On Identifying Phrases Using Collection Statistics

Simon Gog[1,2], Alistair Moffat[1], and Matthias Petri[1]

[1] Department of Computing and Information Systems,
The University of Melbourne, Australia 3010
[2] Institute of Theoretical Informatics,
Karlsruhe Institute of Technology, Germany

Abstract. The use of phrases as part of similarity computations can enhance search effectiveness. But the gain comes at a cost, either in terms of index size, if all word-tuples are treated as queryable objects; or in terms of processing time, if postings lists for phrases are constructed at query time. There is also a lack of clarity as to which phrases are "interesting", in the sense of capturing useful information. Here we explore several techniques for recognizing phrases using statistics of large-scale collections, and evaluate their quality.

1 Introduction and Related Work

Many concepts are expressed as multi-word expressions, for example, "United States of America", and "letter of condolence". But most information retrieval techniques segment both queries and source documents in to words, and compute similarity over those words as if they were independent, an apparent mismatch that suggests that improved retrieval effectiveness is possible if phrases are also employed. For example, in 1991 Croft et al. [5] wrote "there has always been the feeling that phrases, if used correctly, should improve the specificity of the indexing language". That goal has been realized recently by a range of techniques that make use of phrases and consistently – if perhaps modestly – improve search quality [2, 3, 6, 8, 9, 19].

Computational cost has been a factor that has prevented the wider use of multi-word phrases in query evaluation. If index space is a dominant concern, the most economical way of handling phrases is to store a positional inverted index [20], and compute postings list intersections as queries are handled. If query time is important, then phrase-based indexing approaches can be employed, in which some or all terms' postings lists are augmented by information about following words [17]. A third possibility is to directly index certain phrases, and give them postings lists. The number of phrases admitted to the index provides a tunable tradeoff between index size and execution cost.

The question then is, which phrases should be indexed? Based on characteristics of the document collection, is it possible to generate an ordering of phrases that could then be used to decide which phrases should be granted dedicated postings lists? If the text has embedded markup, then it can be used to identify word sequences of interest, including those to be displayed as headings or with different typography, and those used as anchors for hyperlinks [7]. For plain text, a range of automatic extraction methods based on occurrence frequencies have been proposed from both linguistics [4] and computing [12, 16, 18]. However the sheer volume of data that must be processed in order to compute word and phrase statistics over large texts has been an impediment in the past. Our work in this paper is possible because of recent advances in data structures, including the development of succinct self-index technologies, see Navarro [11] for an overview and Patil et al. [13] for one implementation approach.

2 Phrase-Finding

We consider three methods for finding multi-word phrases in text. Two of them are based on previous mechanisms for identifying word bigrams of interest; we extend them to the multi-word situation.

Mutual Information. The concept of mutual information can be used to determine an *association ratio* between two words [4]. Given words w_1 and w_2, mutual information compares the probability of a co-occurrence to the probabilities of observing each word independently. If w_1 and w_2 are associated, the observed probability of the two words occurring together will be much larger than the probability of a co-occurrence by chance. Similar to Church and Hanks [4], we use the number of occurrences of word w_i normalized by the size of the corpus as an estimate of its probability $P(w_i)$. Multi-word expressions can be handled by extending the formulation (see also Van de Cruys [14]):

$$\text{MI-}\textsc{ext}(w_1, w_2 \ldots w_n) = \log_2 \frac{P(w_1 w_2 \ldots w_n)}{P(w_1)P(w_2)\cdots P(w_n)} \,.$$

Pearson's χ^2*.* The χ^2 ($\textsc{Chi}^2$) metric can also be used to test the independence of an observation. The independence of a word bigram $w_1 w_2$ is evaluated by comparing its observed frequency in the collection to its expected frequency [15]. Expected frequencies require bigram statistics such as $F(w_1 w_2)$ and $F(w_1 \neg w_2)$ to be computed, both of which can be efficiently performed using a self-index, a technology that has only recently been available at the required scale.

We extend bigram scores to allow computation of n-gram scores: if $\chi^2(w_i w_{i+1})$ is the score for the word-pair w_i followed by w_{i+1}, we compute

$$\textsc{Chi}^2\text{-}\textsc{ext} = \left(\min_{1 \le i < n} \chi^2(w_i w_{i+1}) \right) \cdot \ln n \,,$$

where the multiplication by $\ln n$ counteracts the diminishing nature of the $\min$ operator, and up-weights longer phrases that are the concatenation of shorter stronger ones.

