for "*cloud* and *silhouette*" rather than "*cloud* and *formation*". In contrast, PWP looks at both the number of unique queried words and the occurrence of these words; therefore in the same situation PWP will assign scores to a document with respect to the occurrence of each word and its position.

Another limitation of MQP is that it tends to give equal scores to documents as it only takes into account the average minimum distance of queried words; as a consequence, it is difficult to rank many documents which obtain an equal score. An example of such is the name of an organization, such as "*Federal Housing Administration*". PWP can mitigate this limitation by assigning higher scores to documents containing more occurrences of queried words in different blocks.

Having alleviated the limitations of MQP, PWP has a restriction regarding multiple-topic documents, which tend to be longer and have more logical blocks compared to single-topic documents. Due to this, pages with multiple-topics are likely to obtain low scores from PWP even though their content covers a searched topic. This is mainly caused by PWP using the maximum possible score to normalize the raw score; hence, scores of related blocks will be reduced by the weight of unrelated blocks. On the other hand, MQP is not affected by multiple-topic documents due to consideration of only one occurrence of the queried-word set. Another limitation of PWP is that it could obtain lower AvgDCG values than MQP because PWP considers both the number of unique queried words and the number of their occurrences.

Table 1 further demonstrates that PROX generally provides significant improvement compared with PWP and MQP on their own, as the combination merges the advantages of both strategies. Consider the example shown in Table 2, where it can be seen that d_1 obtains the highest score from MQP but the lowest from PWP. This means that the distance among queried words of d_1 is the shortest compared with the others but the number of occurrences is the lowest. On the other hand, having the highest relevant judgment, d_2 is ranked second by both MQP and PWP. Once the similarity values are combined, the rank of d_2 is increased first due to a higher frequency of its occurrence, although the value of its average minimum distance is lower than d_1.

Table 2: Example of a combination between MQP and PWP

MQP			PWP			MQP+PWP		
ID	Score	Rel	ID	Score	Rel	ID	Score	Rel
d_1	0.95	0	d_3	0.84	1	d_2	1.54	2
d_2	0.78	2	d_2	0.76	2	d_3	1.49	1
d_3	0.65	1	d_1	0.41	0	d_1	1.36	0

5. CONCLUSION

PWP linearly combines scores of all logical blocks with the use of maximum normalization to represent a final document score, and the whole document is presented to users. Although PWP has limitations, it can alleviate the limitations of MQP and a combination of both can significantly improve the effectiveness, thus supporting previous research.

6. ACKNOWLEDGMENTS

This work is partly supported by the Royal Thai Government through a studentship of S. Palakvangsa-Na-Ayudhya.

7. REFERENCES

[1] R. Baeza-Yates and B. Ribeiro-Neto. Retrieval Performance Evaluation. In *Modern Information Retrieval*, pages 74–84. ACM Press, 1999.

[2] M. Beigbeder. Integrating Boolean and Vector Models of Information Retrieval with Passage Retrieval. In *Winter Intl. Symposium on Information and Communication Technologies*, 2005.

[3] J. P. Callan. Passage-Level Evidence in Document Retrieval. In *Proc. ACM SIGIR'94*, 1994.

[4] E. A. Fox and J. A. Shaw. Combination of Multiple Searches. In *Proc. 2nd TREC*, 1993.

[5] K. Järvelin and J. Kekäläinen. IR Evaluation Methods for Retrieving Highly Relevant Documents. In *Proc. ACM SIGIR'00*, 2000.

[6] S. Lawrence and C. L. Giles. Inquirus, the NECI Meta Search Engine. In *Proc. 7th Intl. World Wide Web Conference*, 1998.

[7] R. Manmatha and H. Sever. A Formal Approach to Score Normalization for Metasearch. In *Proc. Human Language Technology Conference*, 2002.

[8] R. Wilkinson. Effective Retrieval of Structured Documents. In *Proc. ACM SIGIR'94*, 1994.