

CITY UNIVERSITY OF HONG KONG  
香港城市大學

Effectiveness of Phrase in  
Information Retrieval  
片語在信息檢索的成效影響

Submitted to  
Department of Computer Science  
電腦科學系  
in Partial Fulfillment of the Requirements  
for the Degree of Doctor of Philosophy  
哲學博士學位

by

Chang Chor Ming Matthew  
鄭礎明

August 2008  
二零零八年八月

## Abstract

With the advent of Internet and advancement in computer technology, there are two notable challenges in document retrieval: retrieving huge amount of data *efficiently* and identifying the most useful documents, out of many relevant pages, *correctly*.

Although various phrase-finding and indexing methods have been proposed in the past, conclusions on the usefulness of phrases as indexing units have been generally inconsistent. Nevertheless, a number of recent research groups, including the leading groups who have participated in TREC campaigns, have used phrases as indexing units and have been able to obtain some improvement. As phrases have traditionally been regarded as precision-enhancing tools, recent research continues to apply the concept of phrase in different IR problems. In this thesis, following the tradition, we are interested in the concept of phrase in information retrieval, especially for document retrieval.

To address the two challenges, we first propose a *common phrase index* as an efficient index structure to support phrase queries in a very large text database. Our structure is an extension of previous index structures for phrases and achieves better query efficiency with modest extra storage cost. Further

improvement in efficiency can be attained by implementing our index, according to our observation of the *dynamic nature* of common word sets. In experimental evaluation, a common phrase index using 255 common words yields an improvement of about 11% and 62%, in query time for all queries and large queries (queries of long phrases) respectively, over an *auxiliary nextword index*. Moreover, it needs only about 19% extra storage space. Compared with an *inverted index*, our improvement works out to about 72% and 87% for all and large queries respectively. We also propose to implement a common phrase index with dynamic update feature. Our experiments show that more improvement in time efficiency can be achieved.

For improving the quality of retrieval results, we devise a proximity-based ranking function that combines an “*ordered loose phrase*” scoring with the state-of-the-art Okapi probabilistic model (BM25). We say that a phrase occurs in a document in an *ordered loose phrase* form, when the words of the phrase appear sufficiently close to each other, and in the same order as in the query. The occurrence of an ordered loose phrase, constituted by words of a query phrase in a document, may indicate a high relevance of the document to the query. We design our experiments using the query sets in TREC-11, TREC-12 and TREC-13 and the .GOV document collection. The results show that our method compares favorably with the pure BM25 and three recent works based on term proximity and co-occurrence, in most of the performance

measures. For TREC-12 and TREC-13, our results successfully demonstrate that our method can improve the quality of search results significantly.

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Overview . . . . .	1
1.2	Contributions . . . . .	3
1.3	Experiment setups . . . . .	7
1.4	Organization of the thesis . . . . .	9
<b>2</b>	<b>Literature review</b>	<b>10</b>
2.1	Overview of text representation . . . . .	10
2.2	Using phrase in information retrieval . . . . .	17
2.3	Previous works on phrase-based retrieval . . . . .	18
2.3.1	Recent research on phrase-based retrieval . . . . .	28
2.3.2	Previous works related to our focus areas . . . . .	41
2.4	Summary . . . . .	44
<b>3</b>	<b>Efficient indexing for phrase queries</b>	<b>46</b>

<i>CONTENTS</i>	iv
3.1 Preliminaries . . . . .	47
3.2 Common phrase indexes . . . . .	51
3.2.1 Common phrases . . . . .	51
3.2.2 Query evaluation . . . . .	56
3.2.3 Dynamic nature of common and rare words . . . . .	57
3.3 Implementation details . . . . .	58
3.3.1 Vocabulary and inverted files . . . . .	59
3.3.2 Dynamic feature . . . . .	63
3.4 Experimental results . . . . .	64
3.4.1 Comparison of efficiency and storage consumption . . . . .	65
3.4.2 Comparison between dynamic and static CPI . . . . .	69
<b>4 Query-independent phrase scoring for ranking</b>	<b>73</b>
4.1 Preliminary term-based ranking and motivations . . . . .	73
4.2 Query-independent phrase scoring . . . . .	75
4.2.1 Definition of phrase rank . . . . .	76
4.3 Experimental results . . . . .	82
4.3.1 Parameter tuning . . . . .	83
4.3.2 Comparisons of ranking functions . . . . .	84
4.4 Discussion . . . . .	86
<b>5 Ordered loose phrase scoring</b>	<b>89</b>

5.1	Okapi BM25 and term proximity scoring . . . . .	89
5.2	Ordered loose phrase scoring . . . . .	93
5.3	Experimental results . . . . .	97
5.3.1	Parameter tuning . . . . .	99
5.3.2	Comparisons of ranking functions . . . . .	100
5.3.3	Ordered and unordered query term sets . . . . .	103
5.3.4	Further investigation in TREC-11 . . . . .	105
<b>6</b>	<b>Conclusions and future work</b>	<b>108</b>
6.1	Conclusions . . . . .	108
6.2	Future work . . . . .	112