Existence. We implemented one further mechanism, denoted $\textsc{Existence}$, defined as the ratio between the number of documents which contain all words of the candidate phrase and documents which contain the candidate phrase. In this mechanism document boundaries are used, a concept not employed in the first two approaches. For example, if there are five documents in the collection that contain all of w_1, w_2, and w_3, and the sequence $w_1 w_2 w_3$ appears as a phrase in three of them, then the (undamped) conditional probability of existence is given by $3/5 = 0.6$. In practice, to avoid every unique substring being assigned a score of 1.0, we use a dampening constant K, and compute

$$\textsc{Existence}(w_1 w_2 \ldots w_n) = \frac{F(w_1 w_2 \ldots w_n)}{F(w_1, w_2, \cdots, w_n) + K}$$

Rank	MI-EXT	CHI^2/ CHI^2-EXT	EXISTENCE
1.	new mexico senator pete domenici	**punta gorda**	**sri lanka**
2.	**equine protozoal myeloencephaliti**	**puerto rico**	**punta gorda**
3.	virus hpv genital wart	**bryn mawr**	**corpus christi**
4.	methyl ether tertiary butyl	**saudi arabia**	**puerto rico**
5.	civil war 1861 1865	**corpus christi**	**st croix**
6.	1922 fordney mccumber	**sri lanka**	**pro tempore**
7.	oldsmobile ciera cutlass	**cabernet sauvignon**	**saudi arabia**
8.	bull terrier staffordshire	**monte carlo**	**los angeles**
9.	holiday inn sunspree	antirobe aquadrop	**wilke barre**
10.	pratt whitney jt8d	**chichen itza**	**bryn mawr**

Table 1: Example stemmed phrases extracted from *Query Set I*. The CHI^2-EXT method produced the same top-10 results as CHI^2. Phrases corresponding to Wikipedia page titles are in bold.

where $F(s)$ is the document frequency of s in the collection, and $K = 5$ is used, to ensure that a phrase occurs at least five times if its score is greater than 0.5.

Stop words. We further apply stop word trimming. Any word for which the maximum value of the BM25 similarity computation between the word and any document is less than one when using the default parameters (see Zobel and Moffat [20]) is defined to be a stop word. Stop words at the beginning and end of candidate phrases are removed.

3 Experiments and Results

Source Data. We took the 426 GB Gov2 collection and built a self-index structure [11]. To determine potential phrases, we randomly sampled two query sets each containing $10{,}000$ queries from the TREC Million Query Track. We selected unique queries containing two or more words such that each word appeared at least once in Gov2. Each sub-phrase in each query was then evaluated as a candidate using the index, and assigned a score by each of the mechanisms described in the previous section. For example, a four word query generates six candidate phrases.

Table 1 lists the top phrases discovered using *Query Set I*. The Mutual Information-based approach favors longer phrases, whereas the other methods rank two-word phrases higher. The first phrase of length larger than two occurs at rank 160 for CHI^2-EXT and at rank 60 for EXISTENCE.

Forming Judgments. We then sought to compare the lists of candidate phrases. The first step is to make a judgment, for each identified word sequence, as to whether it is indeed a plausible phrase. Once each algorithm's phrase ranking has been suitably annotated, a score can be derived. But generating labeled evaluation data is problematic. One option is to employ experts to create "gold standard" determinations. Another is to use non-expert judgments via a crowd-sourcing service. Both methods have their disadvantages – experts are expensive, and will not necessarily agree with each other no matter how precise their instructions; the wisdom of the crowd may generate more reliable data overall for less money, but is vulnerable to hasty workers.

To obtain preliminary results, we have employed a third alternative, and make use of Wikipedia for implicit decisions. In particular, many multi-word entities have Wiki pages associated with them, for example, `http://en.Wikipedia.org/wiki/White_House` is the page for the "*White House*".

To automate the judging process we downloaded $10{,}947{,}620$ Wiki page titles[3]. The titles were filtered and normalized as follows: categorization suffixes of titles were deleted (for example, the suffix "_(film)" in the title "Personal_Best_(film)"); single term titles were removed; underscores were translated to spaces; and words lowercased and stemmed using a Krovetz stemmer. Phrases were then deemed to be valid if and only if they were in this processed list. This mechanism fails for many interesting phrases, but also works a surprising fraction of the time, including, for example, for "*standing ovation*", "*personal best*", and "*laugh out loud*".

Table 2 gives a breakdown of the set of reference phrases identified from the Wiki URLs. More than seven million Wiki pages had multi-word titles, with around half of them two words long, a quarter three words long, and so on. The "7+" category includes phrases such as "*1954 britain empire and commonwealth games medal count*". A further $3{,}562{,}553$ Wiki page URLs consisted of a single word, or were explicit disambiguation pages. The phrases identified were then used as ground truth in the evaluation.

Length	Number	Fraction
2	3,498,885	47.4%
3	1,895,699	25.7%
4	914,401	12.4%
5	483,618	6.5%
6	264,400	3.6%
7+	328,064	4.4%
Total	7,385,067	100.0%

Table 2: Distribution of Wiki URLs.

Applying a Metric. Once judgments have been formed, a metric can be used to compute a quality score for the ordered list of phrases generated by each of the algorithms. Any IR metric can be used, provided that it is agnostic to the total number of positive judgments. For example, the first $1{,}000$ phrases in each list might be examined, and the fraction of them that are valid expressed as a *precision*@$1{,}000$ score. In the results reported below, we use the top-weighted arbitrary-depth RBP metric [10], with two parameters, $p = 0.99$ and $p = 0.999$, in both cases using generated rankings of $10{,}000$ candidate phrases in decreasing score order. With these parameters, rank-biased precision (RBP) provides deep coverage in the ranked list (to an expected depth of 100 items and $1{,}000$ items, respectively), with a relatively mild bias in favor of positions near the front of the ranking. With p values near 1.0, RBP can be expected to yield outcomes that are closely correlated with precision scores when evaluated to comparable cutoffs.

Results. Table 3 shows that the methods achieved consistent scores over two query sets, and that the EXISTENCE and CHI^2 methods achieve good performance. Note that these are all lower bounds – the Wiki URLs used to provide relevance judgments are not a complete set of phrases, and are biased in favor of entities such as events, people, and places. False positives occur when a candidate phrase is scored highly by an algorithm, but does not appear in the Wiki listing; false negatives when a phrase that is a Wiki page title, is scored lowly by the algorithm. Table 4 shows the top ten false positives identified by the EXISTENCE method, and the ten lowest-scoring candidate phrases that

[3] `File enwiki-20140502-all-titles-in-ns0`, accessed 10 June 2014.

p	Query set	MI-ext	Chi^2	Chi^2-ext	Existence
0.99	I	0.380	0.824	0.823	0.857
0.99	II	0.406	0.831	0.830	0.858
0.999	I	0.406	0.557	0.546	0.562
0.999	II	0.331	0.553	0.538	0.554

Table 3: Rank-biased precision scores for three phrase-finding mechanisms.

False positives		False negatives	
30.	california arnold	19944.	**social death**
31.	marriott wardman	19804.	**nation league**
32.	canton massillon	19630.	**civil movement**
39.	mountain lab	19463.	**project jersey**
50.	cmc heartland	19294.	**independence declaration**
66.	displace homemaker	19247.	**early island**
70.	paul biane	19089.	**north purchase**
71.	2006 2007	19068.	**last snow**
90.	nasa launch	19047.	**thomas plate**
104.	phs 5161	18913.	**satellite states**

Table 4: False positives and false negatives for the Existence method and *Query Set I*.

corresponded to Wiki page titles. False positives tend to follow certain patterns: "*paul biane*", "*cmc heartland*" and "*canton massillon*" are names of people, companies or places not present in Wikipedia; and "*PHS 5161*" is the name of a form referenced often in Gov2. False negatives include ambiguous phrases such as "*last snow*", which is the name of a novel not referenced in Gov2; similarly, "*project jersey*" refers to a java framework only created after the Gov2 corpus was crawled.

4 Conclusion and Future Work

To identify phrases in collections that might warrant being explicitly indexed so as to provide fast querying, we have explored techniques for automatically extracting them using only the statistics provided by the collection itself. Using Wikipedia page titles as a reference point, we have compared those techniques, and found that the new document-aware Existence method creates the best set of phrase candidates. The benefit of the new methodology – compared, for example, to the obvious alternative of simply using the Wikipedia titles directly – is that an ordered list of phrases is created, and that they are sourced from the collection. The latter is important when technical or medical text is being stored, since Wikipedia titles would not provide useful guidance.