# List of Figures

3.1	Common Phrase Index Structure . . . . .	54
3.2	An example of vocabulary and index file for inverted index. . .	60
3.3	An example of vocabulary and index file for auxiliary nextword index. . . . .	61
3.4	An example of vocabulary and index file for common phrase index.	62
4.1	An example of word structure and word importance by term- based (frequency) ranking and <i>phrase rank</i> ranking. . . . .	80
4.2	An example of high $h(d)$ score of query term “construction”: a part of document G18-00-3748877 of .Gov collection. . . . .	81
4.3	An example of low $h(d)$ score of query term “construction”: a part of document G38-93-3383465 of .Gov collection. . . . .	81
4.4	Average P@10, MAP for TREC-11 and TREC-13 queries and .Gov collection for $tf \cdot idf + prank$ , for different $x$ . . . . .	85



4.5	Success@1, Success@5 and Success@10 for TREC-11 and TREC-13 queries and .Gov collection for $tf \cdot idf + prank$ , for different $x$ . . . . .	86
5.1	Average P@NR and P@10 for TREC-11, 12 and 13 for different $K$ . . . . .	100
5.2	Average recall versus precision for different ranking functions in TREC-11. . . . .	102
5.3	Average recall versus precision for different ranking functions in TREC-12. . . . .	103
5.4	Average recall versus precision for different ranking functions in TREC-13. . . . .	104

# List of Tables

1.1	TREC test questions and relevant pages in .Gov test collection.	8
3.1	Function Frequency of Query Last Word. . . . .	53
3.2	The number of queries with different sizes in Excite dating 1997 and 1999 . . . . .	65
3.3	Comparison of efficiency between Auxiliary Nextword Index (ANI) and Common Phrase Index (CPI) with <b>10 common words</b> . The codes T, D and I indicate the comparison in Time (average milliseconds), Retrieved Data Size (average mega bytes) and In- tersection (average number of intersect operations in thousands) respectively. ( $\Delta = \frac{CPI-ANI}{ANI}$ ). . . . .	66

3.4 Comparison of efficiency between Auxiliary Nextword Index (ANI) and Common Phrase Index (CPI) with **20 common words**. The codes T, D and I indicate the comparison in Time (average milliseconds), Retrieved Data Size (average mega bytes) and Intersection (average number of intersect operations in thousands) respectively.  $(\Delta = \frac{CPI-ANI}{ANI})$ . . . . . 66

3.5 Comparison of efficiency between Auxiliary Nextword Index (ANI) and Common Phrase Index (CPI) with **255 common words**. The codes T, D and I indicate the comparison in Time (average milliseconds), Retrieved Data Size (average mega bytes) and Intersection (average number of intersect operations in thousands) respectively.  $(\Delta = \frac{CPI-ANI}{ANI})$ . . . . . 67

3.6 Efficiency in average milliseconds of Inverted Index (II), Auxiliary Nextword Index (ANI) and Common Phrase Index (CPI) with **255 common words**.  $(\Delta(index) = \frac{index-II}{II})$ . . . . . 68

3.7 Size of Inverted Index (II), Auxiliary Nextword Index (ANI) and Common Phrase Index (CPI) with 10, 20 and 255 common words. 69

3.8 Efficiency in average milliseconds of Common Phrase Index with static (CPI) and dynamic (DCPI) set of **20 common words**.  $(\Delta(index) = \frac{DCPI-index}{index})$ . . . . . 71

3.9	The relationship between the number of intervals (or updates) and the number of queries involving common words. . . . .	72
4.1	Average P@10, MAP, Success@1, Success@5 and Success@10 results for different ranking functions or combinations (TREC-11) for $x = 100$ . . . . .	87
4.2	Average P@10, MAP, Success@1, Success@5 and Success@10 results for different ranking functions or combinations (TREC-12) for $x = 100$ . . . . .	87
4.3	Average P@10, MAP, Success@1, Success@5 and Success@10 results for different ranking functions or combinations (TREC-13) for $x = 100$ . . . . .	88
5.1	Four different normalizing factors (AvgTF is average term frequency and AvgDocLen is average document length). . . . .	97
5.2	Results of different measures for different ranking functions (TREC-11). . . . .	101
5.3	Results of different measures for different ranking functions (TREC-12). . . . .	101
5.4	Results of different measures for different ranking functions (TREC-13). . . . .	101

5.5	Effect of using ordered (O) and unordered (U) query terms in OLPS function for TREC-11, TREC-12 and TREC-13. ( $\Delta =$ $\frac{\textit{Unordered-Ordered}}{\textit{Ordered}}$ ). . . . .	104
5.6	Complete report on further investigation in TREC-11. . . . .	107

# List of Algorithms

1	Dynamic Update Function . . . . .	63
2	Delete Common Phrase Function . . . . .	63
3	Create Common Phrase Function . . . . .	64