Our next task is to embed the phrase-finding technology into a retrieval system. That will involve the complete suffix tree traversal of the text to find candidate phrases. An index can then be constructed to fit any given space bound, taking terms in to it, plus postings lists, for how ever many phrases can best fit. It will then be possible to fully explore the complex relationships between query processing speed, index space required, and retrieval effectiveness; see, for example, Anand et al. [1].

Acknowledgments. This work was supported under the Australian Research Council's Discovery Projects funding scheme (project DP110101743), and by the Victorian Life Sciences Computation Initiative (grant VR0052) on its Peak Computing Facility at the University of Melbourne, an initiative of the Victorian State Government, Australia.

References

1. Anand, A., Mele, I., Bedathur, S., Berberich, K.: Phrase query optimization on inverted indexes. In: Proc. CIKM. pp. 1807–1810 (2014)
2. Broschart, A., Berberich, K., Schenkel, R.: Evaluating the potential of explicit phrases for retrieval quality. In: Proc. ECIR. pp. 623–626 (2010)
3. Chieze, E.: Integrating phrases in precision-oriented information retrieval on the web. In: Proc. Conf. Inf. Know. Eng. pp. 54–60 (2007)
4. Church, K.W., Hanks, P.: Word association norms, mutual information, and lexicography. Comp. Ling. 16(1), 22–29 (1990)
5. Croft, W.B., Turtle, H.R., Lewis, D.D.: The use of phrases and structured queries in information retrieval. In: Proc. SIGIR. pp. 32–45 (1991)
6. Geva, S., Kamps, J., Lehtonen, M., Schenkel, R., Thom, J.A., Trotman, A.: Overview of the INEX 2009 ad hoc track. In: Proc. INEX. pp. 4–25 (2009)
7. Lehtonen, M., Doucet, A.: Phrase detection in the Wikipedia. In: Proc. INEX. pp. 115–121 (2007)
8. Liu, S., Liu, F., Yu, C.T., Meng, W.: An effective approach to document retrieval via utilizing wordnet and recognizing phrases. In: Proc. SIGIR. pp. 266–272 (2004)
9. Metzler, D., Croft, W.B.: A Markov random field model for term dependencies. In: Proc. SIGIR. pp. 472–479 (2005)
10. Moffat, A., Zobel, J.: Rank-biased precision for measurement of retrieval effectiveness. ACM Trans. Information Systems 27(1), 2.1–2.27 (2008)
11. Navarro, G.: Spaces, trees and colors: The algorithmic landscape of document retrieval on sequences. ACM Comp. Surv. 46(4), 1–47 (2014)
12. Nevill-Manning, C.G., Witten, I.H.: Compression and explanation using hierarchical grammars. Comp. J. 40(2/3), 103–116 (1997)
13. Patil, M., Thankachan, S.V., Shah, R., Hon, W.K., Vitter, J.S., Chandrasekaran, S.: Inverted indexes for phrases and strings. In: Proc. SIGIR. pp. 555–564 (2011)
14. Van de Cruys, T.: Two multivariate generalizations of pointwise mutual information. In: Proc. Wkshp. Distr. Semantics & Compositionality. pp. 16–20 (2011)
15. Villada Moirón, M.B.: Data-driven identification of fixed expressions and their modifiability. Ph.D. thesis, University of Groningen (2005)
16. Wang, X., McCallum, A., Wei, X.: Topical n-grams: Phrase and topic discovery, with an application to information retrieval. In: Proc. ICDM. pp. 697–702 (2007)
17. Williams, H.E., Zobel, J., Bahle, D.: Fast phrase querying with combined indexes. ACM Trans. Information Systems 22(4), 573–594 (2004)
18. Witten, I.H., Paynter, G.W., Frank, E., Gutwin, C., Nevill-Manning, C.G.: KEA: Practical automatic keyphrase extraction. In: Proc. ACM Conf. Dig. Lib. pp. 254–255 (1999)
19. Zhang, W., Liu, S., Yu, C.T., Sun, C., Liu, F., Meng, W.: Recognition and classification of noun phrases in queries for effective retrieval. In: Proc. CIKM. pp. 711–720 (2007)
20. Zobel, J., Moffat, A.: Inverted files for text search engines. ACM Comp. Surv. 38(2), 6–1 – 6–56 (2006)